A New Packet Scheduling Scheme for Improving Fairness in Wireless Sensor Network

Byunghun Song*, Hyung Su Lee*†, Hee Yong Youn†
*Korea Electronics Technology Institute, Korea
Email: bhsong@keti.re.kr, hslee@keti.re.kr
†School of Information and Communications Engineering
Sungkyunkwan University, Korea
Email: hsleel@skku.edu, youn@ece.skku.ac.kr

***Abstract*— In wireless sensor networks fair allocation of bandwidth among different nodes is one of the critical problems affecting the serviceability of the entire system. Fair bandwidth allocation mechanisms, like fair queuing, usually needs packet scheduling on a per flow basis, and the complexity may prevent them from being cost-effectively implemented and widely deployed. In this paper we propose a scheme that significantly reduces the implementation overhead yet still allows highly fair bandwidth allocation. This is achieved by letting the sensor node estimate the packet rate from each sensor device and attach a label at the packet header including the estimate. The sensor nodes relaying the packets maintain no per flow state; they use FIFO packet scheduling augmented by a probabilistic dropping algorithm that uses the packet label and an estimate of the aggregate traffic. The proposed scheme is evaluated and compared with earlier approaches using ns-2 simulator, and it reveals that the proposed scheme displays almost the same performance as the Deficit Round Robin(DRR) scheme, which has been recognized as the most effective one for achieving fairness but implementation overhead is quite high.**

I. INTRODUCTION

Wireless Sensor Networks (WSN) emerged as a separate field in ad hoc networking. The traffic in these networks is primarily sensor data [1]. Mesh networks represent a promising alternative for broadband Internet access [2]. Many other types of networks rely on multi-hop forwarding either to extend the range of an infrastructure-based wireless network as in the case of bridged access points or to avoid the need for infrastructure altogether as in the case of pure ad-hoc networks. Fairness is one of the most important properties of a wired and wireless network: when network resources are unable to satisfy the demand, they should be divided fairly between the clients of the network. Fair bandwidth allocation protects well-behaved flows from ill-behaved ones, and allows a diverse set of congestion control policies to co-exist in the network [3].

Most often, min-max fairness is the desired ideal fairness scheme. Under this scheme, the clients are split into two groups: the first group consists of the clients that cannot be completely satisfied by network resources, and they all receive the same share of bandwidth. The second group is made up of the clients that need less bandwidth than their fair share, and they receive exactly the amount of bandwidth that they ask for it. Until now, fair allocations were typically achieved by using per-flow queuing mechanisms such as Fair Queuing and its many variants or per-flow dropping mechanisms [4, 5]. These mechanisms are more complex to implement than traditional First-In First-Out (FIFO) queuing with drop-tail, which is the most widely implemented and deployed mechanism in sensor network today. In particular, fair allocation mechanisms inherently require the sensor node to maintain the state and perform operations on a per flow basis. For each incoming packet sensor node needs to classify the packet into a flow, update per flow state variables, and perform certain operations based on the per flow state. The operations can be as simple as deciding whether to drop or queue the packet or as complex as manipulation of priority queues. However, such mechanisms usually need to maintain the state, manage buffers, and perform packet scheduling on a per flow basis, and the complexity may prevent them from being cost-effectively implemented and widely deployed [6, 7].

In this paper we propose and examine an architecture and a set of algorithms that allocate bandwidth in an approximately fair manner while allowing the sensor nodes in WSN to use FIFO queuing and maintain no per-flow state. In our scheme each sensor node computes per-flow rate estimates and attaches a label at the packet header of the incoming packets showing the estimates. The sensor nodes employs FIFO queuing policy and keep no per-flow state. They also employ a probabilistic dropping algorithm that uses the information in the packet label along with their own measurement on the aggregate traffic. Bandwidth allocation within each sensor node is overally fair. Thus, if this approach is adopted within WSN, then approximately fair allocations could be achieved everywhere. The proposed scheme is evaluated and compared with earlier approaches using ns-2 simulator for a typical size sensor network of nine nodes. Simulation reveals that the proposed scheme displays almost the same performance as the Deficit Round Robin(DRR) scheme, which has been recognized as the most effective one for achieving fairness but implementation overhead is quite high. The proposed scheme also turns out to be slightly better than core stateless fair queueing (CSFQ) [10]. This shows that the proposed approach allows fairness much more cost-effectively than other approaches.

The rest of the paper is organized as follows: we first present the background and related work in Section 2. In Section

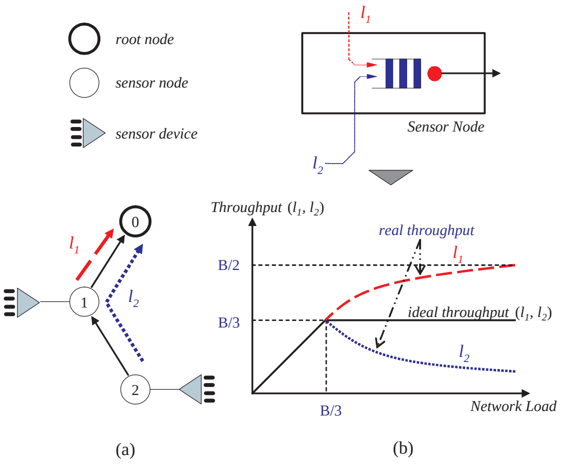
3, we describe the architecture of the proposed scheme. Section 4 presents the performance evaluation of the proposed architecture. Our conclusions are provided in Section 5.

II. BACKGROUND AND RELATED WORK

A. Unfairness Problem

A sensor node in a WSN has to transmit both relayed and its own traffic. Therefore, besides the contention with other nodes for the same destination node, there is an inevitable contention between its own and relayed traffic. This contention does not occur in fixed wireless local loops or wireless LANs in infrastructure mode where user nodes are always at one-hop distance from the base station or the access point [8].

Consider the simple case depicted in Fig. 1(a) where two sensor nodes are having same offered load to be sent to the root node. Sensor node 1 and sensor node 2 received packets from a sensor devices at the rate of r1 and r2, respectively. Ideally, as the offered load at each of the nodes increases, both nodes will receive the same share of the MAC layer throughput, *B*. In practice, with FIFO queuing in WSN today, as the offered load increases, the sensor node closest to the root node will gradually but completely starve the sensor node further away from the root node. The ideal throughput in Fig. 1(b) are obtained under the assumption that bandwidth allocation is fair share between the traffic to be relayed by sensor node 1 (from sensor node 2 to the root node) and the traffic originating at sensor node 1. As can be seen in Fig. 1(b), as the load at both the nodes is increased, sensor node 2 is gradually but eventually completely starved by sensor node 1.



(a) (b)

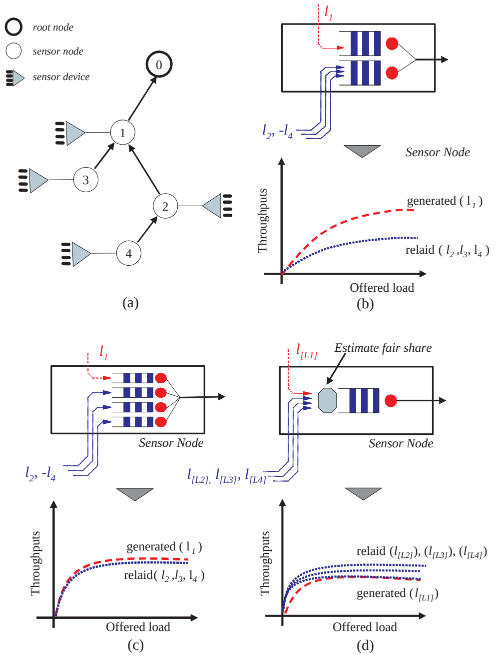
Fig. 1. (a) Simple WSN topology. (b) Throughput of sensor node 1 and 2.

The overall system throughput would be $\frac{B}{2}$ when the offered load is very high, where B is the throughput of the system when only sensor node 1 forwards data to the root node. So, the system operates at 50% efficiency and is unfair. This unfair behavior observed in Fig. 1(b) is rooted from the fact that both the traffic, originating at sensor node 1 as well as the relayed traffic originating from sensor node 2, are queued together at sensor node 1. When the traffic load increases, the network cannot forward all data enqueued at sensor node 1, and the queue starts to overflow. With a probability increasing as the offered load grows, the queue will be full when a new packet arrives from sensor node 1, and it will be dropped immediately. The expected throughput was determined theoretically, and also verified using ns-2.

B. Solutions for Unfairness Problem

To resolve the problem described above, we need to explain the advantages and disadvantages of the following solutions proposed for it. For clarity, a simple wireless sensor network of four nodes as shown in Fig. 2(a) is used. We assume that all the traffic flows are unidirectional toward the root node. Each node in Fig. 2(a) can be modeled as a WSN with queues as shown in Fig. 2(b), (c), and (d).

With the FIFO queuing policy employed in most WSN, each sensor node can be modeled using a queue. Only one FIFO queue can accommodate both the originating and relayed traffic flow. Observe from Fig. 2(a) that sensor node 1 is closest to the root node, which needs to relay the packets originating from node 2, 3, and 4. We denote the traffic flow originating from node 1 as l_1 and the relayed traffic flows as l_2, l_3, and l_4, respectively. With the FIFO queuing discipline, it is clear that the traffic flow l_1 will receive more bandwidth and eventually starve others. This problem stems from the fact that both the relayed and originating traffic share a single FIFO queue in each node. The approaches proposed for solving this problem are as follows.



(a) (b) (c) (d)

Fig. 2. (a) Multihop WSN topology. (b) Isolate queuing. (c) Per-flow queuing. (d) State-elimination queuing

i) Isolate Queuing: To alleviate the unfairness problem the first solution that comes to ones mind is to use different queues for the relayed and the originating traffic, and serve them in a round-robin fashion. This scheme is shown in Fig. 2(b), which isolates the originating traffic in order to protect the relayed traffic from being starved by the originating one. Isolating the originating traffic by putting two fair queues at the network layer still shows significant unfairness of the per-flow throughputs, although the scheme is simple to implement and prevents the severe starvation of relayed traffic. Fairness is guaranteed with this scheme only when the length of the chain of the nodes does not exceed two hops [9].

ii) Per-Flow Queuing: To resolve the unfairness problem, the per-flow queuing is an ideal scheme. As shown in Fig. 2(c), packets of different flows are enqueued separately based on their source. Here we consider only unidirectional flows towards the root node. The Deficit Round Robin (DRR) represents an efficient implementation of the well-known fair queuing discipline. The buffer management scheme assumes that when the buffer is full, the packet from the longest queue is dropped. DRR uses a sophisticated per-flow queuing algorithm, and thus achieves the highest degree of fairness [9].

iii) State-Elimination Queuing: The DRR have the problem of scalability. This problem led to the proposal of stateless core (SCORE) networks and core stateless fair queuing (CSFQ) as shown in Fig. 2(d). The SCORE network architecture differentiates edge router from core router in Internet. Edge and core routers estimate the fair share rate and probabilistically drop packets whose rate value exceeds the fair share rate. Since core routers use the state information contained in the packet header instead of maintaining per-flow state, CSFQ significantly reduces the implement complexity and unfairness problem as compared to per-flow queuing techniques [10].

C. *Problem of Per-flow Queuing in WSN*

Unfortunately, per-flow queuing mechanisms usually needs to maintain the state, manages the buffers, and performs packet scheduling on a per flow basis. The complexity may prevent them from being cost-effectively implemented and widely deployed. Therefore, per-flow queuing seems not the choice for WSN. We expect that the state-elimination queuing policy becomes the preferred approach for fairness improvement in WSN. Its merit might be small for wired Internet but substantial for WSN. We next present the proposed scheme based on state-elimination queueing policy but requiring much simpler implementation overhead.

III. THE PROPOSED SCHEME

We define the network to be fair if, over a period of time, the number of packets received from each node in the network is approximately the same. We do not attempt to ensure that the latencies experienced by all packets to be about equal.

In this section we propose a packet scheduling approach for WSN to improve the fairness. It approximates the service provided by the sensor node with new packet scheduling, but has a much lower complexity in packet forwarding.

The proposed scheme has two key aspects. First, to avoid maintaining per flow state at every node, we use a distributed algorithm in which only the sensor node generating packets maintains the per-flow state[1]. On the contrary, the node relaying packets does not maintain per flow state but instead utilizes the per-flow information carried via a label in each packet header. This label contains an estimate of the rate of the sensed data; it is initialized by the node based on per-flow information, and then updated at each sensor node along the path based only on aggregate information. Second, to avoid per flow buffering and scheduling, as required by fair queuing, we use FIFO queuing with probabilistic dropping on input. The probability of packet dropping is a function of the rate estimate carried in the label and the fair share rate at each sensor node, which is estimated based on the measurement on the aggregate traffic. Thus, our approach avoids both the need for maintaining per-flow state and using complicated packet scheduling and buffering algorithms at the packet relaying node.

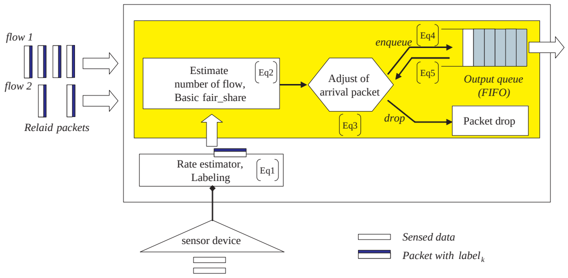


Fig. 3. The architecture of proposed scheme

A. *Algorithms*

The proposed scheme employs a drop-on-input policy. Because the arrival rate estimation of output queue incorporates the packet size, the dropping probability is independent of the packet size and depends only, as explained above, on the average arrival rate of output queue $r_i(t)$ and fair share rate β_i(t). We are left with two remaining challenges: estimating the average arrival rates $r_i(t)$ and the fair share rate β_i(t). We address these two issues in Equation (4), (2) and (5). We should note, however, that the main point of our paper is the overall architecture and that the detailed algorithm presented below presents only a possible approach. While it serves adequately as a proof-of-concept of our architecture, we fully expect that the details of the design will continue to evolve. To our intuition, the arrival rates are known exactly and the system adopts the probabilistic dropping algorithm applied according to Equation (3). Through the piggy-backing of node id in the packet headers, we can estimate the number of flow. C is link capacity. $label_i^j$ is the value of label of the j-th packet in the i-th flow. l_i^k is

---

[1] The packet generating node may also relay the packets arriving for other nodes

the length of packet of the k-th packet in the i-th flow. T_i^k is the time interval between k-th packet and (k-1)-th packet in the i-th flow. *label*$_k$ is the value of label of the present packet. $l_{out,k}$ is the length of the k-th packet. $T_{out,k}$ is the time interval between the k-th packet and the (k-1)-th packet.

$$label_i^j = \frac{\sum_{k=1}^{j} l_i^k}{\sum_{k=1}^{j} T_i^k} \tag{1}$$

$$\beta_i(t) = \frac{C}{number\_of\_flow_i} \tag{2}$$

$$if \begin{cases} \beta_i(t) \geq label_k \text{ , Enqueue} \\ \beta_i(t) < label_k \text{ , Drop Probability is } \left(1 - \frac{\beta_i(t)}{label_k}\right) \end{cases} \tag{3}$$

$$r_i(t) = \frac{\sum_{k=1}^{j} l_{out,k}}{\sum_{k=1}^{j} T_{out,k}} \tag{4}$$

$$if \begin{cases} r_i(t) \geq C,\ \beta_i(t) = \beta_i(t) \times \left(1 - \frac{1-(C/r_i(t))}{number\_of\_flow_j}\right) \\ r_i(t) < C,\ \beta_i(t) = \beta_i(t) \times \left(1 + \frac{(C/r_i(t))-1}{number\_of\_flow_j}\right) \end{cases} \tag{5}$$

IV. PERFORMANCE EVALUATION

In this section we evaluate and compare the performance of the proposed scheme with that of DRR, FIFO, and CSFQ in WSN using ns-2 simulator. Unless explicitly stated, the same amount of traffic is used for all the experimental runs. We evaluate the schemes in terms of throughput and fairness.

A. Simulation Setup

We implemented scheme for improving fairness in WSN nodes, and evaluated the relative throughput and fairness of proposed scheme versus DRR, CSFQ and FIFO. Fig. 4 shows the simulation setup used for the evaluation and comparison. Various bottleneck capacity and delays are tested. The buffer space at the bottleneck router is set to 100% of the bandwidth and delay product of the bottleneck link. Every traffic passes through the bottleneck link; each link is configured to have different RTTs and different starting time and end time to reduce the phase effect. Fig. 4 also shows the NS-2 simulation setup [11].

The transport of event impulses is likely to lead to varying degrees of congestion in sensor networks. In order to illustrate the fairness problem by congestion consider the following simple but realistic simulation scenario.

The network topology for simulation consists of 10 nodes. All nodes are within range of one another, this results in significant interference between them. The exponential moving average weight used to obtain the average transmission rate is set to 2, and each node is initialized with a data generation rate of one packets per second. Also, queue size in each sensor node is 10 packets.

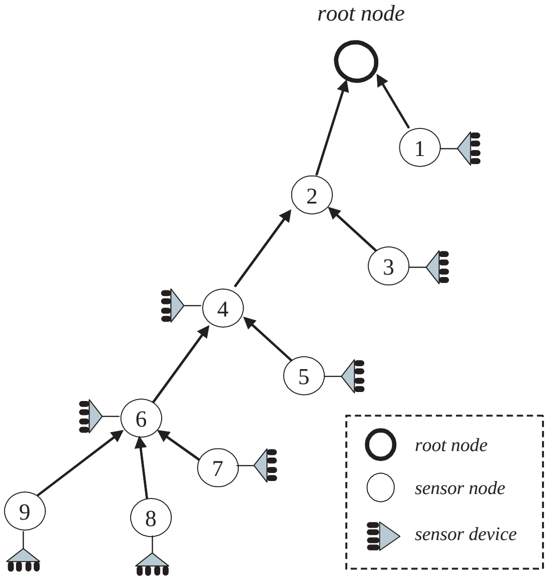


Fig. 4. The network topology employed for simulation

B. Throughput and Fairness

We now analyze how the throughput of a well-behaved flow is affected when the flow traverses several sensor nodes. Recall that DRR is the most effective scheme for achieving fairness in packet scheduling even though the implementation overhead is quite high. Therefore, we select it as a reference to compare the relative performance of the schemes studied here.

Fig. 5 shows the fraction of sensing data that is forwarded versus the number of sensor nodes traversed, which is normalized by the bandwidth of DRR. Note that the proposed scheme is quite close to DRR, and the normalized bandwidth is almost flat regardless of the number of traversed nodes guranteeing the fairness. The normalized bandwidth is according to Equation 3.

$$Normalized\ bandwidth = \frac{Real\ bandwidth}{Ideal\ bandwidth} \tag{6}$$

In particular with FIFO queuing, the sensor node close to the root node gradually but completely starves the sensor node far away from the root node as the offered load increases. CSFQ also performs good. The reason why the proposed scheme and the two schemes show high fairness is that they try to allocate bandwidth fairly among competing flows during congestion in WSN.

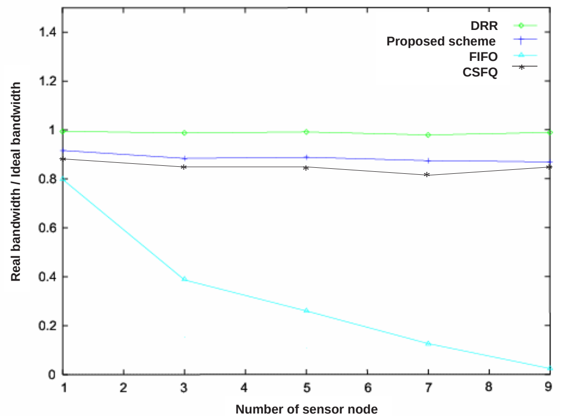


Fig. 5. Normalized bandwidth versus number of sensor nodes

V. CONCLUSION

In this paper we have shown a significant fairness problem existing in practical WSN. We have thus proposed a new packet scheduling scheme for WSN to solve the fairness problem. The improvement in fairness is directly related to the resource investment.

We say that fairness is achieved when equal number of packets are received from each node. Since in general we have many sensors transmitting data to the root node, we consider the scenario where we have many-to-one data forwarding, noting that it can easily be extended to unicast. Such forwarding structures often result in the sensors closer to the root node experiencing congestion, which inevitably cause packets originating from sensors further away from the root node to have a higher probability of being dropped.

We proposed a scheduling scheme and algorithm that allocate bandwidth in an approximately fair manner while allowing the sensor nodes on WSN to use FIFO queuing and letting only the nodes generating the packets maintain per-flow state. The per-flow rate estimate is labeled at the packet header as the packet passes through the nodes. Intermediate sensor nodes of next-hop use FIFO queuing and keep no per-flow state. They employ a probabilistic dropping algorithm that uses the information in the packet label along with their own measurement on the aggregate traffic. The bandwidth allocations within sensor nodes are approximately fair. Finally, we generalized the fairness concept to enable differentiated services in WSN. Our scheme exists in the network layer of the traditional network stack model, and is designed to work with any MAC protocol in the data-link layer with minor modifications.

We will further investigate the mechanism by which the overall throughput as well as fairness can be maximized for large-scale WSNs.

REFERENCES

[1] P. Gupta and P. Kumar, "The Capacity of Wireless Networks," *Proceeding of IEEE Transaction on Information Theory*, Mar. 2000.
[2] S. Capkun, M. Hamdi, and J. Hubaux, "GPS-Free Positioning in Mobile Ad-hoc Networks," *Proceeding of 4th Annual Hawaii International Conference on System Sciences*, Jan. 2001.
[3] B. Schrick and M. Riezenman, "Wireless Broadband in a Box," *Proceeding of IEEE Spectrum Magazine*, June 2002.
[4] M. Subbarao, "Dynamic Power-Conscious Routing for MANETS: An Initial Approach," *Proceeding of IEEE Vehicular Technology Conference*, May 1999.
[5] J. Royer and C. Toh, "A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks," *Proceeding of IEEE Personal Communications*, Apr. 1999.
[6] M. Gerla, K. Tang, and R. Bagrodia, "TCP Performance in Wireless Multihop Networks," *Proceeding of IEEE WMCSA*, Feb. 1999.
[7] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *Proceeding of IEEE Communication Magazine*, Aug. 2002.
[8] L. Kalampoukas, A. Varma, and K. Ramakrishnan, "An Efficient Rate Allocation Algorithm for ATM Networks Providing Min-Max Fairness," UCSC-CRL-95-29 Computer Engineering Dept. University of California, Tech. Rep., June 1995.
[9] Z. Cao, Z. Wang, and E. Zegura. "Rainbow Fair Queueing: Fair Bandwidth Sharing without Per-flow State," *Proceedings of INFOCOM*, March 2000.
[10] I. Stoica, S. Shenker, H. Zhang, "Core-Stateless Fair Queueing: A Scalable Architecture to Approximate Fair Bandwidth Allocations in High Speed Networks," textitProceedings of ACM SIGCOMM, March 1998.
[11] UCB LBNL VINT, "Network Simulator ns (Version 2)", http://www.isi.edu/nsnam/ns/.