International Journal of Innovative
Computing, Information and Control
Volume **8**, Number **7(B)**, July **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 5199–5210

A HIERARCHICAL CONGESTION AVOIDANCE ALGORITHM IN WIRELESS SENSOR NETWORKS

Yunlu Liu[1,2], Juhua Pu[1,2], Lei Shu[3] and Zhang Xiong[1,2]

[1]School of Computer Science and Engineering
Beihang University
No. 37, XueYuan Road, HaiDian District, Beijing 100191, P. R. China
lyunlu@gmail.com

[2]Research Institute of Beihang University in Shenzhen
Nanshan District, Shenzhen 518000, P. R. China

[3]Department of Multimedia Engineering
Graduate School of Information Science and Technology
Osaka University
1-1 Yamadaoka, Suita, Osaka 565-0871, Japan

Received April 2011; revised August 2011

Abstract. *In most wireless sensor networks (WSNs), all data converge to the sink. A node needs to forward data from its downstream nodes that are farther to the sink. Thus the nodes closer to the sink have relatively heavier traffic load, which easily results in congestion at these nodes. To deal with this problem, this paper presents a novel hierarchical congestion avoidance algorithm, HCA, and HCA provides different channel access opportunities to different nodes based on the distribution of traffic loads. In another word, HCA gives the nodes with heavier loads more chance to transmit data to avoid and mitigate congestion occurrence. For each node, its traffic load is estimated via the hierarchical level it locates and the number of children it has. Simulation results show that HCA algorithm reduces packet loss ratio, improves throughput and gains energy saving in single-path WSNs, which highlights the property of mitigating congestion of HCA.*

Keywords: Congestion control, MAC protocol, Wireless sensor networks (WSNs)

1. **Introduction.** In most wireless sensor networks (WSNs), all data converge to the sink in a hierarchical way. This hierarchical structure for data transmission gives nodes unequal responsibilities for traffic propagation. We call nodes that are close to sinks *upstream nodes* and nodes that are relatively farther to sinks *downstream nodes*. On condition that a node m transfers data for a node n, i.e., node n is a downstream node of node m, we call node n is a child of node m, while node m is the parent of node n. Generally, nodes closer to the sink and/or with more children will have higher traffic loads. In MAC protocols based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), such as IEEE 802.11 [1], the channel access mechanism is designed based on equal probability of channel access. In ordinary ad-hoc networks, since each node can also be the traffic destination, the traffic load is spread across the entire network. In this case an equal channel access strategy can work well. However, in a WSN, based on the analysis above, continuous sensory data streams from child nodes may cause and aggravate congestion at the parent node that is closer to the sink, as shown in Figure 1.

Congestion is one of the problems that have negative influence on network performance, e.g., data loss ratio, throughput, and energy consumption. In ad-hoc networks, congestion occurs when traffic load exceeds the network capacity. In WSNs, the origin of congestion in ad-hoc networks is not the only reason of congestion. To simplify the analysis, as shown in Figure 1, suppose only $c1$, $c2$, and p share the channel regardless of the influence of

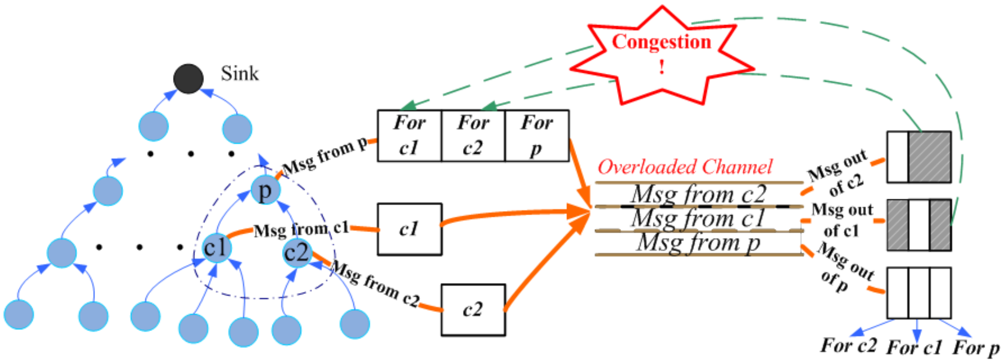


Figure 1. Problem brought by equal channel access in tree structured WSNs

the other nodes. Under equal channel access probability, the data from $c1$, $c2$, and p have the same opportunity to be transmitted. The successfully transmitted messages from $c1$ and $c2$ go to p for further transmission. Suppose each node has continuous data, during a period of time for transmission, p, $c1$, and $c2$ are permitted an equal access to the channel, and thus can send an equal quantity of data. Since p cannot send messages any faster than its children, and has the responsibility for transmitting all their messages, after some time, messages from $c1$ and $c2$ will cause congestion at p, and be discarded. The energy and bandwidth used for these discarded data are wasted. The congestion at p we mention here is mainly caused by the improper assignment of channel access probability. We call this problem as *semblance congestion*, which is a special case of congestion in tree structured WSNs due to the transmission structure. It is special because the origin is different from congestion in general ad-hoc networks. Reducing *semblance congestion* can mitigate congestion as a whole in tree structured WSNs.

Semblance congestion control involves the following three key problems:

1) Finding a suitable metric for semblance congestion control. When congestion occurs, it is hard to distinguish between regular congestion and semblance congestion. In order to evaluate the mitigation of semblance congestion we examine the following common WSN congestion control performance metrics: data loss, throughput, and energy consumption.

2) An efficient benchmark for channel assignment is required (i.e., the basis for the distribution of channel). In ideal conditions, channel assignment should fit the distribution of the traffic load. However, this is hard to measure. Since the closer a node is to the sink, the heavier traffic load it will have, we can approximate the load distribution via the information of hierarchical structure.

3) How to adjust the probability of being allocated access to the channel? CSMA/CA based MAC protocols, e.g., IEEE 802.11, provide several mechanisms to adjust the channel access probability, such as the minimal and maximal size of contention window, back-off increasing factor, inter-frame space, and back-off time distribution [2]. Thus semblance congestion control should have the capability to adjust nodes' access probabilities.

Taking into account these three issues mentioned above, we analyze the problem of semblance congestion in WSNs, and propose a novel semblance congestion control protocol: Hierarchical Congestion Avoidance Algorithm (HCA), and evaluate its performance via simulation experiments.

The rest of this paper is organized as follows. After surveying the related works in Section 2, we present the network model and notation definitions in Section 3. Section 4 addresses semblance congestion mitigation with HCA algorithm, followed by the evaluation of HCA's performance in Section 5. Finally, conclusions of this work are provided in Section 6.

2. **Related Works.** WSNs are restricted by energy consumption, memory, and communication bandwidth. In general, due to the characteristics of WSNs described below, congestion control schemes for traditional networks cannot, for the most part, be used directly in WSNs.

First, in a WSN, each node has to face the semblance congestion problem addressed in Section 1.

Second, energy consumption needs to be taken into consideration. Energy is always a key metric to evaluate an algorithm in WSNs.

Finally, WSNs are not reliable due to the unstable feature of wireless transmission. Information redundancy is widely involved, which means that most WSNs have a tolerance for message loss, i.e., it is generally not necessary to guarantee zero data loss.

It is challenging to design congestion control schemes for WSNs. The existing congestion control solutions for WSNs can generally be classified into two categories: traffic control and backup transmission.

Traffic control methods include source rate control [3-8] and forward rate control [9-13]. In PORT [3], ESRT [4], and CRRT [5], centralized control schemes are adopted based on reliable transmission. In PORT [3], the source-reporting rates are decided at the sink based on the consideration of energy consumption and the fidelity of the phenomenon knowledge. ESRT [4] is a transport solution where the reporting rates of the source nodes are decided at the sink based on the state of event reliability. In CRRT [5], the rate of the source nodes is assigned at the sink based on the rate assignment policy of the applications and is controlled at the sink via congestion notifications from intermediate nodes. IFRC [6] is a distributed method which uses certain rules and an Add Increase Multiply Decrease (AIMD) method to adjust the source rate. To gain efficiency, it requires the nodes to detect all incoming messages in order to check the congestion conditions [6]. DiffQ [7] is a differential backlog based MAC scheduling and router-assisted backpressure congestion control algorithm. CL-APPCC [8] is a predictive congestion control scheme based on a grid structure.

Forward rate control generally uses a distributed method [9,10]. This method may transfer the congestion backwards to the source nodes. It may be combined with source rate control to gain high efficiency, such as PCCP [9], CODA [10], and the work in [11,12].

Backup transmission can be multipath routing [13-15] or sink backup [16]. They generally need additional resources.

Most congestion control related work pays little attention to semblance congestion as described above, though some work on MAC optimization notices the problem of unfair channel assignment and prioritizes the traffic on MAC layer. In [11], the authors address a MAC layer method of mitigating congestion via adjusting the back-off window to increase the probability of the congested node gaining access to the channel, whereas the adjustment in [12] is done at the congested node after congestion has occurred to help reduce the queue occupancy of the congested node. HCA prevents semblance congestion via an adjustment at each node, which avoids the damage caused by congestion before control compared with the method in [12]. A constant parameter is used to apply the adjustment in [12], whereas in HCA the adjustment is applied based on the property of tree transmission structure. In [7], the authors give each flow different MAC layer priority to mitigate congestion combined with traffic control. This idea is similar to HCA; however, they have different objectives. The authors of [7] do not focus on the inherent problem of the WSNs transmitting to a sink node described above, which is the key focus of HCA; they instead focus on WSNs without sink nodes. In [8], authors aim at giving greater channel access opportunities to the nodes waiting for the channel for a longer time to improve the fairness in a grid structure. The objective of the work in [8] is not to tackle semblance congestion. Generally, MAC layer priority based work is oriented to traffic flow, whereas HCA is aimed at reducing semblance congestion caused by the inherent disadvantage of the special network structure. The combination of regular congestion control with HCA may further improve the efficiency of the previous work.

3. Network Model and Notation Definition. We use an undirected graph, $G = (S, E, s)$, to represent a WSN, where s is the sink node, $S = \{n|1, 2, \ldots, N\}$ is the set of sensor nodes, and E is the set of undirected communication edges. Also G can be described as a transmission tree rooted at s, $T = (S, EC)$, which has M levels ($EC \subseteq E$). This single path upstream transmission tree is one of the most common manners observed in WSNs. We make the following assumptions:

- The network is static; both nodes and sink are stable after deployment.
- Each node has the same transmission range; the sink cannot communicate with all nodes directly.
- The energy consumption of each node is mainly due to communication [17]; according to measurement in [18], the ratio of the energy consumption of transmitting to that of receiving is 2.5 to 2.

Generally, in a network such as the one described above, a node transmits the data of both its children and itself. A node closer to the sink and that with more children in T will have a greater traffic load. We state the condition more formally as follows:

Transmission Condition: in single-path WSNs with one sink, a node n, $n \in S$, has no lighter traffic load than any arbitrary child m.

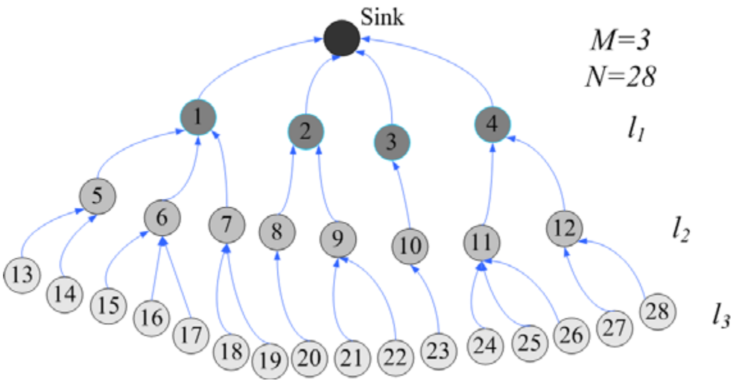


Figure 2. An example of the tree structured WSN

To facilitate the presentation, we give an example of the tree structured WSN shown in Figure 2. A node, n, which has i hops from the sink in tree T is on level $l(n) = l_i$; e.g., in Figure 2, $l(9) = l_2$; the set of nodes on level l_i or $l(n)$ is defined as L_i or $L(n)$, e.g., in Figure 2, $L(5) = L_2 = \{5, 6, 7, 8, 9, 10, 11, 12\}$; the children set of node n is denoted by C_n, e.g., in Figure 2, $C_7 = \{18, 19\}$; the children set of level l_i or $l(n)$ is defined as C_{l_i} or $C_{l(n)}$, e.g., in Figure 2, $C_{l(3)} = C_{l_1} = \{5, 6, 7, 8, 9, 10, 11, 12\}$; the average number of children of level l_i or $l(n)$ is defined as $\overline{C_{l_i}}$ or $\overline{C_{l(n)}}$, e.g., in Figure 2, $\overline{C_{l(5)}} = \overline{C_{l_2}} = 2$; the average number of children for the whole network is defined as $\overline{C} = \sum\limits_{l(i)=1}^{M-1} |C_i| \Bigg/ \sum\limits_{j=1}^{M-1} |L_j|$, which does not take the sink and the leaf nodes which have no child into consideration; the child coefficient of n is defined as $\alpha_n = \begin{cases} |C_n| / (\overline{C_{l(n)}}) & \overline{C_{l(n)}} \neq 0 \\ 0 & \overline{C_{l(n)}} = 0 \end{cases}$, e.g., in Figure 2, $\alpha_6 = 3/2$, $\alpha_5 = 1$, where $\overline{C_{l(n)}} = 0$ means n is the leaf node which has no child. α_n is the ratio of the number of node's children to the average number of children for the nodes on the same level. If α_n is larger than one, n has more children than average of $L(n)$; otherwise, if α_n is less than one, n has less children than average of $L(n)$, $i \in \{1, \ldots, M-1\}$, $n \in$S.

4. **Semblance Congestion Avoidance.**

4.1. **Main concept and mathematical analysis semblance.**

In CSMA/CA based MAC protocols, certain parameters are used for channel access control. Take the IEEE 802.11 protocol as an example, the contention window is one of the factors for channel assignment. The value of contention window, CW, can be set to the value between the minimum contention window, $CW_{\min}$, and the maximum contention window, $CW_{\max}$. If the channel is busy, transmission is deferred for a randomly selected back-off interval in $[0, CW]$. The contention window is initially set to the minimum contention window size, $CW_{\min}$, and doubled when a collision occurs until reaching the maximum contention window size, $CW_{\max}$. The smaller $CW_{\min}$ a node has, the less back-off interval it may take, and the greater channel access opportunity it should have. An equal value of $CW_{\min}$ across the whole network guarantees an equal probability of channel access in 802.11 [1]. In our case, the question becomes how to adjust $CW_{\min}$ to mitigate semblance congestion. The closer the channel assignment is approximate to the traffic distribution, the better result we will achieve. However, due to the limitations of WSNs (e.g., energy limitation), it is costly and unrealistic to provide an accurate centralized control. As described in the introduction, the traffic load of a node is related to its level in the tree structure and the number of children. We approximate the distribution of traffic loads using the hierarchical level and the number of children of each node. Firstly, we give a uniform $CW_{\min}$ value, $CW^{l_i}_{\min}$, $0 \le i \le M$, to the nodes on the same level based on the tree structure, which guarantees the fairness of the nodes on the same level and gives a benchmark to the calculation of the $CW_{\min}$ of each node. Secondly, each node calculates its $CW_{\min}$ based on the $CW_{\min}$ of its level and its child coefficient to fit transmission condition. The equation to set $CW_{\min}$ is given below to adjust channel access opportunity.

$CW_{\min}$ *of level* l_{i+1} *gives the benchmark of* $CW_{\min}$ *of the nodes on level* l_{i+1}:

$$CW^{l_{i+1}}_{\min} = \begin{cases} CW^{0}_{\min}(1+\overline{C_{l_0}})^{\chi} & i = 0 \\ CW^{l_i}_{\min}(1+\overline{C_{l_i}})^{\chi} & 0 < i < M \end{cases} \tag{1}$$

$$\chi = \log\left(1+\overline{C}\right)^{M}\left(A/(CW^{0}_{\min})\right) \tag{2}$$

where $CW^{0}_{\min}$ is the $CW_{\min}$ of the root node, $CW^{l_i}_{\min}$ represents $CW_{\min}$ of layer l_i, A is a constant ($A > CW^{0}_{\min}$), which provides an upper bound for $CW_{\min}$ below. We use the upper bound A to prevent a too large $CW_{\min}$ of a node, which makes the node have a small probability of sending out messages. The upper bound A can also adjust the algorithm based on the network environment (e.g., the scale of the network). Depending on the *Transmission Condition*, we keep the $CW_{\min}$ of node q no less than that of its parent p. The difference between $CW^{l_i}_{\min}$ and $CW^{l_{i-1}}_{\min}$ depends on the average number of children of L_{i-1} and adjustment coefficient χ, which is calculated from A.

$CW_{\min}$ *of node* n *based on* $CW^{l(n)}_{\min}$*:*

$$CW^{n}_{\min} = \begin{cases} ((1-B_{l(n)})\exp(1-\alpha_n) + B_{l(n)})CW^{l(n)}_{\min}, & \alpha_n > 1 \\ CW^{l(n)}_{\min}, & 0 \le \alpha_n \le 1 \end{cases} \tag{3}$$

$$B_{l(n)} = \left(1/\left(1+\overline{C_{l(n)-1}}\right)\right)^{\chi} \tag{4}$$

where we use α_n to adjust the difference of the number of children for the nodes on the same level. $B_{l(n)}$ is the ratio of $CW_{\min}^{l(n)-1}$ to $CW_{\min}^{l(n)}$, which is proved below.

(1)-(4) suffice to guarantee two theorems as follows.

Theorem 4.1. *$CW_{\min}$ of any arbitrary node n is larger than that of its parent m, i.e., $\forall n \in C(m)$, $CW_{\min}^n > CW_{\min}^m$.*

Proof: Since $\overline{C} > 0$, $M > 0$, $A > CW_{\min}^0 > 0$, $(1+\overline{C})^M > 1$, $A/(CW_{\min}^0) > 1$, thus $\chi = \log(1+\overline{C})^M (A/(CW_{\min}^0)) > 0$.

Then $\forall i \leq M$, we have $(1+\overline{C_{l_i}})^\chi > 1$, by (1), we get $CW_{\min}^{l_{i+1}} > CW_{\min}^{l_i}$, which means that $CW_{\min}$ of any arbitrary level is larger than that of its parent level.

Now we do a case study:

Case 1: $\alpha_n > 1$.

Since $0 < 1/(1+\overline{C_{l(n)-1}}) \leq 1$ and $\chi > 0$, by (4), we get $0 < B_{l(n)} \leq 1$. Because $0 < \exp(1-\alpha_n) < 1$, we get $0 < (1-B_{l(n)})\exp(1-\alpha_n) < 1 - B_{l(n)}$, then

$B_{l(n)} < (1-B_{l(n)})\exp(1-\alpha_n) + B_{l(n)} < 1$, by (1) to (4), we get $B_{l(n)} = CW_{\min}^{l(n)-1}/CW_{\min}^{l(n)}$.

Then $((1-B_{l(n)})\exp(1-\alpha_n) + B_{l(n)})CW_{\min}^{l(n)} > CW_{\min}^{l(n)-1}$, with (3), we get $CW_{\min}^{l(n)-1} < CW_{\min}^n < CW_{\min}^{l(n)}$.

If $\alpha_m > 1$, we have $CW_{\min}^m < CW_{\min}^{l(m)}$, $CW_{\min}^n > CW_{\min}^{l(m)}$, then $CW_{\min}^n > CW_{\min}^m$.

Otherwise, if $\alpha_m \leq 1$, we have $CW_{\min}^m = CW_{\min}^{l(m)}$, then $CW_{\min}^n > CW_{\min}^{l(m)} = CW_{\min}^m$.

That is to say, in this case 1, no matter what value α_m is, $CW_{\min}^n > CW_{\min}^m$ always holds.

Case 2: $\alpha_n \leq 1$.

Now we have $CW_{\min}^n = CW_{\min}^{l(n)}$.

If $\alpha_m > 1$, we get $CW_{\min}^m < CW_{\min}^{l(m)}$, then $CW_{\min}^n = CW_{\min}^{l(n)} > CW_{\min}^{l(m)} > CW_{\min}^m$.

Otherwise, $\alpha_m \leq 1$, and we have $CW_{\min}^m = CW_{\min}^{l(m)}$, then $CW_{\min}^n = CW_{\min}^{l(n)} > CW_{\min}^{l(m)} = CW_{\min}^m$.

That is to say, in this case 2, no matter what value α_m is, $CW_{\min}^n > CW_{\min}^m$ always holds.

With both case 1 and case 2, Theorem 4.1 holds.

With Theorem 4.1, we can make sure that (1)-(4) meet the *Transmission Condition*. Using (1)-(4), any arbitrary node has no greater channel access opportunity than its parent.

Theorem 4.2. *$CW_{\min}$ of any given node has an upper bound of A, i.e., $\forall n \leq N$, $CW_{\min}^n \leq A$.*

Proof:

$$\begin{aligned} CW_{\min}^{l_M} &= CW_{\min}^{l_{M-1}}(1+\overline{C_{l_{M-1}}})^\chi \\ &= CW_{\min}^0(1+\overline{C_{l_0}})^\chi(1+\overline{C_{l_1}})^\chi \cdots (1+\overline{C_{l_{M-1}}})^\chi \\ &= CW_{\min}^0[(1+\overline{C_{l_0}})(1+\overline{C_{l_1}}) \cdots (1+\overline{C_{l_{M-1}}})]^\chi. \end{aligned}$$

Applying the AG inequality (inequality of arithmetic and geometric means), we get

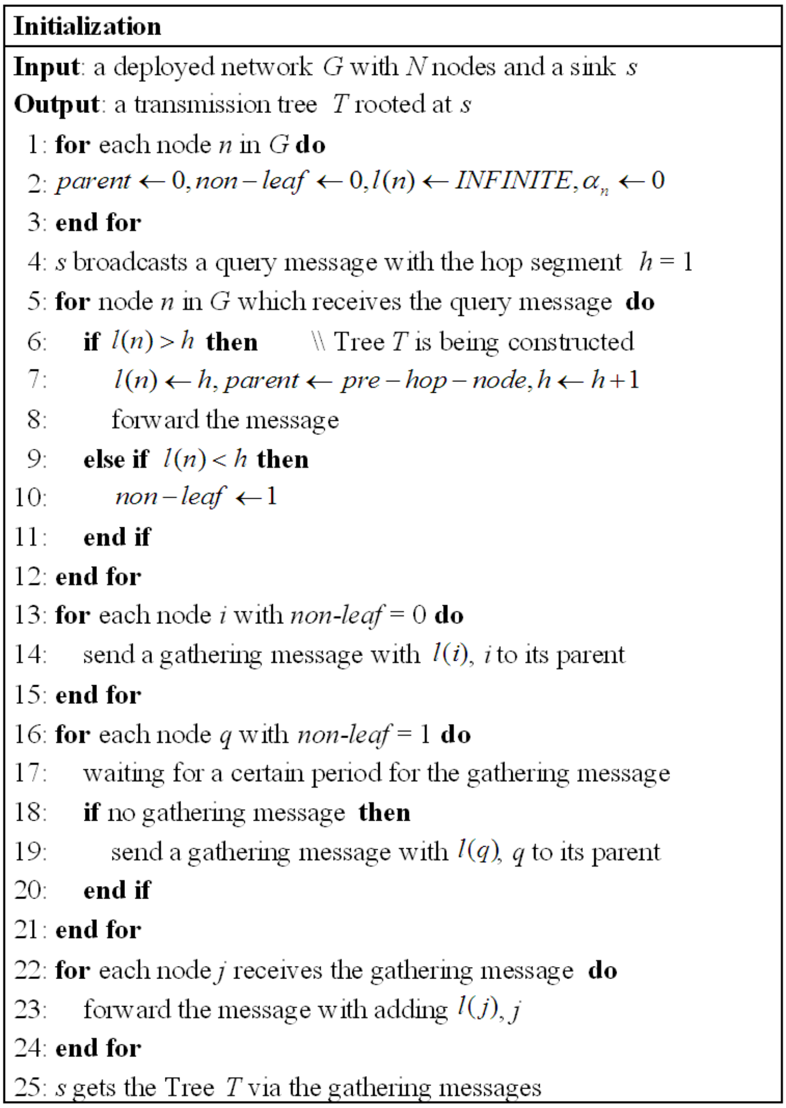
$$\begin{aligned} &\left[\left(1+\overline{C_{l_0}}\right)\left(1+\overline{C_{l_1}}\right)\cdots\left(1+\overline{C_{l_{M-1}}}\right)\right] \\ &\leq \left(\left(1+\overline{C_{l_0}}+1+\overline{C_{l_1}}+\cdots+1+\overline{C}_{l_{M-1}}\right)\Big/(M)\right)^M = \left(1+\overline{C}\right)^M. \end{aligned}$$

Hence, by the definition of $CW_{\min}^{l_{i+1}}$ $(i \leq M)$ in (1), we get

$$CW_{\min}^{l_M} \leq CW_{\min}^0\left[(1+\overline{C})^M\right]^\chi = CW_{\min}^0(A/(CW_{\min}^0)) = A.$$

According to the proof of Theorem 4.1, we get $\forall i \leq M$, $CW_{\min}^{l_{i+1}} > CW_{\min}^{l_i}$.
On the analogy of this, we get $CW_{\min}^{0} < CW_{\min}^{l_1} < \cdots CW_{\min}^{l_M} \leq A$.
If $\alpha_n > 1$, as proven above, $CW_{\min}^{n} < CW_{\min}^{l(n)}$, by $CW_{\min}^{l(n)} \leq A$, we get $CW_{\min}^{n} < A$.
Else if $\alpha_n \leq 1$, $CW_{\min}^{n} = CW_{\min}^{l(n)}$, by $CW_{\min}^{l(n)} \leq A$, we get $CW_{\min}^{n} \leq A$.

4.2. **HCA algorithm.** Based on (1)-(4), we propose our HCA algorithm which is shown in two parts: initialization, and HCA implementation, see Figure 3 and Figure 4 respectively. Given a deployed network with N sensor nodes and a sink, s, the initialization is the process of constructing a transmission tree T rooted at s. Each node n keeps several parameters: *parent* (parent id), *non-leaf* (1 if the node is not a leaf), $l(n)$ (the level of n), and α_n (the children coefficient). The parameters are initially set to zero. Sink s broadcasts a query message, which includes a hop segment, h, to record the hop number the message goes throw. Node n compares the value of its level with the hop count in the message: if its level is greater than the hop count, n sets the pre-hop node as its parent, and forwards the message, increasing h by one; otherwise it sets *non-leaf* $= 1$ which means it is not a leaf in the tree. T has been constructed. The next step is to send s the information. The nodes which have *non-leaf* $= 0$ send gathering messages

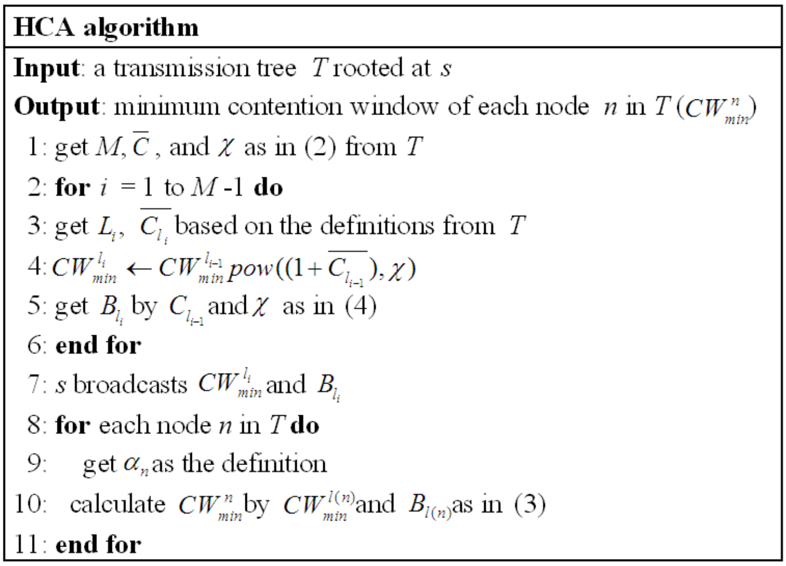


Initialization

Input: a deployed network G with N nodes and a sink s
Output: a transmission tree T rooted at s

```
for each node $n$ in $G$ do
    $parent \leftarrow 0, non-leaf \leftarrow 0, l(n) \leftarrow INFINITE, \alpha_n \leftarrow 0$
end for
$s$ broadcasts a query message with the hop segment $h = 1$
for node $n$ in $G$ which receives the query message do
    if $l(n) > h$ then        \\ Tree $T$ is being constructed
        $l(n) \leftarrow h, parent \leftarrow pre-hop-node, h \leftarrow h+1$
        forward the message
    else if $l(n) < h$ then
        $non-leaf \leftarrow 1$
    end if
end for
for each node $i$ with $non\text{-}leaf = 0$ do
    send a gathering message with $l(i)$, $i$ to its parent
end for
for each node $q$ with $non\text{-}leaf = 1$ do
    waiting for a certain period for the gathering message
    if no gathering message then
        send a gathering message with $l(q)$, $q$ to its parent
    end if
end for
for each node $j$ receives the gathering message do
    forward the message with adding $l(j)$, $j$
end for
$s$ gets the Tree $T$ via the gathering messages
```

FIGURE 3. Initialization



```
HCA algorithm
Input: a transmission tree $T$ rooted at $s$
Output: minimum contention window of each node $n$ in $T$ ($CW^{n}_{min}$)
 1: get $M, \overline{C}$, and $\chi$ as in (2) from $T$
 2: for $i$ = 1 to $M$-1 do
 3:   get $L_i$, $\overline{C_{l_i}}$ based on the definitions from $T$
 4:   $CW^{l_i}_{min} \leftarrow CW^{l_{i-1}}_{min} pow((1+\overline{C_{l_{i-1}}}), \chi)$
 5:   get $B_{l_i}$ by $C_{l_{i-1}}$ and $\chi$ as in (4)
 6: end for
 7: $s$ broadcasts $CW^{l_i}_{min}$ and $B_{l_i}$
 8: for each node $n$ in $T$ do
 9:   get $\alpha_n$ as the definition
10:   calculate $CW^{n}_{min}$ by $CW^{l(n)}_{min}$ and $B_{l(n)}$ as in (3)
11: end for
```

FIGURE 4. HCA algorithm

with their level and id to their parents. Node n which receives the message adds its level and id to the message and forwards it. After the initialization, s gets the structure information of the tree. For each level l_i of T, $CW^{l_i}_{\min}$ and B_{l_i} are computed at s based on (1), (2), and (4). s then broadcasts $CW^{l_i}_{\min}$ and B_{l_i}. The node n which receives the information calculates $CW^{n}_{\min}$ based on (3), and sets the minimum contention window. The algorithm here is suitable for WSNs under the assumptions in Section 3. Under the condition that the number of children for each node is approximately the same, which is relatively common under uniform node deployment, (1) and (3) can be simplified to $CW^{n}_{\min} = CW^{l(n)}_{\min} = CW^{0}_{\min}(1+\overline{C})^{l(n)\chi}$ by $\overline{C_n} = \overline{C_{l(n)}} = \overline{C}$, $n \in S$, $i \in \{1, \ldots, M-1\}$.

Then steps 5 and 8-10 in HCA algorithm are omissible in this instance.

5. Simulation Results.

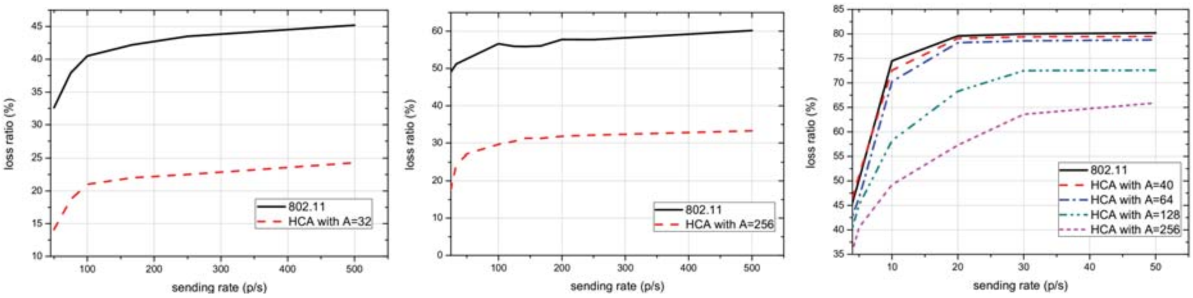
We simulate HCA on the NS2 simulator [19]. First, the performance of HCA is compared with the IEEE 802.11 protocol in three networks: one central sink with 20, 50, and 100 nodes in a 200 m × 200 m sensing field. The transmission range of each node is 40 m. The capacity of queue is set to 1000 packets since most motes [20] have 512K bytes memory.

Second, we give the comparison results among HCA, IEEE 802.11 and DPCC [21] with small queue capacity (60 packets) in a network with 50 nodes. We use small queue capacity to keep in accordance with the DPCC, because the DPCC uses queue length management to mitigate congestion and uses a small queue in the simulation setting [21]. Because we are only concerned with the comparison of energy consumption, we just give the relative measurements here. The energy consumption for transmitting a packet is set to 2.5 units, and for receiving a packet is set to 2 units according to [18].

In all our simulations, we focus on such three important performance metrics as data loss ratio, throughput, and energy consumption.

5.1. Data loss ratio.

Figure 5 depicts the packet loss ratio with varying packet sending rates. Data loss ratio is an important benchmark of congestion control. Here the improvement of the data loss ratio of HCA is due to the mitigation of the semblance congestion. Figure 5(a) shows the 20 node network. HCA with $A = 32$ reduces an average of more than 50% of the loss ratio compared to the 802.11 MAC protocol. Figure 5(b) presents the 50 node network. HCA with $A = 256$ reduces loss ratio by around 50% over 802.11.

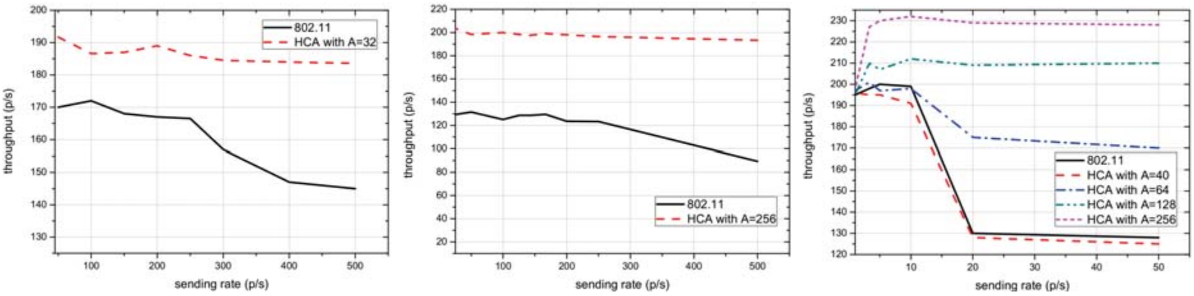


(a) 20 node network (b) 50 node network (c) 100 node network

Figure 5. Curves of data loss ratio under different sending rates comparing the 802.11 protocol with HCA

Figure 5(c) shows the 100 node network. The performance of HCA with different value of A is analyzed. The loss ratio goes up to 80% with 20 p/s sending rate for the 802.11 protocol. HCA with $A = 40$ behaves no better and sometimes even worse than 802.11 because the lower value of $CW_{\min}$ of the nodes which are closer to the sink may introduce more conflicts. HCA with $A = 256$ has a much more stable loss ratio with the change of the sending rate. Even when the loss ratio increases to 80% with 802.11, HCA with $A = 256$ achieves 15% less. According to the above results, it is clear that the value of A in HCA has a strong influence on efficiency, and the correct choice depends on the scale and the setting of the network.

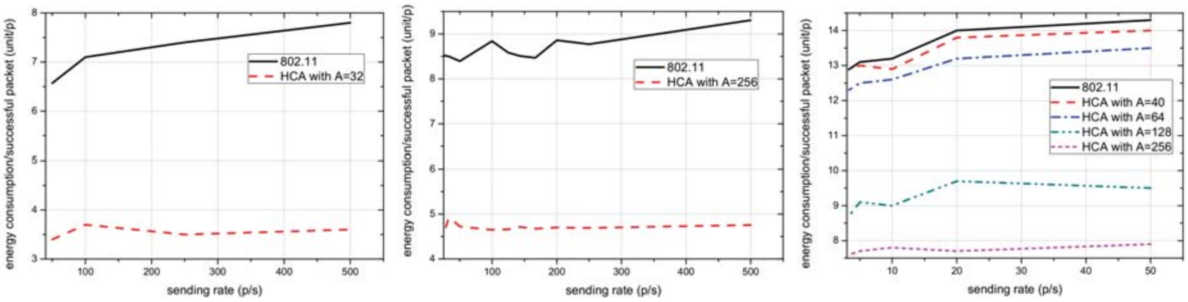
5.2. **Throughput.** Figure 6 shows HCA's benefit on throughput. HCA gains larger throughput than the 802.11 MAC protocol in almost all cases except when $A = 40$ in the 100 node network as shown in Figure 6(c). Furthermore, the throughput of HCA is more stable than that of 802.11 in almost all the cases. The simulation results show that HCA performs better than the 802.11 protocol with HCA in throughput.



(a) 20 node network (b) 50 node network (c) 100 node network

Figure 6. Throughput-rate curves comparing the 802.11 protocol with HCA

5.3. **Energy consumption.** Figure 7 shows the comparison of the energy consumption per successfully received message between 802.11 MAC protocol and HCA. Energy consumption is an unneglectable metric in protocol design of WSNs [22,23]. In Figure 7(a), we can see that energy consumption per successful message of HCA even goes down slightly when sending rate is beyond 250 p/s. We can find in Figure 5(a) that the data loss ratio of HCA during this period goes up slightly. We consider the reason is earlier drop of more failed messages. HCA saves energy via blocking unsuccessful packets at an early stage as well as by reducing the loss ratio. In the 100 node network, HCA with $A = 40$

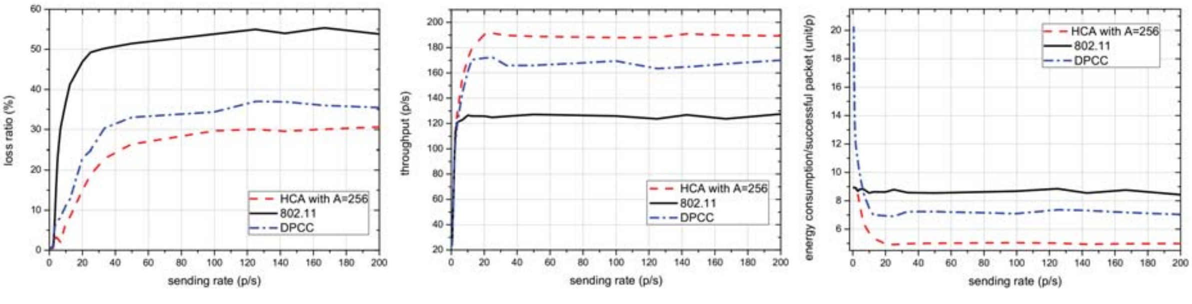


(a) 20 node network (b) 50 node network (c) 100 node network

Figure 7. Curves of energy consumption under different sending rates comparing the 802.11 protocol with HCA

consumes more energy per successful message than the 802.11 protocol when sending rate is below 5 p/s. The reason is much the same as for the data loss ratio that the small values of $CW_{\min}$ of the nodes closer to the sink inflict more channel conflicts. The number of retransmission increases which results in more energy consumption. The simulation results show that HCA offers a performance improvement on energy consumption.

5.4. **Comparison with DPCC [21] and 802.11.** Figure 8 shows the comparison of data loss ratio, throughput, and energy consumption of HCA, 802.11 and DPCC [21]. The interval to perform the calculations of the rate and back-off time of DPCC is set to 0.5 second. The ideal queue level in DPCC is set to 48 packets. The gain parameter k_{bv} in DPCC is set to 0.8. In Figure 8, it is very clear that HCA performs much better than both 802.11 and DPCC in packet loss ratio, throughput and energy consumption under a small queue capacity. In Figure 8(c), when the sending rate is below 8 packets per second, DPCC consumes much more energy than the other two. Only when the ratio of sending rate to control rate is in a certain range, the energy consumption of DPCC is less than that of the 802.11 protocol, which verifies that the introduction of control messages degrades the performance on energy consumption.



(a) Data loss ratio (b) Throughput (c) Energy consumption

Figure 8. Comparison of (a) data loss ratio, (b) throughput, and (c) energy consumption between HCA, DPCC, and the 802.11 protocol for 50 node network

6. **Conclusion.** In this paper, we have studied semblance congestion in WSNs, and proposed a hierarchical congestion avoidance algorithm: HCA. Different from most existing congestion control methods which aim at eliminating congestion, HCA manages to mitigate semblance congestion with little expense. Meanwhile, HCA cuts down resource

usage especially energy consumption via decreasing unsuccessful message transmission. Hierarchical and children information is used to adjust $CW_{\min}$ and further gains channel bandwidth assignment. Simulation results verify that HCA reduces loss rate and energy consumption significantly. HCA algorithm makes the following contributions:

1) An analysis on semblance congestion is addressed. To the best of our knowledge, it is still an outstanding problem.

2) A mathematical model is given to calculate the minimum contention window for a node based on observation and analysis of the transmission condition which describes the origin of semblance congestion.

3) Two theorems are given, which guarantees the minimum contention window of any arbitrary node n is larger than that of its parent m, i.e., any arbitrary node n has less probability of accessing the channel than its parent m. This proves the correspondence of minimum contention window and the transmission condition. An upper bound is provided to guarantee a reasonable value of minimum contention window.

4) Different from existing congestion control schemes, HCA focuses on channel assignment without rate control or additional resources.

5) HCA can be applied on its own or combined with other congestion control schemes, which increases the flexibility of HCA.

HCA concentrates on channel access assignment, which facilitates the combination of HCA with other congestion control protocols to make better use of network resources. It pays little attention to data overflow, which is the origin of ordinary congestion. In the future work, the combination of HCA and other congestion control schemes on data overflow is going to be done to gain better effectiveness.

Acknowledgment. This work is supported by the National Natural Science Foundation of China [60803120], Doctoral Fund of Ministry of Education of China [20091102110017], and Grant-in-Aid for Scientific Research (S) (21220002) of the Ministry of Education, Culture, Sports, Science and Technology, Japan. Yunlu Liu thanks Prof. Satyanarayanan and his group for the suggestions and support which facilitated this work. The authors also thank Dave Cooper, Nan Li, Hongcui Wang and anonymous reviewers for helpful suggestions, comments, and revision.

REFERENCES

[1] *IEEE Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (2007 revision)*, IEEE 802.11, 2007.

[2] W. Liu, W. Luo, X. Chen and Y. Fang, A QoS-enabled MAC architecture for prioritized service in IEEE 802.11 WLANS, *Proc. of IEEE Global Telecommunications Conference*, vol.7, pp.3802-3807, 2003.

[3] Y. Zhou and M. R. Lyu, PORT: A price-oriented reliable transport protocol for wireless sensor networks, *Proc. of the 16th IEEE International Symposium on Software Reliability Engineering*, Chicago, pp.117-126, 2005.

[4] Y. Sankarasubramaniam, O. B. Akan and I. F. Akyildiz, ESRT: Event-to-sink reliable transport in wireless sensor networks, *Proc. of the 4th ACM International Symposium on Mobile Ad hoc Networking; Computing*, Annapolis, pp.177-188, 2003.

[5] M. M. ALAM and C. S. Hong, CRRT: Congestion-aware and rate-controlled reliable transport in wireless sensor networks, *IEICE Trans. Commu.*, pp.184-189, 2009.

[6] S. Rangwala, R. Gummadi and R. Govindan, Interference-aware fair rate control in wireless sensor networks, *Proc. of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Pisa, Italy, pp.63-74, 2006.

[7] A. Warrier, S. Janakiraman, S. Ha and I. Rhee, DiffQ: Practical differential backlog congestion control for wireless networks, *Proc. of the 28th IEEE Conference on Computer Communications*, Janeiro, Brazil, pp.262-270, 2009.

[8] J. Wan, X. Xu, R. Feng and Y. Wu, Cross-layer active predictive congestion control protocol for wireless sensor networks, *Sensors*, vol.9, pp.8278-8310, 2009.

[9] C. Wang, B. Li, K. Sohraby, 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, pp.786-795, 2007.

[10] C. Y. Wan, S. B. Eisenman and A. T. Campbell, CODA: Congestion detection and avoidance in sensor networks, *Proc. of the 1st International Conference on Embedded Networked Sensor Systems*, Los Angeles, pp.266-279, 2003.

[11] B. Hull, K. Jamieson and H. Balakrishnan, Mitigating congestion in wireless sensor networks, *Proc. of the 2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, pp.134-147, 2004.

[12] M. H. Yaghmaee and D. A. Adjeroh, Rate control for service differentiation and congestion control in wireless multimedia sensor networks, *Computer Networks*, vol.53, pp.1798-1811, 2009.

[13] M. A. Razzaque and C. S. Hong, Congestion detection and control algorithms for multipath data forwarding in sensor networks, *Proc. of the 11th IEEE International Conference on Advanced Communication Technology*, Phoenix Park, Korea, pp.651-653, 2009.

[14] M. Z. Ahmad and D. Turgut, Congestion avoidance and fairness in wireless sensor networks, *Proc. of IEEE Global Telecommunications Conference*, New Orleans, pp.1-6, 2008,

[15] A. A. Bhorkar, T. Javidi and A. C. Snoeren, Achieving congestion diversity in wireless ad-hoc networks, *Proc. of the 30th IEEE International Conference on Computer Communications*, Shanghai, China, 2011.

[16] C. Wan and A. T. Campbell, Siphon: Overload traffic management using multi-radio virtual sinks in sensor networks, *Proc. of the 3rd ACM Conference on Embedded Networked Sensor Systems*, San Diego, pp.116-129, 2005.

[17] C. Intanagonwiwat, R. Govindan, D. Estrin, J. S. Heidemann and F. Silva, Directed diffusion for wireless sensor networking, *IEEE/ACM Transactions on Networking*, vol.11, pp.2-16, 2003.

[18] W. Ye, J. Heidemann and D. Estrin, An energy-efficient MAC protocol for wireless sensor networks, *Proc. of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, New York, pp.1567-1576, 2002.

[19] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*, 1st Edition, Springer, New York, 2009.

[20] F. Zhao and L. J. Guibas, Sensor network platform and tools, in *Wireless Sensor Networks: An Information Processing Approach*, Oxford, UK, Elsevier, 2004.

[21] M. J. Zawodniok and S. Jagannathan, Predictive congestion control protocol for wireless sensor networks, *IEEE Transactions on Wireless Communications*, pp.3955-3963, 2007.

[22] Y. Obashi, H. Chen, H. Mineno and T. Mizuno, An energy-aware routing scheme with node relay willingness in wireless sensor networks, *International Journal of Innovative Computing, Information and Control*, vol.3, no.3, pp.565-574, 2007.

[23] Y. Jin, J. Jo and Y. Kim, Energy-efficient multi-hop communication scheme in clustered sensor networks, *International Journal of Innovative Computing, Information and Control*, vol.4, no.7, pp.1741-1749, 2008.