

Hybrid Congestion Control Protocol in Wireless Sensor Networks*

JANG-PING SHEU^{1,2}, LI-JEN CHANG² AND WEI-KAI HU²

¹*Department of Computer Science
National Tsing Hua University
Hsinchu, 300 Taiwan*

²*Department of Computer Science and Information Engineering
National Central University
Chungli, 320 Taiwan*

In wireless sensor networks, congestion occurs when the traffic load being offered exceeds the available capacity of sensor nodes. In most applications, every sensor node will send the event it has sensed to a sink node. This operation makes the sensors closer to the sink, resulting in congestion. Congestion may cause packets loss, lower network throughput and sensor energy waste. To address this challenge, there is a proposal for a distributed algorithm that mitigates congestion and allocates appropriate source rate to a sink node for sensor networks. The proposed algorithm is a hybrid congestion control protocol that considers not only the packets delivery rate but also retains the buffer size of each node. The proposed protocol may avoid packets drop due to traffic congestion and improve the network throughput. The simulation results show that the performance of the proposed protocol is better than the previous works.

Keywords: buffer management, congestion control, flow control, wireless sensor networks, rate adjustment

1. INTRODUCTION

A wireless sensor network (WSN) is a network composed of distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions – such as temperature, sound, vibration, pressure, motion or pollutants – at different locations. WSNs have a wide range of applications in object tracking [1], object localization [2], habitat observation [3, 4], health monitoring [5], and battlefield sensing among others.

WSNs are different from traditional wireless networks in many aspects. For example, the forwarding rate of a sensor is decided not only by the channel capacity but also by the lifetime of the neighbor sensors due to a limited energy supply. A wireless sensor network is also constrained by memory space, computation capacity, communication bandwidth, and energy supply. Consequently, the capacity of sensors is limited. Therefore, a lot of academic research topics are discussing how to prolong the whole network lifetime. For example, the power control issue [6] and energy-aware routing [7] adapt the energy transmission range to save power. The load balance issues [8-11] average the work load of each node. Moreover, the congestion control protocols [12-19] can mitigate network bottlenecks and improve the network performance.

Received September 4, 2007; revised April 9, 2008; accepted May 2, 2008.

Communicated by Ten-Hwang Lai.

* This paper was partially supported by the National Science Council of Taiwan, R.O.C. under grant No. NSC 96-2221-E-007-174.

Studies have been trying to address congestion because it causes a lot of problems. The energy spent by upstream neighbors on a packet is wasted when the packet is dropped. When congestions happen without any control protocol being implemented, more packets will be dropped and more energy is wasted. The congestion occurring in a node may result in a quick decline of a network throughput. Consider the sensor network that typically operates under a light load but may suddenly be active in response to certain importance events such as fire blaze, earthquake, mudflows, and landslides. The sudden surge of data from hundreds or even thousands of sensors must be delivered to a small number of sinks, which may cause congestion, especially nodes near the sinks.

Congestion control has been an active area of networking research for several decades, but relatively less attention has been paid to congestion control in the emerging aspect of wireless sensor networks. To address this challenge, we must solve the flow control and fairness problems. The flow control seeks to manage the data rate from upstream neighbors once congestion happens. The fairness problem aims to ensure that the nodes have equal or weighted probability to share the network bandwidth [15].

The WSNs are not like traditional wired networks. In wired networks, there is a TCP congestion control to avoid congestion. The TCP congestion control consists of four intertwined algorithms: slow start, congestion avoidance, fast retransmission, and fast recovery. The wireless sensor networks cannot use TCP congestion control effectively because the sensor is constrained by memory space, computation capacity, communication bandwidth, and energy. Besides, TCP inevitably will lead to packet loss and waste energy resource. Under wireless environments, TCP's congestion detection mechanism based on duplicated ACK will also lead to low throughput.

There are various congestion control schemes [12-19] in sensor networks. The congestion control protocols can be classified into three categories: rate-based, buffer-based and priority-based schemes. The rate-based scheme is to utilize a forwarding node that calculates the flow rate of upstream and downstream to determine the forwarding rate. The key of the buffer-based scheme is to make sure that a sensor i will send a packet to its downstream neighbor k only when k has buffer space to hold the packet. The priority-based scheme emphasizes the importance of priority, which means that the important sensors have higher priority and can gain higher throughput than normal ones.

In this paper, a Hybrid Congestion Control Protocol (HCCP), considering both the packets delivery rate and remaining buffer size of each node is proposed. The scheme does not need to maintain the global flow information and each node makes use of its current remaining buffer size and net flow size to calculate its congestion degree information. The congestion degree is defined to reflect the current congestion level at each node. Then, the congestion degree is exchanged periodically between neighbors. As such, each node can use its congestion degree and neighbors' congestion degrees to prevent the emergence of congestion. The simulations show that the proposed protocol can reduce packets drop rate and increase packets delivery ratio effectively.

2. RELATED WORK

Congestion control generally follows three steps: congestion detection, congestion advertisement, and transmission rate adjustment. In this section, the various congestion

control schemes [12-19] in WSNs are described. They may be classified into priority-based, rate-based, and buffer-based schemes.

2.1 Priority-Based Scheme

In the priority-based scheme, the emphasis rests mainly on priority. Most of the recent studies about congestion control for WSNs only guarantee simply fairness, which means that the sink receives the same throughput from all sensors. In fact, sensors may be deployed in different places and they come with different hardware and capacity. They also have different sensing events. Therefore, the priorities of sensors may differ.

The important sensors have higher priority, which means the important sensors can gain higher throughput. In [17] a priority-based congestion control protocol is proposed. This scheme uses packet inter-arrival time along with packet service time to measure a parameter defined as congestion degree and imposes hop-by-hop control based on the measured congestion degree as well as the node priority index. The packet inter-arrival time (t_a^i) is defined as the time interval between two sequential arriving packets and the packet service time (t_s^i) is referred to as the time interval between when a packet arrives at the MAC layer and when its last bit is successfully transmitted. Based on the average inter-arrival time and average service time, it defines a new congestion index: congestion degree $d(i) = t_s^i / t_a^i$. If $d(i)$ is larger than 1, the node experiences congestion. Each sensor node i piggybacks congestion degree in the header of data packets to be forwarded. The notification is triggered when the node overhears a congestion notification from its parent node in a period of time. Finally, each node allocates data rate to its upstream nodes based on its priorities.

2.2 Rate-Based Scheme

The basic idea of the rate-based scheme is for a forwarding node to estimate the number of flows coming from each upstream neighbor and assign transmission rate based on fairness once congestion is detected. In [12] an event-to-sink reliable transport protocol (ESRT) is proposed for congestion control. ESRT is a centralized protocol that regulates the reporting rate of sensors in response to congestion detected by a sink. Each sensor node monitors its local buffer level and sets a congestion notification bit in the packets forwarded to the sink if the buffers overflow. When the sink receives a packet with the congestion notification bit set, it infers congestion and broadcasts a control signal notifying all source nodes to reduce their reporting frequency.

A distributed congestion detection and avoidance protocol (CODA) is proposed in [13]. In CODA, once congestion is detected, the receiver broadcasts a suppressive message to its upstream neighbors and at the same time make local adjustment to prevent propagating the congestion downstream. When an upstream node receives a backpressure message, based on its own local network conditions it determines whether to further propagate the backpressure signal or not. This scheme does not consider the fairness issue. The authors in [14] propose a mitigating congestion protocol which combines three congestion mitigating mechanisms: hop-by-hop flow control, rate limiting and prioritized MAC layer. This scheme requires a tree routing structure to work correctly.

A localized algorithm for aggregate fairness protocol is proposed in [15]. When a

sensor receives more packets than it can forward, the sensor will calculate and allocate the data rates of upstream neighbors by a weighted fairness function. However, the fairness function of this congestion control protocol was not considered carefully with the remaining buffer size and transmission rate at the same time. It only considers that the sum of data rate from upstream neighbors must be less than the sum of data rate it can forward to downstream neighbors when congestion is detected.

2.3 Buffer-Based Scheme

In the buffer-based scheme, the key for congestion control is to make sure that a sensor i sends a packet to its downstream neighbor j only when j has buffer space to hold the packet. This scheme is simple and effective, but it does not consider the data rate of the upstream and downstream neighbors. It eliminates the complicated rate-based signaling that is required by many existing congestion control approaches. This scheme, unlike the rate-based approaches, does not drop packets.

Let N_i be the set of the neighbors of node i . The remaining buffer size of i changes when it receives a packet from upstream neighbors or forward a packet to a downstream neighbor. When node i sends out a packet, it piggybacks its current buffer state in the frame header of the package. Consider a neighbor sensor $j \in N_i$. When j receives or overhears a packet from i , it caches the buffer state of i . The sensor j has a packet to forward i , only if the buffer of i is not full. If the buffer of i is full, j withholds the packet until it overhears a packet from i , piggybacking a non-full buffer state. In [16] a congestion avoidance protocol based on lightweight buffer management in sensor networks is proposed. Although it can realize and guarantee that the packet does not drop in the forwarding way, the buffer utilization is low.

Most congestion control protocols do not consider buffer state and data rate at the same time. The rate-based scheme is for a forwarding node to estimate the number of flows coming from each upstream neighbor and assign rate in accordance with the fairness once the congestion is detected. The buffer-based scheme makes sure that a sensor i sends a packet to its downstream neighbor j only when j has buffer space to hold the packet. Both of the two schemes consider either in data rate or buffer state to allocate the data rate for its upstream neighbors.

Here, we show the drawback of rate-based scheme and buffer-based scheme. We assume that the packet length is fixed. The unit of buffer size is packet and the data rate is number of packets per unit time. In Fig. 1 (a), assume that the average data rate $R_{a,d}$, $R_{b,d}$ and $R_{d,c}$ in current transmission period are 9, 6 and 5, respectively and the remaining buffer size of d is 9. In Fig. 1 (b), assume that the average data rate $R_{a,d}$, $R_{b,d}$ and $R_{d,c}$ are 6, 3, and 8, respectively and the remaining buffer size of d is 3. Considering the buffer-based scheme, the congestion degree of d in Fig. 1 (b) is higher than that of in Fig. 1 (a). However, considering the impact of data rate, the congestion degree of d in Fig. 1 (b) is smaller than that of in Fig. 1 (a). This is because the net flow size of d is 10 in Fig. 1 (a) and congestion will happen in the next time period but the net flow size of d in Fig. 1 (b) is only one.

On the other hand, in Fig. 1 (c), assume that the remaining buffer size of d is 2 and the average data rate $R_{a,d}$, $R_{b,d}$ and $R_{d,c}$ in current transmission period are 4, 5, and 6, respectively. In Fig. 1 (d), assume that the remaining buffer size of d is 12 and the average

data rate $R_{a,d}$, $R_{b,d}$ and $R_{d,c}$ are 5, 7, and 7, respectively. If we consider the impact of the data rate, the congestion degree of d in Fig. 1 (d) is higher than that of in Fig. 1 (c). However, if we consider the remaining buffer size, the congestion degree of d in Fig. 1 (d) is less than that of in Fig. 1 (c). This is because the remaining buffer size of d is two in Fig. 1 (c) and congestion will happen in the next time period. To avoid the above drawback of rate-based and buffer-based schemes, our congestion control protocol takes both of the buffer capacity and the data rate into considerations. So, it is a kind of “hybrid” protocol.

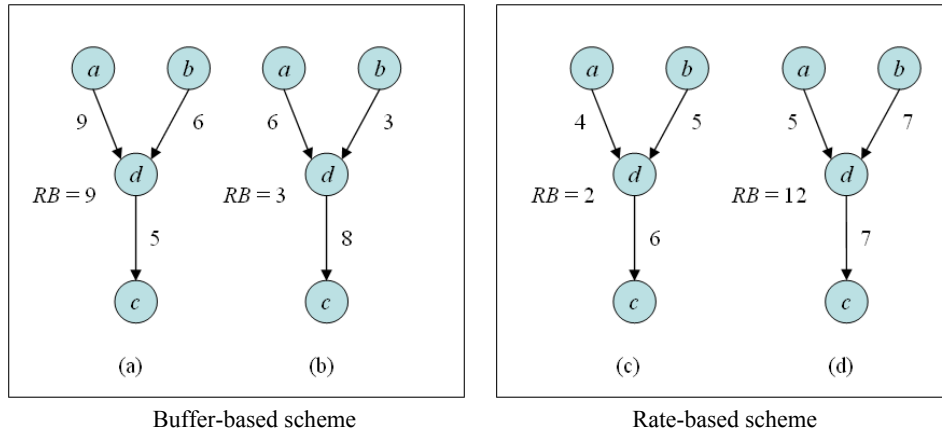
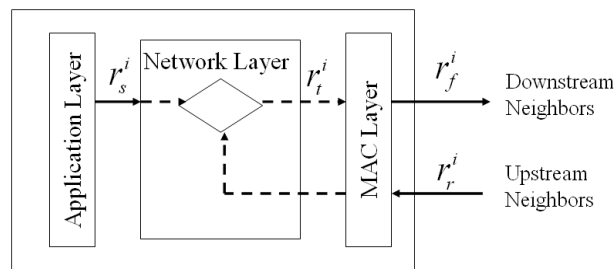


Fig. 1. Examples of buffer-based and rate-based congestion control protocols.

3. HYBRID CONGESTION CONTROL PROTOCOL (HCCP)

There are two types of congestion in WSNs: Channel Collision and Buffer Congestion. The first type of channel collision can be solved by CSMA, FDMA, TDMA, and CDMA, in MAC layer [16]. A growing number of sensor networks use CSMA for medium access. For example, the widely used Berkeley motes use a simple CSMA MAC as part of the TinyOS platform. The CSMA can improve channel collision but cannot solve the congestion problem. It may cause the buffer of a sensor overflow if several neighbors of the sensor have packets with high data rate to the sensor.

The proposed protocol exists between the network layer and MAC layer. The node model of each sensor is shown in Fig. 2. The relay traffic rate of node i (r_r^i) is received from its upstream neighbors through the MAC layer of i . The source traffic rate of node i (r_s^i) is generated by node i . The total data rate (r_t^i) of i through the network layer to MAC layer are converged both r_r^i and r_s^i . So that, the $r_t^i = r_s^i + r_r^i$. The forward data rate (r_f^i) is the total rate that all of its downstream neighbors allow it to pass. A packet could be queued at buffer at network layer when the forward rate r_f^i is smaller than the total rate r_t^i . If r_t^i is continuously bigger than r_f^i , the buffers will fill up quickly. Finally, the buffers will overflow, and the congestion will take place. In order to avoid the congestion, we can reduce the r_s^i , r_r^i or both.

Fig. 2. Packets flow model of sensor i .

In this paper, we study the problem of data gathering for a sensor network, from which all source nodes send packets to the sink. Every sensor node sends message from many-to-one convergent traffic to the sink when an event is sensed. Because all sensors are data sources, all of them may relay data from their upstream neighbors toward the sink. Each sensor cannot move after deployment. The sensors have one or more parents and may have many children and grandchildren nodes or not. Our protocol can be implemented to the existence MAC protocol and routing protocol. Some of congestion control protocols may only be suitable for single-path or have the restriction of tree-based routing. For instance, the algorithm proposed in [17] is used for tree-base routing protocol. The scheme in [14] assumes a single-path routing structure, which can allocate bandwidth fairly among different data sources. The centralized protocol proposed in [12] is used for single-path or tree-based routing. Our HCCP is a distributed protocol that mitigates congestion. Each node in the network individually allocates or reduces the data rate of upstream or downstream neighbors to avoid the congestion. No matter what kind of routing protocol is used in the network layer, each node just locally negotiates with its upstream and downstream neighbors about the data rate. So our congestion control protocol can work well with any routing protocol.

We assume each link is symmetric. Each sensor node has two type neighbor nodes: one is a group of upstream neighbors and another is a group of downstream ones. Let U_i be the set of upstream neighbors of node i , which pass through i and forward to the sink. Let D_i be the set of downstream neighbors of node i , which are the next hop on the routing path from i to the sink. We assume that each sensor node has a counter that can calculate the data rate from upstream neighbors and recode the data rate to downstream neighbors. An upstream neighbor must be its parent and a downstream neighbor must be its children.

The remaining buffer size of i is represented by RB_i and the net flow size of i is represented by NS_i . Assume that the packet length is fixed. Each sensor node i has a congestion degree CD_i , which is the index of congestion level. According to the congestion degree, we can classify the current traffic load of each node into light-load state and heavy-load state. Sensor nodes have a neighbor table to record the congestion degree and traffic information of neighbors.

In the following, we present our Hybrid Congestion Control Protocol (HCCP), which mitigates congestion and allocates appropriate source rate to the sink node for sensor networks. HCCP comprises two phases: congestion detection phase and data rate adjustment phase. HCCP does not maintain the global flow information. Each node

makes use of its current remaining buffer size and net flow size to calculate its congestion degree. And the congestion degree is exchanged periodically between neighbors. Therefore, each node can use its congestion degree and its neighbors' congestion degree to prevent congestion.

3.1 Congestion Detection Phase

Since each sensor may have one or more upstream and downstream neighbors, there exist many input flows from upstream neighbors and output flows to downstream neighbors. Congestion probably occurs when the flows cross each other complicatedly. Prevention is better than cure. In addition, prevention can prevent packet loss and improve energy-efficiency. When a sensor node detects congestion, it may cost a lot of time and network bandwidth to solve the congestion problem. Thus, we would like to detect the congestion in advance and take the preventive measures. In this phase, the congestion degree is predicted based on a time period T . Each sensor will count the current upstream and downstream data rates of its neighbors and predict whether congestion will happen or not in the next time period T . The time period T can be neither too long nor too short. If T is too short, it will cause high control overhead due to the frequently congestion detection. On the other hand, if T is too long, the congestion will happen before time is expired. It will cause low performance of congestion control.

For a sensor i , if the flows rate coming from the upstream neighbors is far greater than the flows rate it can forward to downstream neighbors, and the buffer of sensor i cannot hold the net flow size in the next time period, it will suppress the upstream neighbors to slow down their data rate and the buffer state of sensor i is set as heavy. If the remaining buffer size of sensor i in the next time period is greater than or equal to the total flows size coming from the upstream neighbors minus the total flows size forwarding to downstream neighbors, we define the buffer state of sensor i is in light-load state.

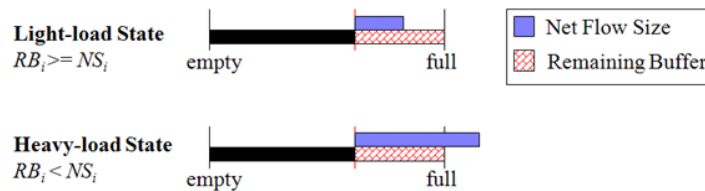


Fig. 3. The example of light-load state and heavy-load state buffers.

In Fig. 3, we divide the buffer state into light-load state and heavy-load state. When the buffer enters a light-load state, it guarantees that congestion will not happen in the next time period, we do not need to perform any congestion control. Otherwise, when the buffer state enters heavy-load state the congestion may happen in the next time period, we must trigger the congestion control process to avoid the congestion and assign the proper data rate for its neighbors according to their congestion degrees.

In order to avoid the buffer overflow, a sensor i must estimate the net flow size from all neighbors within a time period. Let $R_{i,j} \{\forall i \in N, j \in D_i\}$ be the average downstream data rate from node i to j per unit time. Let $R_{k,i} \{\forall i \in N, k \in U_i\}$ be the average upstream

data rate from node k to i per unit time. The $R_{i,j}$ and $R_{k,i}$ can be easily measured at each sensor i by a counter on a packet-by-packet basis. Then a net flow size NS_i is the source traffic rate of sensor i (r_s^i) plus all flows from upstream neighbors of sensor i and minus all flows that sensor i can forward to downstream neighbors during a time period T as follows:

$$NS_i = (r_s^i + \sum_{j \in U_i} R_{j,i} - \sum_{k \in D_i} R_{i,k}) \times T, \forall i, j, k \in N. \quad (1)$$

In order to indicate the index of congestion, we define a congestion degree CD_i , which is the remaining buffer size minus the net flow size of each sensor i during a time period T as follows:

$$CD_i = RB_i - NS_i. \quad (2)$$

If the CD_i is smaller than 0, the buffer state of i will become heavy and congestion may happen in the next time period. The sensor i will broadcast a suppressive message to advertise its neighbors to slow down their data rates. For sensors to know the congestion degrees of their neighboring nodes, they will advertise their congestion degrees to each other. For each sensor, the advertisement is triggered by either of the following two events: (1) in the beginning of each time period T and (2) the buffer state from light-load to heavy-load. In order to reduce the control message overhead, if a sensor has the data traffic, we piggyback the congestion degree in the header of the data packet.

3.2 Data Rate Adjustment Phase

Assume that the sensors will forward the data packets to the downstream neighbors as fast as possible. When sensor i obtains the congestion degrees of its upstream and downstream neighbors, it will calculate the value of r_i^j and r_j^i , and updates its congestion degree. Once the CD_i of sensor i is larger than or equal to 0, it means that the buffer state of i is light, and therefore, it will do nothing. On the other hand, if the CD_i of sensor i is smaller than 0, it will suppress the data rate of upstream neighbors of i . In order to allocate effectively data rates to upstream neighbors, the upstream neighbors that tend to congest will be allocated more data rate. Sensor i can estimate each upstream neighbor's tendency towards congestion by CD_x and $R_{x,i}$. We define a tendency congestion degree $\alpha_i(x)$ represents the degree of congestion probability of x if the total traffic from x to i is prohibited in a time period T . Then we have

$$\alpha_i(x) = CD_x - R_{x,i} \times T, \forall x \in U_i. \quad (3)$$

If $\alpha_i(x)$ less than 0, it represents if sensor i suppress the data rate $R_{x,i}$, sensor x may congestion in the next time period. Otherwise, if $\alpha_i(x)$ is larger than or equal to 0, it represents congestion will not happen in the next time period even if sensor i suppresses the data rate $R_{x,i}$. Therefore, the more negative the value of $\alpha_i(x)$ is, the more data rate will be allocated to x by sensor i . Let SUM_i be the summation of absolute value of $\alpha_i(x) < 0$.

$$SUM_i = \sum_{x \in U_i} |\alpha_i(x)|, \forall \alpha_i(x) < 0 \quad (4)$$

Here, we define a potential traffic capacity PC_i , which is the remaining buffer size plus the sum of flows size that sensor i can forward to its downstream neighbors during a time period T as follows:

$$PC_i = RB_i + \sum_{k \in D_i} R_{i,k} \times T. \quad (5)$$

The PC_i represents how many packets that the sensor i can hold from upstream neighbors in the next time period. The sensor i will calculate whether its PC_i is enough to satisfy those sensors with $\alpha_i(x) < 0$. Let $PC'_i = PC_i - SUM_i$ be the remaining potential traffic capacity of i . If $PC'_i \geq 0$, it means that the potential traffic capacity of i can satisfy the requirement of upstream neighbor nodes with $\alpha_i(x) < 0$. Sensor i will first consider to allocate data rate to the sensors whose $\alpha_i(x)$ is less than 0. The remaining potential traffic capacity of i will then allocate to all upstream neighbors evenly. Let N_{U_i} be the number of upstream neighbors of sensor i . Sensor i will allocate the new data rate ($R'_{x,i}$) to its upstream neighbors as follows:

$$\begin{cases} R'_{x,i} = (|\alpha_i(x)| + \frac{PC'_i}{N_{U_i}}) & \text{if } (\alpha_i(x) < 0) \\ R'_{x,i} = (\frac{PC'_i}{N_{U_i}}) & \text{if } (\alpha_i(x) \geq 0) \end{cases}. \quad (6)$$

On the other hand, if $PC'_i < 0$, it means that the potential traffic capacity of i cannot satisfy the requirements of the upstream neighbors with $\alpha_i(x) < 0$. Sensor i will allocate all of the potential traffic capacity to the sensors whose $\alpha_i(x)$ is less than 0 according to the value of $\alpha_i(x)$. The more negative the value of $\alpha_i(x)$ is, the more the data rate is allocated to sensor x . Thus, sensor i will allocate the new data rate to its upstream neighbors as follows:

$$\begin{cases} R'_{x,i} = (PC_i \times \frac{|\alpha_i(x)|}{SUM_i}) & \text{if } (\alpha_i(x) < 0) \\ R'_{x,i} = 0 & \text{if } (\alpha_i(x) \geq 0) \end{cases}. \quad (7)$$

After a sensor suppresses the data rate of its upstream neighbors, it may cause their buffers to overflow and the congestion may happen at these upstream neighbors. However, these nodes will further suppress their upstream neighbors in the same way. This process repeats hop-by-hop towards the source node or leaf nodes. The whole network will reach the most effective congestion free.

Now we give two examples to illustrate our congestion control scheme. In Figs. 4 and 5, the two fields in brackets above each node denote the total flow size from upstream neighbors and the total flow size to downstream neighbors, respectively. The number on each link represents the data rate with which the sensor forwards to its downstream neighbor. For example, in Fig. 4, the symbol $[3/2]$ of sensor x represents that the

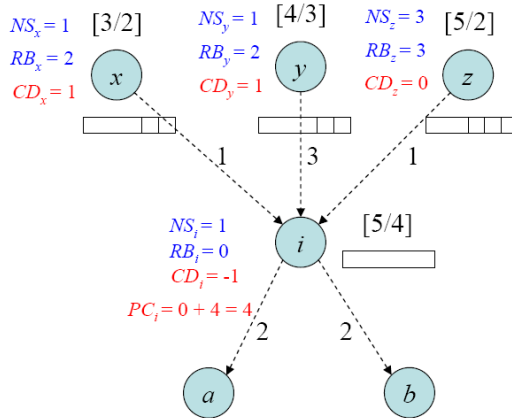


Fig. 4. Example of data rate adjustment for $PC'_i \geq 0$.

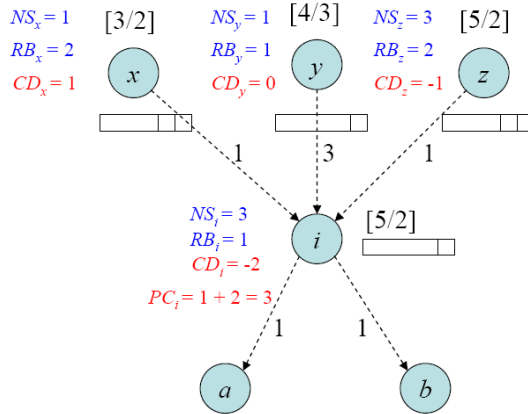


Fig. 5. Example of data rate adjustment for $PC'_i < 0$.

total flow size from upstream neighbors to x is 3 and the total flow size to downstream neighbors of x is 2. Assume that the remaining buffer size of sensors $x, y, z,$ and i are 2, 2, 3, and 0, respectively. The congestion degrees of sensors $x, y, z,$ and i are 1, 1, 0, and -1, respectively. The potential traffic capacity PC_i is 4, and $\alpha_i(x), \alpha_i(y),$ and $\alpha_i(z)$ are 0, -2, -1, respectively. We have $PC'_i = 4 - 3 = 1$. Since $PC'_i > 0$, sensor i will first allocate data rates 2 and 1 to sensors y and z , respectively. Then, the remaining potential traffic capacity of i will be evenly distributed to all upstream sensors. Based on Eq. (6), the new data rates of $R'_{x,i}, R'_{y,i},$ and $R'_{z,i}$ are $1/3, 7/3,$ and $4/3$, respectively.

Another example is shown in Fig. 5. Assume that the remaining buffer size of sensors $x, y, z,$ and i are 2, 1, 2, and 1, respectively. Then the congestion degree of sensors $x, y, z,$ and i is 1, 0, -1, and -2. Thus, the sensor i will adjust the data rate of its upstream neighbors $x, y,$ and z because its $CD_i < 0$ and may congest in the next time period. The potential traffic capacity PC_i is 3, and $\alpha_i(x), \alpha_i(y),$ and $\alpha_i(z)$ are 0, -3, -2, respectively. We have $PC'_i = 3 - 5 = -2$. The $PC'_i < 0$ means the sensors y and z may congest if sensor i suppress their data rates. Therefore, sensor i will allocate data rates to sensors y and z

according to their tendency congestion degrees $\alpha(y)$, and $\alpha(z)$. Based on Eq. (7), the new data rates of $R'_{x,i}$, $R'_{y,i}$, and $R'_{z,i}$ are 0, 9/5, and 6/5, respectively.

The HCCP Algorithm

For each sensor i in every time period T :

Each sensor broadcasts its congestion degree to neighboring nodes.

Phase 1: Congestion detection

1. Sensor i calculates its own CD_i .
2. Sensor i estimates whether congestion will happen or not in the next time period T .
3. When congestion will happen in the next time period is detected, sensor i broadcasts a suppressive message to suppress the data rate of its upstream neighbors.

Phase 2: Data Rate Adjustment

1. Sensor i calculates its own PC_i , PC'_i , $\alpha(x)$, and SUM_i .
 2. If $PC'_i \geq 0$, sensor i will allocate new data rate to the upstream neighbors according to Eq. (6).
 3. Otherwise, sensor i will allocate the new data rate to the upstream neighbors according to Eq. (7).
-

4. SIMULATION RESULTS

This section presents the simulation results. We compare the performance of HCCP with the previous rate-based and buffer-based congestion avoidance schemes. The packet collision caused by random media access is resolved by the MAC layer protocols. We measure the performance of our scheme with others in terms of the following metrics: packet drop rate, total source rate, and control overhead. We use ns-2 simulator for our simulations. Five hundred sensors are randomly placed in a 1,000 m \times 1,000 m area. The transmission range of the sensor is 100 m, with the transmission rate of 512 kbps. There is a sink deployed at the center of the deployment area. There are 100 data source nodes, randomly selected from the 500 sensors. The routing paths from source nodes to sink were discovered by AODV [20] protocol which is a shortest path protocol. The initial data rate of each source node is configured to four packets per unit of time. It may generate at a lower data rate due to congestion control. Each packet is 40 bytes long. The buffer at each sensor can hold 32 data packets. The simulation time is 200 seconds. We compare our HCCP scheme with rate-based scheme AFA [15] and buffer-based scheme BB [16].

4.1 Packet Drop Rate Comparison

In AFA scheme, a sensor i has a packet to forward j , only if the buffer of j is not full. If the buffer of j is full, i will hold the packet until it overhears a packet piggybacking a non-full buffer state from j . Thus, it does not cause packet drop. The BB scheme must make sure that a sensor i sends a packet to its downstream neighbor j only when j has buffer space to hold the packet. It does not cause packet drop, too. Our congestion control protocol can detect the congestion in advance and take the preventive measures.

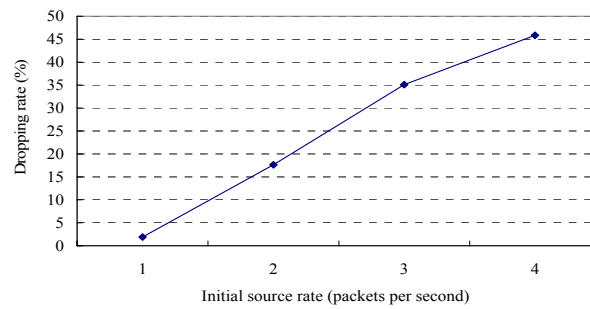
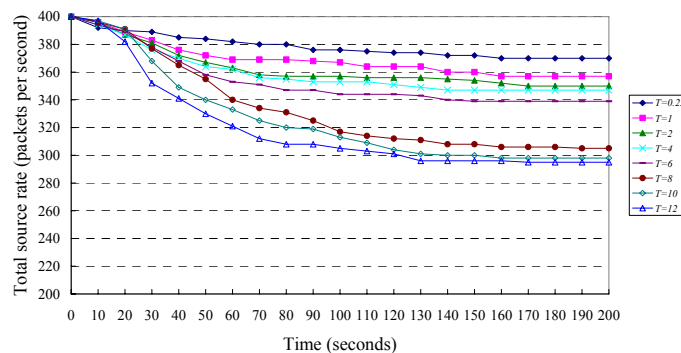


Fig. 6. Dropping rates without congestion control.

Fig. 7. The total source rate changes with various time periods T .

Therefore, HCCP ensures that there is no packet drop as well as BB and AFA. Fig. 6 shows the packet-dropping rate without congestion control. The dropping rate increases as the initial source rate increases.

4.2 Total Source Rate Comparison

The total source rate is defined as the total number of data packets generated by the data sources per second. Fig. 7 demonstrates, for HCCP scheme, how the time period T affects the total source rate changes with respect to simulation time. After time = 160 seconds, the source rates of all simulations is stable. In Fig. 7, the smaller time period T is, the larger the total source rate is. The total source rate will not increase anymore when T is smaller than 0.25 seconds. However, a smaller time period will cause higher control overhead due to the frequent congestion detection.

Fig. 8 confirms how much overhead of congestion HCCP can control with respect to the time period T . The overhead of congestion control is defined as the number of congestion control packets over the number of total delivery packets. When congestion in the next time period is detected, the sensor i will broadcast its congestion degree to advertise the neighbors to suppress the data rate of upstream neighbors. The more the number of congestions is detected, the higher the cost is needed. In Fig. 8, the percentage of congestion control overhead is minimum at $T = 4$ and maximum at $T = 0.25$. If T is too

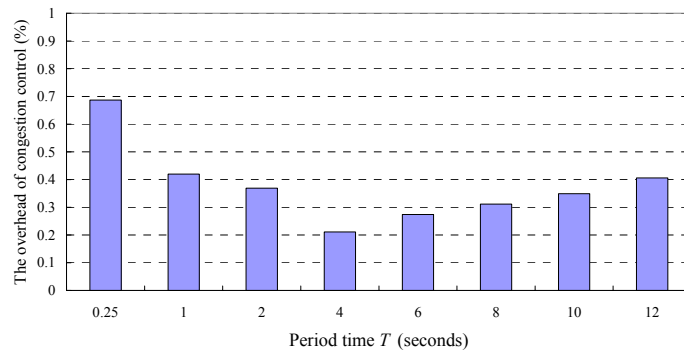


Fig. 8. The overhead of congestion control in HCCP with respect to time period T .

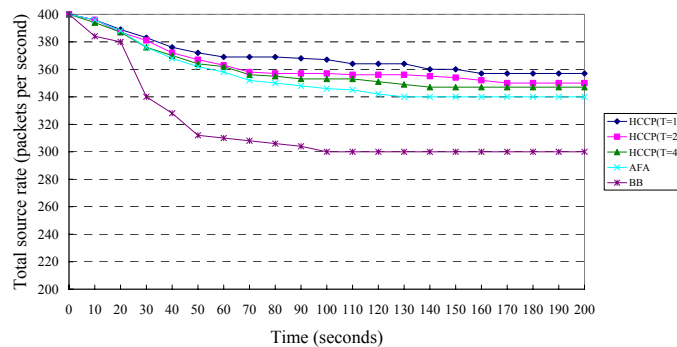


Fig. 9. Total source rate comparisons with three schemes.

long, the congestion will happen before time is expired. It will cause low performance of congestion control. Therefore, the time period T is set as 1, 2, and 4 seconds in the following simulations.

Fig. 9 compares the total source rates of three schemes with respect to simulation time. The BB scheme uses the 1/6-buffer algorithm to solve the hidden terminal problem. Every sensor advertised only one sixth of its remaining buffers. Therefore, the total source rate of BB is lower than others. AFA is not only utilizing the buffer effectively than BB but it can effectively allocate the data rate of upstream neighbors. The HCCP scheme combines the advantages of buffer-based and rate-based schemes. HCCP considers the packets delivery rate and remaining buffer size of each node at the same time. It can allocate effectively the data rate of upstream neighbors according to their tendency of congestion degrees. These simulations show that our congestion protocol HCCP is able to adjust effectively the proper data rate for sensors and obtains the better total data rate than other schemes.

Fig. 10 shows the total reduced source rate. The total reduced source rate is defined as the reduction of total source rate over the total source rate. Our proposed protocol has a better performance than other schemes with various initial source rates.

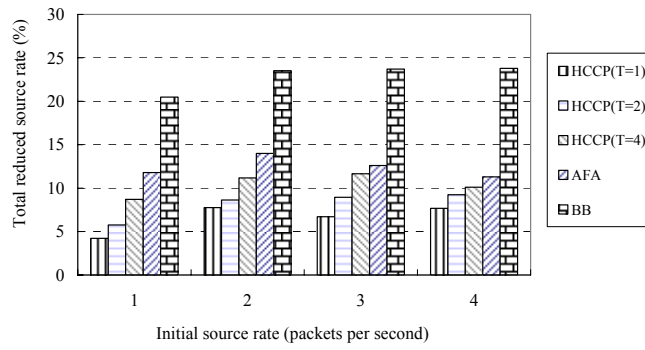


Fig. 10. Reduced source rates with respect to initial source rates.

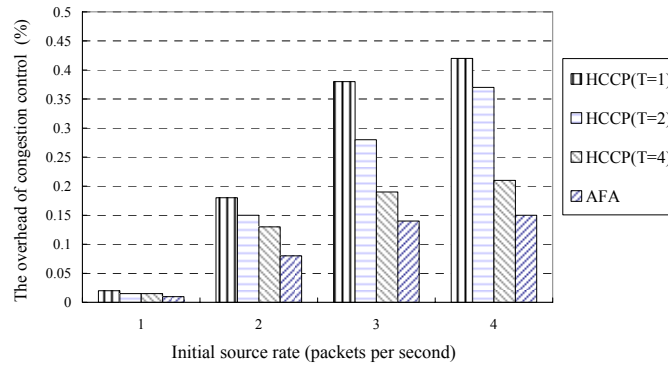


Fig. 11. The overhead of congestion control respect to initial source rate.

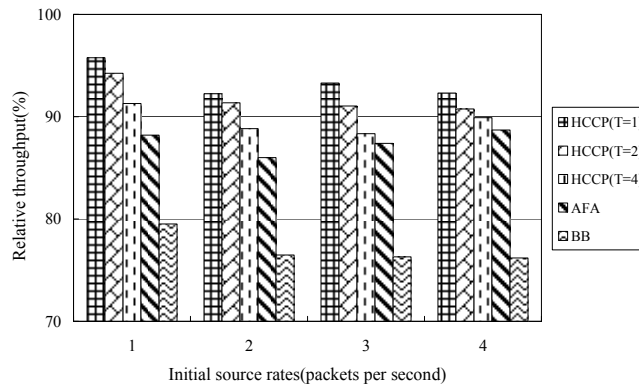


Fig. 12. The relative throughput with respect to initial source rates.

4.3 Control Overhead Comparison

The last simulation compares the overhead of congestion control with each scheme. The BB scheme always piggybacks its current buffer state by one bit in the frame header

of each data packet. The BB scheme does not need to broadcast suppressive message. Thus, we ignore the control overhead of BB scheme. The AFA and HCCP will broadcast the suppressive message when congestion is detected. Since the HCCP is more conservative than AFA scheme for congestion control, the HCCP needs higher control overhead than AFA scheme. Fig. 11 shows the overhead of congestion control with respect to initial source rate. In various initial source rates, the control overhead of our proposed scheme is higher than other schemes. However, the overhead of our protocol is smaller than 0.42% compared to the total number of data packets and the affection for the network throughput is negligible. Since the HCCP, AFA, and BB schemes have no packets dropping rate, the throughput of the three schemes is relative to the total source data rate and control overhead. The relative throughput of the three schemes is shown in Fig. 12.

5. CONCLUSIONS

In this paper, we have addressed the problem of congestion control in the sensor networks. We propose a Hybrid Congestion Control Protocol (HCCP), which considers both the packets delivery rate and remaining buffer size of each node. We discuss the congestion control problem including the congestion detection, congestion information advertisement, and data rate adjustment. In congestion detection phase, our HCCP detects the congestion in advance with a time period T and takes the preventive measures. In data rate adjustment phase, the upstream neighbors that tend to congest will be allocated more data rate. Simulation results show that the performance of our proposed protocol is better than the previous works in terms of total source rate. Although our control overhead is higher than other protocols, the control message compared to the total number of data packets is extremely low and can be negligible.

REFERENCES

1. J. Xu, X. Tang, and W. C. Lee, "EASE: An energy-efficient in-network storage scheme for object tracking in sensor networks," in *Proceedings of IEEE Sensor and Ad Hoc Communications and Networks*, 2005, pp. 396-405.
2. P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of the IEEE INFOCOM*, 2000, pp. 775-784.
3. A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao, "Habitat monitoring: Application drive for wireless communications technology," in *Proceedings of the ACM SIGCOMM Workshop Data Communication in Latin America and the Caribbean*, 2001, pp. 20-41.
4. E. Biagioni and K. Bridges, "The applications of remote sensor technology to assist the recovery of rare and endangered species," *International Journal of High Performance Computing Applications*, Vol. 16, 2002, pp. 315-324.
5. L. Schwiebert, S. Gupta, and J. Weinmann, "Research challenges in wireless networks of biomedical sensors," in *Proceedings of the International Conference on Mobile Computing and Networking*, 2001, pp. 151-165.
6. S. Lin, J. Zhang, G. Zhou, G. Lin, H. Tian, and J. A. Stankovic, "ATPC: Adaptive transmission power control for wireless sensor networks" in *Proceedings of the*

- ACM Conference on Embedded Networked Sensor Systems*, 2006, pp. 223-236.
7. S. C. Huang and R. H. Jan, "Energy-aware, load balanced routing schemes for sensor networks," in *Proceedings of the IEEE International Conference on Parallel and Distributed Systems*, 2004, pp. 419-425.
 8. H. Yang, F. Ye, and B. Sikdar, "A dynamic query-tree energy balancing protocol for sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, 2004, pp. 1715-1720.
 9. T. S. Chen, H. W. Tsai, and C. P. Chu, "Gathering-load-balanced tree protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2006, pp. 8-13.
 10. H. Dai and R. Han, "A node-centric load balancing algorithm for wireless sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference*, 2003, pp. 548-552.
 11. Z. Yang, L. Yuan, X. Du, and Q. Zhang, "Multipath load balancing delivery based on decisive energy ratio in wireless sensor networks," in *Proceedings of the IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, 2005, pp. 277-280.
 12. Y. Sankarasubramaniam, Ö. B. Akan, and I. F. Akyildiz, "ESRT: Event-to-sink reliable transport in wireless sensor networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2003, pp. 1003-1016.
 13. C. Y. Wan, S. B. Eisenman, and A. T. Campbell, "Congestion detection and avoidance in sensor networks," in *Proceedings of the ACM Conference on Embedded Networked Sensor Systems*, 2003, pp. 266-279.
 14. B. Hull, K. Jamieson, and H. Balakrishna, "Mitigating congestion in wireless sensor networks," in *Proceedings of the ACM Conference on Embedded Networked Sensor Systems*, 2004, pp. 134-147.
 15. S. Chen and Z. Zhang, "Localized algorithm for aggregate fairness in wireless sensor networks," in *Proceedings of the ACM International Conference on Mobile Computing and Networking*, 2006, pp. 274-285.
 16. S. Chen and N. Yan, "Congestion avoidance based on lightweight buffer management in sensor networks," in *Proceedings of the IEEE International Conference on Parallel and Distributed Systems*, 2006, pp. 934-946.
 17. C. Wang, K. Sohrawy, V. Lawrence, B. Li, and Y. Hu, "Priority-based congestion control in wireless sensor networks" in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2006, pp. 22-31.
 18. C. Wang, B. Li, K. Sohrawy, M. Daneshmand, and Y. Hu, "Upstream congestion control in wireless sensor networks through cross-layer optimization," *IEEE Journal on Selected Areas in Communications*, Vol. 25, 2007, pp. 786-795.
 19. C. Wang, B. Li, K. Sohrawy, M. Daneshmand, and Y. Hu, "A survey of transport protocols for wireless sensor networks," *IEEE Network Magazine*, Vol. 20, 2006, pp. 34-40.
 20. C. E. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in *Proceedings of the IEEE International Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90-100.



Jang-Ping Sheu (許健平) received the B.S. degree in Computer Science from Tamkang University, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees in Computer Science from National Tsing Hua University, Taiwan, R.O.C., in 1983 and 1987, respectively.

He is currently a Chair Professor of the Department of Computer Science, National Tsing Hua University. He was a Chair of Department of Computer Science and Information Engineering, National Central University from 1997 to 1999. He was a Director of Computer Center, National Central University from 2003 to 2006. His current research interests include wireless communications and mobile computing. He was an associate editor of *Journal of the Chinese Institute of Electrical Engineering*, *Journal of Information Science and Engineering*, *Journal of the Chinese Institute of Engineers*, and *Journal of Internet Technology*. He is an associate editor of the *IEEE Transactions on Parallel and Distributed Systems*, *International Journal of Ad Hoc and Ubiquitous Computing*, and *International Journal of Sensor Networks*.

He received the Distinguished Research Awards of the National Science Council of the Republic of China in 1993-1994, 1995-1996, and 1997-1998. He received the Distinguished Engineering Professor Award of the Chinese Institute of Engineers in 2003. He received the certificate of Distinguished Professorship, National Central University in 2005. He received the K. T. Li Research Breakthrough Award of the Institute of Information and Computing Machinery in 2007. Dr. Sheu is a Fellow of the IEEE, a member of the ACM, and Phi Tau Phi Society.



Li-Jen Chang (張力仁) was born in Taipei, Taiwan in 1981. He received his M.S. degree in Department of Computer Science and Information Engineering from National Central University, Taiwan, Republic of China, in 2007, and the B.S. degree in Department of Computer Science and Information Engineering from Providence University, Taiwan, Republic of China, in 2005, respectively. Currently, he is a software engineer in Software Design Department of ASUSTek Computer Inc.



Wei-Kai Hu (胡為凱) received the B.S. degree in Department of Computer Science and Information Engineering from Tunghai University, Taiwan, in 2002. Since September 2003, he has been working toward the Ph.D. degree in the Department of Computer Science and Information Engineering, National Central University, Taiwan. His current research interests include mobile computing, ad hoc network, and wireless sensor network.