Analysis of Resource Increase and Decrease Algorithm in Wireless Sensor Networks

Jaewon Kang
Computer Science
Rutgers University
jwkang@cs.rutgers.edu

Yanyong Zhang
WINLAB
Rutgers University
yyzhang@winlab.rutgers.edu

Badri Nath
Computer Science
Rutgers University
badri@cs.rutgers.edu

Abstract

In this paper, we first attempt to formally define the resource control framework that adjusts the resource provisioning at the hotspot during congestion. In an effort to find the optimal resource control under the fidelity and energy constraints, we present a resource increase and decrease algorithm called Early Increase/Early Decrease (EIED) that tries to adjust the effective channel capacity quickly to the incoming traffic volume in an energy-efficient manner, thereby increasing the fidelity (or accuracy) level observed by the application during congestion. Under the framework of energy-constrained optimization, we prove this algorithm incurs the lowest overhead of energy consumption for the given fidelity level that is required by the application.

1. Introduction

As the technologies in MEMS, processor design, and wireless communication advance, a wide range of remote monitoring applications using networks of sensor nodes receive much attention recently. When targeted events are detected (referred to as *crisis state*) in this network, the traffic volume often exceeds the amount of available resources such as channel capacity [8, 9], thus resulting in congestion at various nodes. As a result, the network may enter an unstable state in which packets are randomly dropped, a severe case of which is called congestion collapse. Intuitively, we have two options to alleviate congestion: throttling the incoming traffic volume (referred to as *traffic controlling*) and increasing the available resources (referred to as *resource controlling*).

Although traffic control strategies are effective in traditional wired networks and are also suggested in some sensor network scenarios [8, 9, 13], they are unsuitable for our purpose for two main reasons. Firstly, reducing the traffic during a crisis state is unacceptable. The data during a crisis state are of great value, often critical, to the applications such as earthquake or fire monitoring systems. Failure to ship some of these data to the sink can hurt the applications' accurate monitoring and defeat the very reason why the sensor network was deployed in the first place. Secondly, increasing resource provisioning during congestion is easier in sensor networks (unlike its wired or other wireless counterparts) due to its elastic availability of resources [1, 11, 12]. Therefore, a natural way of managing these limited resources (such as channel capacity, remaining energy, transmission power, etc.) is to conserve as much resources as possible during a dormant state while exploiting them wisely during crises.

Even though congestion control in sensor networks is important in the sense that it should be performed to make sure the reported data amount exceeds the required fidelity level and the lifetime of the network is optimized by saving scarce energy during congestion, it has received little attention until recently and most of recent studies focus on controlling traffic to alleviate congestion [8, 9, 13]. The question, "*How should network resources such as wireless channel capacity be systematically controlled in conjunction with the congestion level under fidelity and energy constraints?*", has not been studied rigorously even though resource control strategies have the potential of minimizing energy consumption during a dormant state as well as accommodating the traffic surge during a crisis state, thus avoiding congestion and satisfying the fidelity requirement. In this paper, we set out to answer this challenging question.

The remainder of this paper is organized as follows. We introduce related work in Section 2. In Section 3, we explain the traffic increase and decrease algorithm. In Section 4, we formulate the goals of resource controlling along with the performance metrics for the potential resource controls. In Section 5, we propose a resource increase and decrease algorithm called Early Increase/Early Decrease (EIED) that incurs the lowest overhead under fidelity and energy constraints. In Section 6, the EIED algorithm is verified by simulation. Section 7 concludes this paper.

2. Related Work

Most of prior work have been focused on the efficient usage of traffic control during congestion in sensor networks. CODA [9] presents the first detailed study on congestion detection and avoidance in sensor networks. In CODA, as soon as a node detects congestion, it broadcasts a backpressure message upstream. An upstream node that receives the backpressure throttles its outgoing traffic by dropping packets or by forwarding the backpressure message upstream again. Sankarasubramaniam, Akan and Akyildiz [8] proposed an event-to-sink reliable transport (ESRT) protocol, which can serve as a congestion control protocol as well since congestion is considered to be the primary cause for unreliable delivery. Hull, Jamieson, and Balakrishnan [6] studied three congestion control techniques: hop-by-hop traffic control, limiting source rate, and a prioritized medium access control (MAC). They show network congestion can be greatly alleviated when these three traffic control techniques operate in concert rather than in isolation. Ee and Bajcsy [3] proposed a distributed congestion control scheme based on hop-by-hop automatic repeat request (ARQ) in many-to-one routing scenario. To avoid congestion, motes are allowed to generate data at a rate network can handle. Woo and Culler [10] proposed an adaptive traffic control scheme in which locally generated traffic and route-through traffic are assigned bandwidth proportionally to provide fairness among the flows with different path lengths, which also avoids congestion. Yi and Shakkottai [13] proposed a hop-by-hop congestion control scheme that allocates bandwidth to various users in a fair manner. They show a hop-by-hop traffic control scheme push-backs and spreads congestion over space, called spatial spreading, leading to scattered small peak loads.

3. Traffic Increase and Decrease Algorithm

Controlling traffic in the context of congestion avoidance has been extensively studied mostly through the active queue management (AQM) in wired networks. Two main goals of the traffic control strategy are *high resource utilization* and *fairness*. To achieve these goals, each node in the network senses its resource utilization state and feeds this information fairly back to either its previous hop nodes (hop-by-hop strategy) or its sources of traffic (end-to-end strategy) who then adjust their outgoing traffic volume.

Depending on the relation between the offered load at time t, $T_i(t)$ and the fixed resource amount R_i at node i, the traffic control component of node i sends a binary feedback to the source of each flow shown below.

$$b_i(t) = \begin{cases} 0 & if \ T_i(t) < R_i \quad (underloaded) \\ 1 & if \ T_i(t) \geq R_i \quad (overloaded) \end{cases} \quad (1)$$

The feedback of 1 means that the network is overloaded and the source needs to reduce its traffic (the rate at which it pumps data); 0 means that the flow can increase its traffic. The source has several options in systematically decreasing/increasing its traffic after receiving the feedback. In [2], Chiu and Jain have proved a simple additive increase and multiplicative decrease (AIMD) algorithm shown in Equation 2 satisfies the sufficient conditions for convergence to a high utilization and fair state in a distributed manner, regardless of the starting state of the network.

$$T_i(t+1) = \begin{cases} T_i(t) + a_I & if \ b_i(t) = 0 \\ m_D T_i(t) & if \ b_i(t) = 1, \end{cases} \quad (2)$$

where a_I and m_D are constants and $a_I > 0$ and $0 < m_D < 1$. This algorithm has been extensively studied in the literature and adopted by some versions of TCP congestion window control algorithms [4, 5].

4. Problem Formulation

In this section, we attempt to formally define the framework of the resource control strategy for the first time. The framework of the resource control strategy differs from that of the traffic control strategy in several ways. Firstly, node i's effective resource capacity (shown in Equation 1), which has been fixed to R_i in the traffic control framework, is now represented as a time-varying function, $R_i(t)$ due to its elastic availability. Secondly, fairness among competing flows are not realized by the resource control operation alone unless per-flow resource provisioning is implemented, which, however, incurs too much overhead. Thirdly, the resource control operation can be performed locally due to its centralized view of the incoming traffic load. Fourthly, an increase operation is applied during the overloaded state and a decrease operation is applied during the underloaded state, which is opposite to the traffic control scheme. Finally, it is rather costly to increase/decrease resource provisioning than to increase/decrease the outgoing traffic volume. For example, compare the cost of TCP's congestion window adjustment with the cost of channel capacity adjustment by changing transmission power or the number of active nodes in the sensor field.

The criteria for selecting a resource control policy are as follows.

- *Fidelity:* One of the motivations for resource controlling is to ship as much incoming traffic as possible, at least above the required fidelity level F (in bits per unit of time), to one or more sinks during a crisis state, so that the delivered data can produce a meaningful view of the sensed event and subsequently incur necessary actions by the application that extracts the delivered data from the sinks.

- *Energy Efficiency:* Increasing the effective resource capacity can improve the quality of service observed by the application, but it may also increase the total energy consumption due to the higher data transmission rate and the maintenance overhead of the increased resource provisioning. In addition, if the increased resource capacity still does not accommodate the incoming traffic, some of the traffic should be discarded due to congestion, thereby nullifying the energy expended for receiving the traffic from the previous hop nodes.

- *Packet Energy Efficiency:* Packet energy indicates the average energy consumed by the network to successfully forward a packet from a source to a sink.

The goal of the resource control strategy is how to systematically adjust the effective resource capacity available to a node in conjunction with the incoming traffic volume while satisfying above criteria. The notations of $T_i(t)$ and $R_i(t)$ are used throughout this paper indicating the aggregate incoming traffic and the effective resource capacity of node i at time t, respectively.

5. Proposed Resource Control Policy

5.1. Congestion Feedback

Unlike the traffic control strategy which employs a binary congestion feedback shown in Equation 1, a trinary congestion feedback is used by the resource control strategy as follows:

$$b_i(t) = \begin{cases} 0 & if \ T_i(t) < R_i(t) - w_l \\ -1 & if \ R_i(t) - w_l \leq T_i(t) \leq R_i(t) - w_u \\ 1 & if \ T_i(t) > R_i(t) - w_u, \end{cases}$$
(3)

where w_u and w_l are constants and $w_u \geq 0$, $w_l \geq 0$, and $w_l \geq w_u$. The upper and lower watermarks of node i at time t are expressed as $(R_i(t) - w_u)$ and $(R_i(t) - w_l)$, respectively. $(T_i(t) - R_i(t))$ denotes the congestion degree or the resource deficiency of node i at time t.

When node i's aggregate incoming traffic volume $T_i(t)$ falls below the lower watermark, the congestion feedback is set to 0 indicating the node's current level of effective resource capacity can be lowered. When the node's aggregate incoming traffic volume lies between the upper and lower watermarks, the congestion feedback function returns -1 indicating no resource control is required. When the node's aggregate incoming traffic volume exceeds the upper watermark, the congestion feedback function returns 1 indicating the node's effective resource capacity should be increased.

5.2. Resource Control under Fidelity and Energy Constraints

In this section, we find the resource control that achieves the highest fidelity level with the lowest energy consumption, thereby maximizing the packet energy efficiency, which combines those two key criteria together. The problem of maximizing the packet energy efficiency, which is measured from the whole network, can be reduced to the problem of minimizing the *bit energy* consumed by each individual node in order to successfully forward 1 bit to the next hop if the packet size is constant. The bit energy consumed by a node is calculated by dividing the total energy consumption by the total outgoing traffic that are successfully received by the next hop node. Therefore, the goal of the resource control strategy under fidelity and energy constraints during the event period of D is represented as follows:

$$minimize \quad \frac{\int_0^D E_i(t)dt}{\int_0^D min(T_i(t), R_i(t))dt}, \tag{4}$$

where $E_i(t)$ is the total energy consumed by node i at time t and $min(T_i(t), R_i(t))$ indicates the outgoing traffic from node i at time t. First, we define three energy constants as follows:

- e_l: average energy consumed by 1 unit of channel capacity of a node per unit of time when the node is idle. 1 unit of channel capacity can accommodate 1 bit of incoming traffic. Therefore, if a channel capacity of R is idle for the period of d, the total energy consumption is $e_l R d$.

- e_r: average energy consumption by a node to successfully receive 1 bit from its neighbor node. e_r is calculated by dividing the total energy expended by the receiving activities of a node by the total amount of traffic (in bits) received from the previous hop nodes.

- e_t: average energy consumption by a node to successfully transmit 1 bit to the next hop node. e_t is calculated by dividing the total energy expended by the transmitting activities of a node by the total amount of traffic (in bits) successfully forwarded to the next hop nodes.

Generally, $e_t \gg e_r \geq e_l$ holds [12]. Now, we'll find the resource control that minimizes the bit energy consumption under an arbitrary incoming traffic function $T_i(t)$ to find the optimal resource control policy.

Theorem The bit energy consumed by node i is minimized when the node's effective channel capacity is equivalent to the aggregate incoming traffic, i.e. $R_i(t) = T_i(t)$.
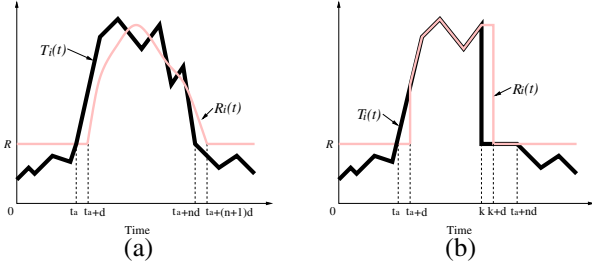
Figure 1. (a) Arbitrary resource control and (b) EIED combined with traffic controlling

Figure 2. Congestion at intermediate nodes and resource controlling by creating multiple paths.

Proof To find a resource control that minimizes the bit energy consumption of node i, arbitrary $R_i(t)$ and $T_i(t)$ are depicted in Figure 1(a). Each bit received but discarded later (possibly due to queue overflow or high channel contention) by node i consumes e_r energy. Each bit received and forwarded to next hop node by node i consumes $(e_r + e_t)$ energy. Therefore, the total energy consumed by node i between t_a and $t_a + (n+1)d$ is

$$\int_{t_a}^{t_a+(n+1)d} E_i(t)dt, \qquad (5)$$

where

$$E_i(t) = \begin{cases} (e_r + e_t)T_i(t) + e_l(R_i(t) - T_i(t)) & if \ T_i(t) < R_i(t) \\ (e_r + e_t)T_i(t) & if \ T_i(t) = R_i(t) \\ e_r T_i(t) + e_t R_i(t) & if \ T_i(t) > R_i(t) \end{cases}$$
$$= \begin{cases} (e_r + e_t - e_l)T_i(t) + e_l R_i(t) & if \ T_i(t) < R_i(t) \\ (e_r + e_t)T_i(t) & if \ T_i(t) = R_i(t) \\ e_r T_i(t) + e_t R_i(t) & if \ T_i(t) > R_i(t). \end{cases}$$

The successfully forwarded traffic volume from node i at time t is $min(T_i(t), R_i(t))$ and represented as follows in each case.

$$min(T_i(t), R_i(t)) = \begin{cases} T_i(t) & if \ T_i(t) < R_i(t) \\ T_i(t) & if \ T_i(t) = R_i(t) \\ R_i(t) & if \ T_i(t) > R_i(t) \end{cases} \qquad (6)$$

Therefore, the bit energy consumed by node i at time t is represented as follows:

$$\begin{cases} (e_r + e_t - e_l) + \frac{R_i(t)}{T_i(t)}e_l & if \ T_i(t) < R_i(t) \\ e_r + e_t & if \ T_i(t) = R_i(t) \\ \frac{T_i(t)}{R_i(t)}e_r + e_t & if \ T_i(t) > R_i(t) \end{cases} \qquad (7)$$

In the resource control scheme, only $R_i(t)$ is adjustable. Therefore, the bit energy consumption is minimized to $(e_r + e_t)$ when $R_i(t)$ is equal to $T_i(t)$. $\qquad \square$

Based on the Theorem, a new resource control policy can be defined as follows:

$$R_i(t+1) = \begin{cases} T_i(t) & if \ b_i(t) = 0 \\ R_i(t) & if \ b_i(t) = -1 \\ T_i(t) & if \ b_i(t) = 1 \end{cases} \qquad (8)$$
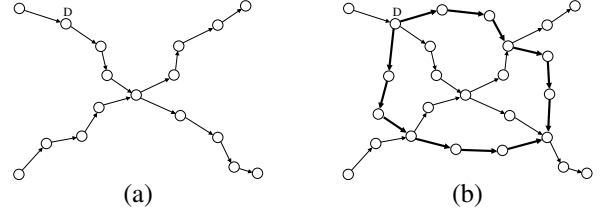
The constants, w_u and w_l used in the trinary congestion feedback shown in Equation 3 are 0 in this equation. This control keeps adjusting its effective channel capacity to the aggregate incoming volume, spending or conserving its limited energy as *early* as possible. We call this policy *Early Increase/Early Decrease* (EIED). Above policy assumes each node can expend its energy infinitely to accommodate the incoming traffic, which, however, is unrealistic. Figure 1(b) shows the EIED policy combined with traffic controlling. Node i stops the resource control operation at time k and starts the traffic control operation by asking its previous hop node(s) to reduce the incoming traffic volume to the default resource capacity R at time k, i.e. $T_i(k) = R$. The effective channel capacity is also reduced to the default resource capacity R at time $k + d$, i.e. $R_i(k + d) = R$. The traffic controlling can be triggered (i) when the cumulative energy consumption after the resource control scheme is triggered exceeds the energy budget B per event or (ii) when the cumulative outgoing traffic exceeds the required fidelity amount. Large k will increase the fidelity amount, but consume more energy, thereby exceeding the energy budget B.

6. Verification of EIED by Simulation

In this section, the effectiveness of the EIED policy is verified under the fidelity and energy consumption metrics and compared with the *ideal traffic control scheme*. The ideal traffic control scheme adjusts the reporting rates of source nodes, so that it can achieve the highest fidelity level for the given channel capacity.

6.1. Simulation Environments

To make a dense network for the strong connectivity, 225 nodes with a communication radio range[1] of 50m are randomly deployed in the 530m by 530m sensor field. Therefore, the maximum neighbor degree is around 6 when the

---

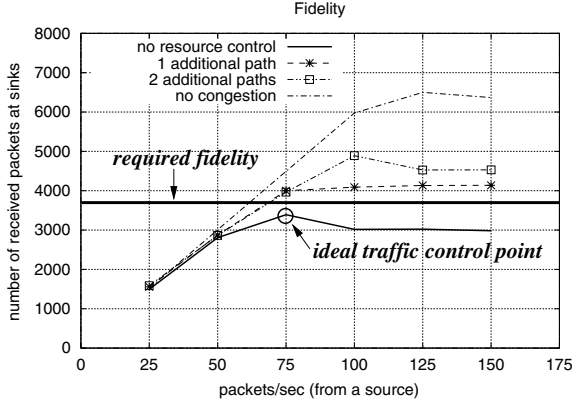[1]The interference radio range of a node is set to be twice the communication range.

Figure 3. Fidelity amount observed by sinks

Figure 4. Energy consumption per delivered packet

radio of each node is on. A scenario of alleviating congestion under cross traffic is simulated as shown in Figure 2(a). Only active nodes are depicted. After node D detects congestion, it keeps creating an additional path quickly until congestion is alleviated and tears down the additional paths when its channel capacity is under-utilized. Therefore, the nodes cooperate to always keep the end-to-end channel capacity to the incoming volume level by adjusting the number of available paths. Figure 2(b) shows that the end-to-end channel capacity of a flow is increased by dynamically creating 2 additional paths. To create an additional path, directed diffusion [7] is used for routing protocol.

To create a hotspot at an arbitrary location by two crossing flows, two sinks that are fully apart are randomly picked up in the sensor field in each instance of simulation. For each sink, 3 sources are randomly picked up, resulting in 6 sources in total. Each of 3 sources for the given sink exclusively selects one of 3 event periods of 10-20, 30-40, and 50-60 seconds during the simulation time of 70 seconds and reports the event at a high rate which we will vary during its event period. The event reporting rate of a source during a dormant state is 1 packet/second. The underlying MAC protocol is IEEE 802.11 DCF with RTS/CTS. The data packet size is 100 bytes and the raw channel capacity is 2 Mbps.

6.2. Fidelity

Figure 3 shows the fidelity amount, i.e. total amount of delivered data to the sinks, during the event period with respect to different reporting rates at a source. As a baseline scenario, sources and sinks are carefully selected so that no congestion can happen, which is drawn on "no congestion" curve. To find an ideal traffic control, no congestion controlling is performed, which is drawn on "no resource control" curve. The EIED schemes with the maximum number of additional paths of 1 and 2 are shown on "1 additional path" and "2 additional paths" curves, respectively.
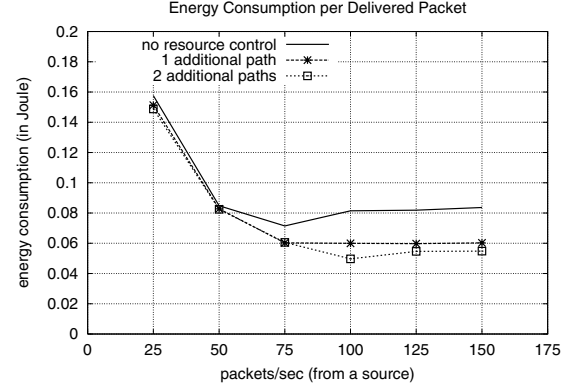
The results show as more additional paths are created, more packet are delivered to the sinks. A traffic control scheme will perform ideally when it tunes each source's reporting rate to 75 packets/second on "no resource control" curve, where its fidelity amount is the largest. However, the fidelity amount at the ideal traffic control point is still below the required fidelity amount. Compared with this ideal traffic control, a single additional path enhances the fidelity level by at most 22 % while two additional paths achieve at most 44 % fidelity enhancement. The curve with more additional paths accommodates more traffic, thereby alleviating congestion by approaching the "no congestion" curve if the reporting rate of a source is less than or equal to 100 packets/second. When the source's reporting rate exceeds 100 packets/second, "no congestion" curve also shows the saturated fidelity amount, which, however, is caused not by the congestion of the cross traffic but by the reporting rate that is too high.

6.3. Packet Energy Efficiency

Figure 4 shows the packet energy, i.e. the average energy consumed by the network for each delivered packet, under the same scenarios. It shows that even the packet energy consumption of the ideal traffic controlling (i.e. the "no resource control" curve when the reporting rate of a source is 75 packets/sec) is still higher than those of EIED schemes. These results show the additional energy consumed by the EIED policy offsets the energy wasted by packet drops due to congestion and resource controlling does not incur extra energy consumption as long as it reduces the packet drop rate. This result also shows that successfully delivering one packet in a dormant state (when the reporting rate a source is less than 50 packets/second in the figure) consumes significantly more energy than in a crisis state since nodes spend

most of energy idle-listening, which strongly manifests the need of deploying an energy-conserving scheme (for example, dynamically changing a node's duty cycle).

7. Conclusion

This paper for the first time attempts to formally define the resource control framework that adjusts the effective resource capacity rather than the incoming traffic at the hotspot to alleviate congestion. We describe several potential metrics and intuitions behind the resource control that are applicable especially to the sensor network where the availability of resource is elastic and a certain level of throughput called fidelity is required. In an effort to find the optimal resource control under the fidelity and energy constraints, we present a resource increase and decrease algorithm called EIED that achieves the highest fidelity level with the lowest energy overhead by controlling the effective channel capacity to the level of incoming traffic load. We also discover that the EIED algorithm performs better than the ideal traffic control scheme in terms of fidelity and energy efficiency that are the key metrics in sensor networks.

References

[1] A. Cerpa and D. Estrin. ASCENT: Adaptive Self-Configuring Sensor Networks Topologies. In *Proceedings of IEEE INFOCOM*, June 2002.

[2] D.-M. Chiu and R. Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN Systems 17(1):1-14*, 1989.

[3] C. T. Ee and R. Bajcsy. Congestion control and fairness for many-to-one routing in sensor networks. In *Proceedings of ACM SenSys*, Nov. 2004.

[4] S. Floyd. Connections with Multiple Congested Gateways in Packet-Switched Networks Part 1: One-way Traffic. *ACM Communication Review, 21(5):30-47*, Oct. 1991.

[5] S. Floyd and K. Fall. Promoting the Use of End-to-End Congestion Control in the Internet. *IEEE/ACM Transactions on Networking*, Aug. 1999.

[6] B. Hull, K. Jamieson, and H. Balakrishnan. Mitigating congestion in wireless sensor networks. In *Proceedings of ACM SenSys*, Nov. 2004.

[7] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM)*, August 2000.

[8] Y. Sankarasubramaniam, O. Akan, and I. Akyildiz. ESRT: Event-to-Sink Reliable Transport in Wireless Sensor Networks. In *proceedings of the ACM MobiHoc Conference*, 2003.

[9] C. Wan, S. Eisenman, and A. Campbell. CODA: Congestion Detection and Avoidance in Sensor Networks. In *proceedings of the ACM SenSys 2003*, 2003.

[10] A. Woo and D. Culler. A Transmission Control Scheme for Media Access in Sensor Networks. In *Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2001)*, July 2001.

[11] T. Yan, T. He, and J. A. Stankovic. Differentiated Surveillance Service for Sensor Networks. In *Proceedings of the ACM SenSys 2003*, 2003.

[12] F. Ye, G. Zhong, S. Lu, and L. Zhang. PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks. In *Proceedings of the 23rd International Conference on Distributed Computing Systems*, May 2003.

[13] Y. Yi and S. Shakkottai. A Hop-by-hop Congestion Control over a Wireless Multi-hop Network. In *Proceedings of IEEE INFOCOM'04*, March 2004.