Ad Hoc Networks 37 (2016) 240–255

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

# An energy-efficient QoS routing for wireless sensor networks using self-stabilizing algorithm

Da-Ren Chen[*]

*Department of Information Management, National Taichung University of Science and Technology, No. 129, Sec. 3, Sanmin Rd., North Dist., Taichung City 404, Taiwan, ROC*

ARTICLE INFO

*Article history:*
Received 23 March 2015
Revised 10 August 2015
Accepted 27 August 2015
Available online 8 September 2015

*Keywords:*
Wireless sensor networks
Self-stabilizing algorithm
TDMA
Real-time routing protocol

ABSTRACT

Transmission delays caused by wireless multi-hop communications usually hamper the time-sensitive applications on wireless sensor networks (WSNs). In this paper, transmission delay of data packets is quantified as the number of hops from a sensor to base station (BS), and a tolerable delay (TD) of each packet denotes the initial value of aging tag (*AT*) to present their QoS metric. On the basis of predictability of TDMA schedule, we propose a self-stabilizing hop-constrained energy-efficient (SHE) protocol for constructing minimum energy networks for hard real-time routing. The protocol first constructs ad hoc multi-hop paths within a cluster while controlling the number of nodes in the cluster so as to meet the TD of data packets from member nodes to their *CH*. An adaptive routing protocol is then proposed to convey aggregate data packets from *CHs* to BSs in different routes depending on their current *AT* values, thus meeting their QoS requirements while prolonging the network lifetime.

## 1. Introduction

Wireless sensor networks (WSN) consist of a large number of sensor nodes (SNs) which are capable of sensing, gathering, processing and transmitting data. SNs collect data on the target environment, and send the data to base station (BS) using wireless transmission techniques [5,6]. WSNs have been widely applied to industrial, military and civilian applications such as industrial plant management, motor/engine monitoring, target tracking, surveillance, health care system management and geographic information analysis, etc. [1–4].

In WSN applications [7,8], data transmission delay is a problem in multi-hop WSN networks. Many such applications entail time-critical requirements [2–4] such as weather monitoring, security and tactical surveillance, and are usually performed by time-sensitive tasks with bounded delay or deadline. WSN applications are also increasingly used to monitor the physical condition of aged people and hospitalized patients [7,8]. In multi-hop wireless ad-hoc sensor networks, transmission delays dominate over processing delays [7,9,10], and the effective control of transmission delays is needed to perform time-sensitive applications in such networks. We propose a self-stabilizing hop-constrained energy-efficient (SHE) protocol that uses a tolerable delay (TD) for hard real-time transmission in terms of the number of hops for packets in the network layer. This notion provides good compatibility and a metric for the application of a variety of WSN architectures.

Many researchers have investigated the transmission delay with respect to MAC layer [9,10,13,14] using traditional real-time scheduling for packet schedules on end-to-end packet delivery. Based on MAC layer protocols such as CSMA and TDMA, many energy-efficient and higher-throughput methods are proposed [7,13–15,41,42,52,53], a few of which focused on network layer in terms of quality of service (QoS) for WSN routing. For example, Souil et al. [41] propose an adaptive MAC protocol called AMPH with QoS for heterogeneous WSNs. AMPH provides heterogeneous traffic using traffic prioritization and increases channel utilization based on its hybrid and adaptive natures. Yigitel et al. [42] propose a QoS-aware MAC protocol called Diff–MAC with for wireless

---

[*] Tel.: +886 4 22196883/+886 933205261.
*E-mail address:* danny@nutc.edu.tw, danny@gm.nutc.edu.tw

http://dx.doi.org/10.1016/j.adhoc.2015.08.022
1570-8705/

multimedia sensor networks (WMSNs). Diff–MAC integrates existing QoS features and provides service differentiation and QoS provisioning to transmit heterogeneous traffic to meet the requirements of different traffic classes. Other protocols such as hexagonal topology [7,16,17] have been proposed for wireless networks based on MAC layer protocols. The protocols consider multi-hop transmission from SNs to BS while each cluster head (*CH*) contains only single hop to its SN members. In the structure discussed in [7,18], the topological control attempts to achieve specific connectivity among the nodes, and the deployments of SNs are determined manually. Their data packets are routed through pre-determined paths. Recently, there are some network layer and clustering protocols with QoS for WSN [34–36,43,44,47]. Tang and Li [43] propose a QoS routing protocol with optimal energy allocation for cluster based linear WSN topology. Each cluster has a *CH* modeled by single server fixed rate (SSFR) with finite capacity and different data arrival rate in accordance with the traffic from other clusters. Nazir and Hasbullah [44] propose an energy-aware QoS routing protocol called EEQR for cluster-based WSNs. EEQR provides QoS service for transmitting packets in accordance with network traffic including bandwidth and delay constraints, and uses different forwarding strategies. In [47], Ye et al. propose a redeployment method to solve energy balanced problem among mobile heterogeneous SNs and *CH*s in WSNs. This method with virtual force model can also improve the QoS of the deployment.

In a large-scale WSN, a cluster-based routing protocol [19–26] reduces energy consumption by performing data aggregation and fusion within a cluster, thus improving network lifetime and reducing network congestion. Moreover, clustering reduces channel contention, resulting in higher network throughput under heavy loads. In addition to supporting network scalability, clustering reduces the routing table kept in each SN [27]. Most cluster-based routing protocols in the literature determine the *CH*s stochastically and hamper the maximum network life time. In this paper, the transmission of each data packet is regarded as the hard real-time transmission with a predefined tolerable delay (TD). We aim to design a clustering method and a routing protocol to meet the hop constrains or hard deadlines of each data packet arriving at BS and improve both network and coverage lifetime.

In a distributed system with arbitrary states, a self-stabilizing algorithm can lead the system to a legitimate state and keeps the system in the legitimate state unless the system meets a subsequent transient fault [29,30,45,46]. Many studies regarding self-stabilizing algorithms have adopted Dijkstra's central demon model [31] or Burns' distributed model [32]. In a large-scale WSN, inter-cluster routing will substantially affect the lifetime of *CH*s. When the *CH*s located further from the BS submit their aggregate data directly to the BS, it will drain their batteries quickly and thus bring forward the time to cluster re-configuration. The proposed distributed self-stabilizing algorithm enables each *CH* to concurrently contact its adjacent *CH*s to construct time- and energy-efficient routes to the BS.

Radio transmission energy is critical to energy efficiency in WSNs, because the energy grows at least quadratically with the transmission range [16]. Some existing WSN protocols use identical or maximum transmission power to send data packets regardless of node distances. Many off-the-shelf low-power RF transceiver designs such as CC2420 and CC1000 [28,40] are programmed to a maximum of 31 different power levels, and have been implemented in ZigBee systems, PC peripherals, home/building automation and consumer electronics. These designs also benefit multi-hop cluster for power saving because the transmission ranges shorter than those in single-hop methods can be selected, and thus reduce the number of required clusters and prolong network lifetime.

The rest of this paper is organized as follows: Section 2 introduces existing methods that are related to this work. Section 3 describes the network model including system assumptions, QoS metrics and packet format. Section 4 describes the power model and outlines the proposed clustering and routing protocols. The protocols are also explained by examples. The performance evaluations given in Section 5 include time complexity analysis and simulations comparing to the existing methods. Finally, Section 6 concludes the paper.

## 2. Related work

Low-energy adaptive clustering hierarchy (LEACH) [23] is a distributed cluster-based routing protocol in which a large number of SNs are divided into several clusters. For each cluster, an SN is selected as a *CH* determined according to a given probability. LEACH-C [24] providing for the same steady-state phase as LEACH is a centralized deterministic clustering approach. It outperforms LEACH by dispersing the *CH*s throughout the network. In the set-up phase of LEACH-C, additional information such as location and residual energy are sent to the BS which computes ideal clustering decisions. Another noteworthy clustering protocol, called hybrid energy-efficient distributed clustering (HEED) [11], considers the residual energy of each SN and determines a good distribution of the *CH*s in the sensing area. In literature, some network layer clustering protocols with QoS for WSN [34–36,48–50] have been proposed. Diaz et al. [50] propose an ad hoc cluster-based protocol for logical sensor network topology to provide QoS-based multimedia streams. The QoS parameters such as packet loss, jitter, delay, bandwidths and multimedia properties are used for end-to-end transmissions. They also derive the maximum diameter for individual clusters, which is suitable for different types of multimedia streams. Atto and Guy [49] propose a power-aware cross-layer based protocol to provide QoS for wireless video sensor networks. The protocol keeps a high capture rate for some sensor nodes in the security applications in order to make better decisions to signal other nodes to move their status. Suganthe and Balasubramanie [48] propose a network layer protocol for mobile ad hoc networks which transmit data using message ferries and gateways. This protocol improves transmission rate and decreases communication overhead based on message ferrying scheme.

The chain-based technique is suitable for reducing the energy consumption in WSN. In the chain-based protocol, all of the SNs are interconnected into a chain, and a chain head will gather the data packets forwarded by the SNs along the chain. Power-efficient gathering in SN information systems (PEGASIS) is proposed to create a chain structure in WSN

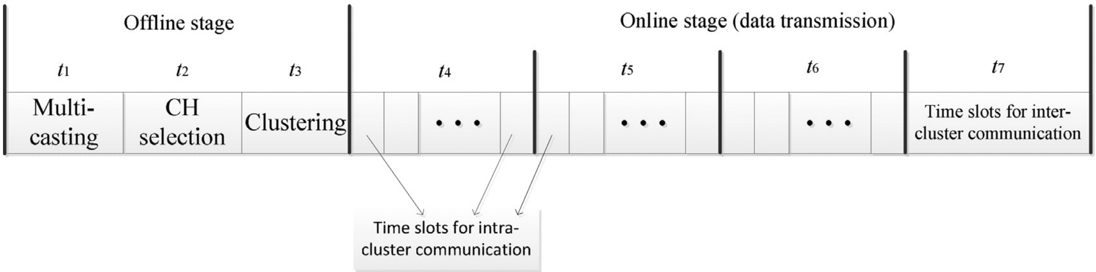


**Fig. 1.** Timeline for TDMA schedule.

by using greedy methods [25]. The object of this method is to shorten the data transfer distance between two SNs, thus reducing the energy dissipation of each SN. The flow-based routing protocol is a cluster-based and multi-level hierarchical protocol. Tao et al. [26] propose a multi-hop cluster protocol in which each SN is multi-hopped to BS while using single-hop to its *CH*. In general, an SN communicating to its *CH* by single hop may not reach the ideal energy efficiency, and thus is not suitable for large-scale WSNs due to limited battery power.

In the WSN literature, many clustering protocols use the information regarding overlapping area among SNs such as node density or degree to determine a *CH* or form a cluster [15,17–20]. When using a *truly* multi-hop protocol, network information cannot provide broader and detailed distribution information of SNs outside the overlapping area and cannot effectively reduce the number of the active *CH*s. Time-to-live (TTL) [37,38] is an aging technique that limits the lifespan of IP packets in the internet. Aging tag (AT) applied in SHE is to localize clustering information thus controlling cluster sizes.

In this paper, the proposed QoS protocol, SHE is composed of intra-clustering and inter-clustering methods. The intra-clustering method applies multicast aging messages to sketch suitable multi-hop clusters. SHE provides intra-cluster multi-hop transmission while most previous works study one-hop cluster protocols. The proposed clustering method initializes the aging tags (ATs) of individual data packets to TD and changes them from SNs to *CH* according to their individual hop constraints. The inter-clustering method constructs spanning trees among *CH*s and BSs using a distributed self-stabilizing algorithm to determine suitable routes for aggregated packets according their remaining hops.

## 3. System model

### 3.1. Network model

We assume that direct-sequence spread spectrum (DSSS) is applied in the proposed model to reduce inter-cluster interference. We use transmitter-based code assignment [39], where all transmitters within the cluster use the same spreading code. Once the spanning tree among *CH*s is constructed, the BS assigns unique spreading codes to the *CH*s according to their Euclidean distances from the BS. In each cluster, the *CH*s transmit the TDMA schedule [39,52,53] and their unique spreading code to their affiliated member nodes. After the TDMA schedule is known to all nodes in the cluster, *offline stage* is completed, and *online stage* (data transmission) can begin. The proposed method follows the timeline shown in Fig. 1. TDMA schedule in a cluster head has good predictability corresponding to the number of its affiliated member nodes. We assume that the delay due to the MAC or data link layer is regarded as the part of individual hop transmissions.

### 3.2. Assumptions and notations

Firstly, we define notation $H_{lower} \in N$ as the lower bound in the hop-count for all packets/messages arriving at the BS, where $N$ denotes the set of natural numbers. In other words, all message services in the system must tolerate the delay of at least $H_{lower}$ hops. Based on the TDMA schedule, the delay for each hop transmission can be parameterized and estimated. For a service that cannot tolerate $H_{lower}$ hops delay will be suspended and prompted by the system. There are three kinds of messages or packets in the underlying model. *Exploring messages* (*EMs*) broadcasted by SN with given AT (AT is initialized to $H_{lower}$) are the beacons that contain necessary information for finding suitable clusters and *CH*s. *Sensing packets* conveying sensing data are generated by SNs periodically and forwarded by other SNs until they reach their *CH*s. *Aggregating packets* produced by *CH*s are the fusion of the sensing packets and are forwarded to the BS. *Exploring* messages *EMs* are broadcasted once by each SN in the initial network stage. *Sensing* and *Aggregating* packets have individual tolerable delays (i.e., hop-constrained) for the QoS metric when they are produced by source nodes. There exists an immobile BS in the target WSN of which transmission range covers all the SNs in the target field and has no power limitation. There exists a set $\tau$ of $n$ immobile homogeneous SNs which are randomly deployed over the target field. The communication range of each SN is defined by a disk centered on the SN with omni-directional coverage in which all SNs can receive messages from that SN. We assume each SN has $t$ transmission power levels $r_1, r_2, \ldots, r_t$, where $r_t = r_{max}$ denotes the maximum transmission range. The set of SNs within the transmission range $r_i$ of SN $k$ is defined as $A_k^i$. All SNs have identical and limited initial energy, and other assumptions are listed below:

- Each SN is well aware of its channel error, such that it can derive the most suitable transmission range which consumes the minimum power while providing acceptable data transfer rate.
- Communication links are unidirectional. Node $y$ can receive the signal from node $x$ but not necessarily vice versa.

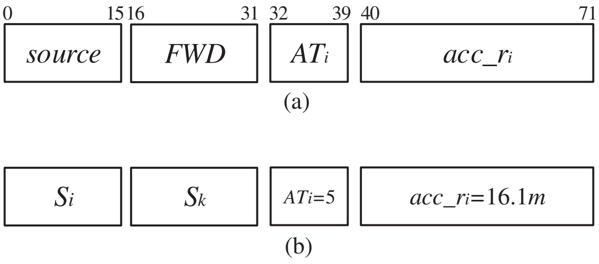


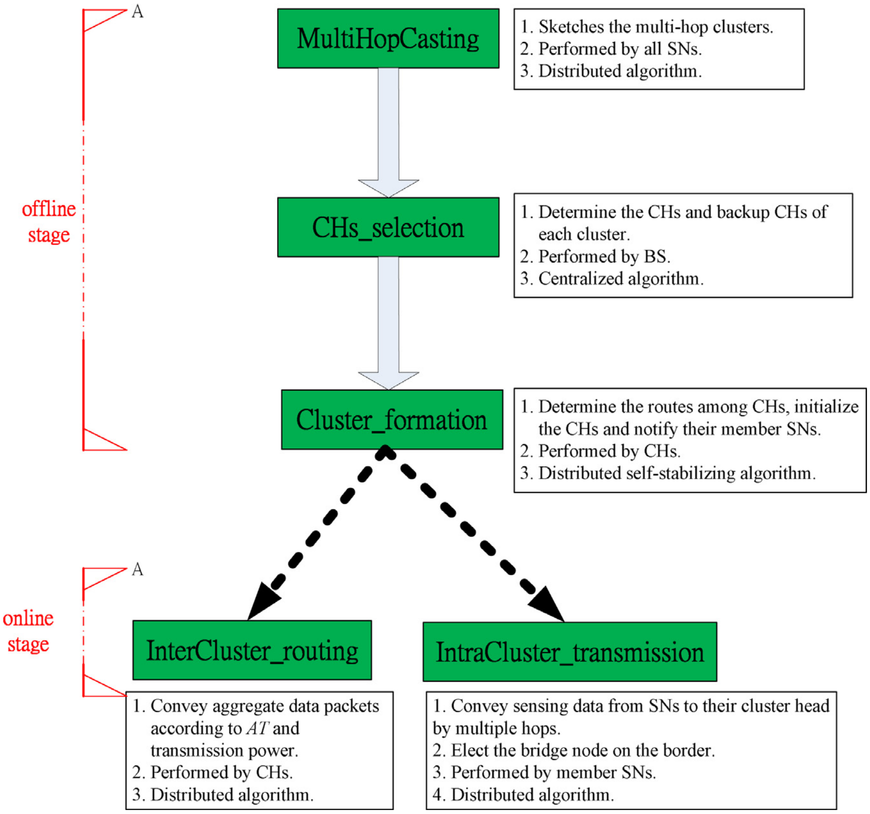
**Fig. 2.** The (a) format and (b) example of exploring message $EM_i$.

- If the transmitting power is known, each node can compute the approximate distance to another node based on the received signal strength index (RSSI) [51].
- On the basis of TDMA schedule [39,52,53], the transmission delay due to MAC layer protocols can be modeled and predicted.

At network initiation, SN $S_k$ forwards an exploring message $EM_k$ generated by SN $S_i$ that contains at least four entries as shown in Fig. 2(a). They denote the ID of source node $S_i$, the recent forwarder $S_k$ denoted as *FWD*, the value of aging tag $AT_i$, and accumulated transmission radius $acc\_r_i$. Fig. 2(b) presents an example of an exploring message that originated from node $S_i$ and recently forwarded by $S_k$. $EM_i$ has currently five available hops and walks an accumulated length of 16.1 m along its route. Each SN queues the exploring messages using *MsgQ* and removes duplicate items having identical source.

## 4. Proposed method

The framework of the proposed method shown in Fig. 3 comprises *offline* and *online* stages. To construct a cluster, the proposed multi-hop clustering *MultiHopCasting*, *CH_selection* and *Cluster_formation* select suitable *CHs* and backup *CHs*. The data packet routing within a cluster is achieved by *IntraCluster_transmission*. In the *offline* stage, the algorithms determine the location and size of each cluster by multicasting messages $EM_i$. They leave important *footprints* for constructing clusters with diameters of at most $H_{lower} \times 2$ hops. The cluster formation phase is performed only once at the network initialization stage, resulting in a significant reduction in cluster re-configuration overhead, i.e., time and power consumption. Multi-hop routing within each cluster also decreases the number of clusters in the network. To meet the hop constraints of each message, inter cluster routing protocols are composed of *Cluster_formation* and *InterCluster_routing* which consider the $AT_i$ of individual aggregate data packets *i*. By applying the distributed self-stabilizing algorithm, *Cluster_formation* constructs the routes with the minimum number of hops between *CHs* and the BS. The algorithm *InterCluster_routing* is an adaptive routing protocol that conveys aggregate data messages in different routes depending on their individual hop constraints, i.e., the *AT*.



1. Sketches the multi-hop clusters.
2. Performed by all SNs.
3. Distributed algorithm.

1. Determine the CHs and backup CHs of each cluster.
2. Performed by BS.
3. Centralized algorithm.

1. Determine the routes among CHs, initialize the CHs and notify their member SNs.
2. Performed by CHs.
3. Distributed self-stabilizing algorithm.

1. Convey aggregate data packets according to *AT* and transmission power.
2. Performed by CHs.
3. Distributed algorithm.

1. Convey sensing data from SNs to their cluster head by multiple hops.
2. Elect the bridge node on the border.
3. Performed by member SNs.
4. Distributed algorithm.

**Fig. 3.** Protocol framework.

### 4.1. *MultiHopCasting and CHs_selection for multi-hop clustering*

In *MultiHopCasting*, each SN $s_i$ broadcasts exploring messages *EM* with an initial hop-count $AT_i = H_{lower}$ to outline a multi-hop cluster with a diameter not more than $H_{lower} \times 2$ hops. Each SN also forwards the messages from its neighbors and decreases their *AT* values by one. It records the *valid EM* and discards the *EM* with $AT_i = 0$ or from the same source but traversing through longer routes. After all *EM*s have vanished from the network, each SN summarizes the received *EM*s in *MsgQ*. Algorithm *CHs_selection* adjusts the locations and shapes of clusters by merging smaller clusters. At the end of cluster formation, the algorithm determines the *CH*s and their backup *CH*s. In *MultiHopCasting*, assuming $Q_i$ and $Ne_i$ denote the set of items from *MsgQ* and the neighbors of SN $s_i$, respectively, and $|Q_i|$ and $|Ne_i|$ denote the number of elements in the sets. Notation $Me_i$ denotes a set of member SNs from which the messages are forwarded by the SNs in $Ne_i$ to SN $s_i$. In accordance with the *FWD* fields of *MsgQ*, each SN knows its neighbors and counts the frequencies of its neighbors. When time $t_0$ expires, each SN has to determine whether to be a potential *CH*.

```
Algorithm MultiHopCasting executed by a SN $s_k$

Receive $CL(t_0)$ from BS and transmission range $r_x$ covers at least $\sigma$
neighbors, $1 \le x \le r_{max}$

On broadcasting an exploring message
 1. Find a radius current_radius such that at least two SNs can be
    covered,
 2. $ID=s_k$, sender $= s_k$, hop_count $AT_k = H_{lower} - 1$,
    $acc\_r_k = current\_radius$,
 3. Broadcast $EM_k$ using current_radius.

On receiving $EM_i$ from SN $s_j$,
 4. IF $AT_i = 1$ THEN discard $EM_i$,
 5. IF $EM_i$ is repeated in MsgQ,
 6.    IF $w_i^+ < w_i$ THEN
 7.       update i-th item in MsgQ, $hc_i = hc_i^+$, $acc\_r_i = acc\_r_i^+$,
          $sender_i = sender_i^+$.
 8.    ELSE discard $EM_i$.
 9. ELSE insert $EM_i$ in MsgQ according to $hc_i$.

On forwarding a exploring message $EM_i$
10. $hc_i --$,
11. $acc\_r_i += current\_radius$,
12. $sender_i = s_k$,
13. broadcast $EM_i$ using current_radius.

On expiration of $t_0$
14. Find a set $Ne_k$ of neighbors by FWD in $MsgQ_k$ and count their
    frequency by sender field to obtain $Me_k$.
15. Find an SN $s_x$ in $Ne_k$ with maximum value in $Me_k$.
16. Return position $Po_x$ and $MsgQ_x$ of $s_x$ to BS as a potential CH.
```

When an SN $s_k$ receives a duplicated $EM_i$with identical *source* in the $MsgQ_k$, it updates $MsgQ_k$ according to the entry with the smaller $acc\_r_i$ value. In lines 3–8, the notations $w_i$ and $w_i^+$shown in Eq. (2) denoting the energy cost are respectively derived from *source* = $s_i$ in $MsgQ_k$ and a new arrival packet originated from $s_i$.

$$w_i = \left(\frac{acc\_r_i}{H_{lower} - AT_i}\right)^2 \times (H_{lower} - AT_i), \tag{1}$$

$$w_i^+ = \left(\frac{acc\_r_i^+}{H_{lower} - AT_i^+}\right)^2 \times (H_{lower} - AT_i^+). \tag{2}$$

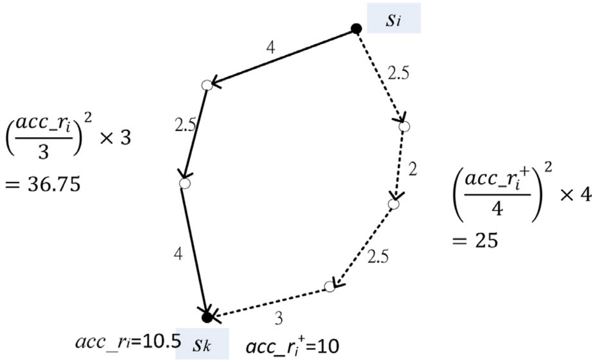


**Fig. 4.** An example to the cost of routes.

For power-aware routing, the square of the average $acc\_r_i$ and $acc\_r_i^+$ represent the average energy cost of each link because the transmission distance is at least in quadratic relation to the transmission power. For example, Fig. 4 depicts that there are two routes from $s_i$ to $s_k$ with different accumulated lengths (weights). The equations can be modified as various power models to adapt actual network applications

In Fig. 2(a), each *EM* message is not longer than 9 bytes, and its transmission hops are limited by *AT* values such that the memory required to store *MsgQ* is relatively small.

Algorithm *CHs_selection* performed by the BS determines the clusters and their *CH*s. It also evaluates their required backup *CH*s (*BCH*s) so as to maximize cluster lifetime without re-configuration. Because the lifetime of most SNs determines the lifetime of a cluster, we first estimate the average lifetime of *CH*s and non-*CH* SNs. The lifetime lets us compute the number of required backup *CH*s to provide maximum cluster lifetime. Assuming the battery of each SN has initial capacity of 1 *Ah* ampere-hour in $v$ electron volt, denoting $Ah \times v$ watt-hour or

$$E_{bat} = 3600 \times Ah \times v \tag{3}$$

joule. The total energy of a cluster *c* is

$$E_{cap} = 3600 \times Ah \times v \times |Q_c| \tag{4}$$

joule, where $|Q_c|$ is the size of a set $Q_c$. Each non-*CH* SN submits *m* packets per hour using TDMA method. Since $AT_i$ denotes the remainder number of links traversed by a message to *CH*, there are

$$\frac{m \times \sum_{s_i \in Q_c} AT_i}{\text{hour}}. \tag{5}$$

hops per hour required by the SNs in cluster *c*. In the worst-case, $AT_i = H_{lower}$ to reach the *CH*. According to their *MsgQ*s, the average transmission distance of each SN in cluster *z* is

$$\gamma_z = \frac{\sum_{s_i \in Q_z} \frac{acc\_r_i}{H_{lower}}}{|Q_z|}. \tag{6}$$

By using the radio model in [13,24], we compute the expected lifetime of each SN in a cluster excluding *CH*. In this model, both the free-space ($r_k^2$ power loss) and multi-path fading ($r_k^4$ power loss) channel models are based on the distance between the transmitter and receiver. By Eq. (6), when $r_k = \gamma_z$, the average energy required to transmit a $\ell$-bit message in the distance of $\gamma_z$ can be derived as [24]

$$E_T(\ell, z) = E_{T-elec}(\ell) + E_{T-amp}(\ell, z) = \begin{cases} lE_{elec} + l\varepsilon_{fs}\gamma_z^2, & \text{if } \gamma_z < \gamma_0 \\ lE_{elec} + l\varepsilon_{mp}\gamma_z^4, & \text{if } \gamma_z \geq \gamma_0 \end{cases} \tag{7}$$

where $\gamma_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}$, $E_{elec}$ is the required energy to transmit or receive $\ell$ bits of data. The amplifier energy due to multi path $\varepsilon_{mp}\gamma_z^4$ or free space $\varepsilon_{fs}\gamma_z^2$, counts on the distance to the receiver under an acceptable bit-error rate. When an SN receives $\ell$-bit data, it requires energy

$$E_R(\ell) = \ell E_{elec}. \tag{8}$$

The energy parameters reported in [13] are $E_{elec} = 50nJ/\text{bit}$, $\varepsilon_{fs} = 10pJ/\text{bit}/m^2$ and $\varepsilon_{mp} = 0.0013pJ/\text{bit}/m^2$. In Eq. (5), the expected energy requirement for all SN of cluster $z$ in an hour is

$$E_{req}(z) = (E_T(\ell, z) + E_R(\ell)) \times m \times \sum_{s_i \in Q_z} AT_i. \tag{9}$$

In Eqs. (4) and (9), the lifetime of member SNs represents the expected lifetime of cluster $z$:

$$L_C(z) = \frac{E_{cap}}{E_{req}(z)}. \tag{10}$$

Assume the distance between any SN and BS is smaller than or equal to $r_{max}$. The power consumption required by each *CH* for receiving packets from member SNs, aggregating the packets and sending packets to the BS. The energy requirements for a *CH* can be denoted as:

$$E_{CH-req} = \left(mn'\ell E_{elec} + mn'\ell E_{DA}\right) + \left(\ell E_{elec} + \ell\varepsilon_{fs}\gamma_{max}^2\right). \tag{11}$$

where $n'$ and $m$ denote the number of its neighbors and packets from each neighbor per hour, respectively. Notation $\gamma_{max}$ is the maximum distance between a *CH* and BS, and $E_{DA}$ is energy requirement for one bit. By using Eqs. (3) and (11), the lifetime of a *CH* can be approximated as

$$L_{CH} = \frac{E_{bat}}{E_{CH-req}}. \tag{12}$$

Let $CHL(z)$ be the set of required *CHs* in cluster $z$, we assume that service duration of *CHs* and their clusters are the same, the number of required *CHs* can be denoted as

$$|CHL(z)| = \left\lceil \frac{L_c(z)}{L_{CH}} \right\rceil. \tag{13}$$

In accordance with $|CHL(z)|$, BS selects suitable backup *CHs* by using the received *MsgQ* from the potential *CHs*. Eq. (11) is improved by reducing $\gamma_{max}^2$ using the proposed inter-cluster routing, thus further reducing *CH* power consumption.

Algorithm *CHs_selection* executed by BS.

```
1.  Receive a set $PCH$ of potential $CH$s and their $MsgQ$.
2.  Construct a set $V$ of all SNs in the network.
3.  Compute the values of $|CHL(z)|$ for $s_z \in PCH$.
4.  Remove $s_i \in PCH$ from $PCH$, where $|CHL(z)| > |Ne_z|$.
5.  Let $FCH=\{\emptyset\}$ be a set of formal $CH$s.
6.  WHILE $V \neq \{\emptyset\}$.
7.      IF $\{(i,\ j)|s_i,\ s_j \in PCH,\ s_i \in Q_j$ and $s_j \in Q_i,\ i \neq j,\}\{\emptyset\}$,
8.      THEN $V \leftarrow V - Q_i - s_i$, $FCH \leftarrow FCH \cup^{s_i}$ and $PCH \leftarrow PCH - s_j$.
9.      ELSE $V \leftarrow V - Q_i - s_i$ and $FCH \leftarrow FCH \cup^{s_i}$.
10. END WHILE.
11. Update $MsgQ_i$.
(Determine backup $CH$s in the neighbors of $CH$s)
12. Let $s_i \in FCH$ and $FBCH_i = \{\emptyset\}$ be a set of its formal backup $CH$s for $s_i$,
13. WHILE $|FBCH_i| < |CHL(i)|$,
14.     Find a SNs $s_k \in Ne_i$ such that $Ne_k$ has the least different from
        $Ne_i$ and set $\delta_i = Ne_i - Ne_k$,
15.     $FBCH_i = FBCH_i \cup^{s_k}$.
16.     Set the radius of $s_k$ so as to cover the SNs in $FBCH_i$.
17. END WHILE.
18. Append the set $FBCH_i$ in $MsgQ$ and broadcast the ID of $s_i \in FCH_i$ and
    their $MsgQ$ to all SNs in the network.
```

The *CHs* are determined by Algorithm *CHs_selection* performed by the BS. The BS receives the information from the potential *CHs* (*PCHs*) with maximum $|Me_i|$ among their neighbors, because they can receive packets from a large number of SNs from their neighbors. Therefore, one of those *PCHs* should become a formal *CH*. In lines 6–10, when a pair $s_i$ and $s_j$ are drawn from *PCH* and appear in the *MsgQ* of each other, the *PCH* with a larger number of neighbors is selected as a formal *CH*. Another SN will be drawn from *PCH* without checking, otherwise the other SNs are drawn one by one from *PCH* and the SNs in its *MsgQ* are drawn from set $V$ until $V = \{\emptyset\}$. After formal *CHs* are determined, BS performs the metric derived from Eq. (13) to determine the number of their required *BCHs*. Each *CH* determines the members of its *BCH* by referring to the information given in *MsgQ*. When determining a set of *BCHs* from the adjacent SNs of a *CH*, other member SNs of the *CH* may not be covered by the *BCHs* within the diameter of $H_{lower}$ hops. Therefore, the BCHs of the *CH* have to adjust their transmission radius so as to cover each other. When the *CH* is dead and replaced by one of the *BCHs*, other neighbors of the *CH* (i.e., the old *CH*) still use one hop to reach the *BCH* (i.e., the new *CH*). As a result, original members far from the *CH* in the opposite side to the *BCH* still have a maximum $H_{lower}$ hops to reach the *BCH*. In Fig. 5(a), for example, $n_1$ and $n_2$ are adjacent to the current (old) *ch*. When they are selected as the *BCHs*, say $bch_1$ and $bch_2$, they have to cover each other by expanding their transmission range from $r_1$ to $r_2$ as shown in Fig. 5(b), such that the members of $bch_2$ take the same hop count to reach $bch_1$ when *ch* is replaced by $bch_1$.

### 4.2. *Cluster_formation to construct inter-cluster routes*

Distributed algorithm *Cluster_formation* performed by SNs has two goals. First, when the SNs are informed by their *CHs*, the routes from *CHs* to BS are constructed by a proposed self-stabilizing algorithm. Second, when a member SN receives the broadcasted messages from algorithm *CHs_selection* performed by the BS, it acknowledges its *CH* or becomes a *BCH* that modifies its transmission radius and becomes dormant.

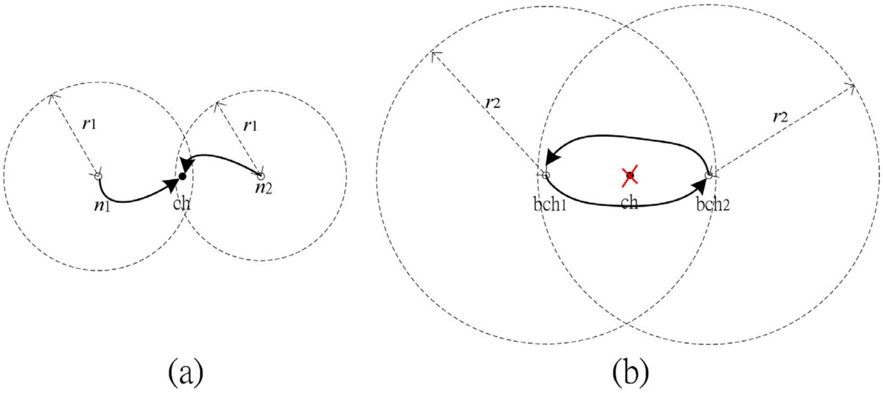


**Fig. 5.** Coverage adjustment by backup *CH*s.

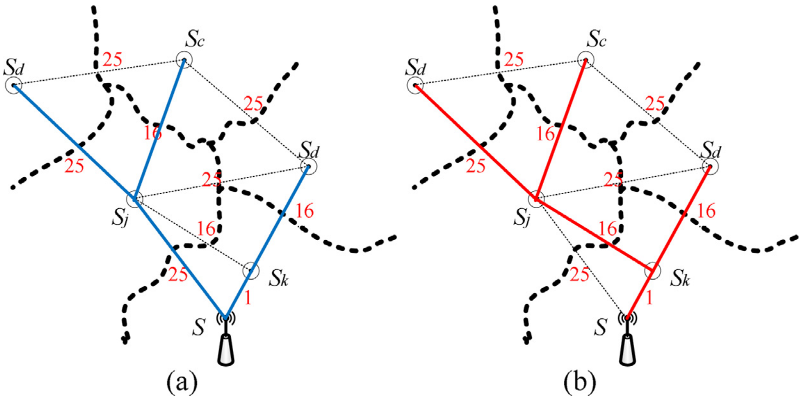


**Fig. 6.** An example of (a) MHT and (b) MCT.

To transmit aggregate data packets to BS in QoS fashion, *Cluster_formation* constructs energy-efficient routes and meets the tolerable delay threshold of individual data packets. First, the distributed self-stabilizing algorithm constructs a minimum cost spanning tree (MCT) and a minimum hop tree (MHT) among *CH*s and BS. The cost of MCT is regarded as the transmission radius applied by the *CH*s. For each *CH*, the algorithm also identifies its directions toward the BS. After building MCT and MHT, the second part of the algorithm determines the role of SNs and completes their configurations individually. In the next section, new protocols are proposed to convey aggregated packets along different paths in accordance with packet TD.

Let $G = (V,\ E)$ be a weighted biconnected graph where $V$ and $E$ respectively denote the set of vertex and edge. The weighting function $W(e_{u,v})$ denotes the length of edge $e_{u,v} \in E$ and each vertex is assigned a unique identifier. Every edge is associated with a nonnegative real number $W_T(e_{u,v}) = W^2(e_{u,v})$ according to Eq. (2) and represents the transmission power required between vertices $u$ and $v$. In Fig. 6, given a source SN $S$, the minimum cost paths between other *CH*s denote a MCT rooted at $S$. The minimum number of hops along the paths between *CH*s denote *MHT* rooted at $S$. Notations $MCT(i)$ and $MHT(i)$ respectively denote the distance of $CH_i$ to $s$ using *MCT* and *MHT*. The condition for applying *MCT* on a data packet must satisfy $AT_i \geq MCT(i)$ otherwise *MHT* is used. If neither *MHT* nor *MCT* are applicable to a data packet, it should directly be transmitted to BS.

For each vertex *i*, we define the following notations for the self-stabilizing algorithm in the biconnected graph.

$N(i)$: the set of adjacent vertices to *i*.
$d(i)$: the transmission cost of path from *i* to source.
$\ell(i)$: the distance of SN *i* to *s* in hops.
$mct\_p(i)$: the parent of *i* in *MCT*.
$mht\_p(i)$: the parent of *i* in *MHT*.

```
Algorithm Cluster_formation executed by CH and SNs.
When a message with $MsgQ_j$, $i = j$:
  InterCluster_MCT executed by BS or CH $s_i$.
  Phase I
  R1:if $i = s \wedge d(i) \neq 0$,
        then $d(i) = 0$ and $\ell(i) = 0$.
  R2:if $i \neq s \wedge d(i) \neq \mathrm{Min}\{d(j) + W(i,\ j) \mid jN(i)\}$,
        then $d(i) = \mathrm{Min}\{d(j) + W(i,\ j) | jN(i)\}$ and $\ell(i) = \ell(j) + 1$.
  Phase II
  R3:if $i = s \wedge \quad mct\_p(\mathrm{i}) \neq i$,
        then $mct\_p(i) = i$.
  R4:if $i \neq s \wedge\ j = \mathrm{Min}\{k \mid kN(i) \wedge\ d(i) = d(k) + W(i, k)\} \wedge\ mct\_p(i) \neq j$,
        then $mct\_p(i) = j$.
```

Algorithm *InterCluster_MHT* executed by BS or *CH* $s_i$

```
$\overline{\text{Phase I}}$
$\overline{\text{R1}}$: if $i{=}s \wedge \ell(i) \neq 0$,
        then $d(i) = 0$ and $\ell(i) = 0$.
$\overline{\text{R2}}$: if $i \neq s \wedge \ell(i) \neq \text{Min}\{\ell(j) |\ jN(i)\}$, $J$ denotes a set of $j$,
        then $\ell(i) = \text{Min}\{\ell(j) + 1 | jN(i)\}$ and
        $d(i) = Min\{d(h) + W(i, h) | h_{\in} J\}$.
$\overline{\text{Phase II}}$
$\overline{\text{R3}}$:if $i{=}s \wedge \quad mht\_\ p(i) \neq i$,
        then $mht\_p(i) = i$.
$\overline{\text{R4}}$:if $i \neq s \wedge\ j = \text{Min}\{k \mid kN(i) \wedge \ell(i) = \ell(k)d(i) = d(k)\} \wedge mht\_p(i) \neq j$,
        then $mht\_p(i) = j$.
```

```
Set $s_i \propto s_j$ and compute $mct\_B_{i,j}$ and $mht\_B_{i,j}$ in accordance with
$Sensor\ Q_i \cap Sensor\ Q_j$
Multicast $mct\_B_{i,j}$ and $mht\_B_{i,j}$ to the $s_k \in FBCH_i$.
```

```
When an message with $MsgQ_j$, $i{\neq}j$:
  Let $min\_acc\_r = \infty$, $max\_hc = 0$.
  IF $s_i$ has been notified as a $CH$,
      THEN EXIT this procedure.
  IF the ID of $s_i$ does not exist in $MsgQ_j$,
      THEN discard the message.
  IF $s_i \in FBCH_j$,
      THEN $s_i$ switch to dormant mode.
  ELSE
      In accordance with the values of $acc\_r_i$ and $hc_i$ in $MsgQ_j$,
      Set $min\_acc\_r = acc\_r_i$ and $max\_hc = hc_i$.
      Set $s_j$ be the $CH$ of $s_i$.
  END IF
```

The self-stabilizing algorithm *InterCluster_MCT* executes in two phases. For each $CH_i$, it calculates the length of the minimum-cost path from $CH_i$ to $s$ during Phase I. In Phase II, $mct\_p(i)$ is derived for each $CH_i$. In the algorithm *InterCluster_MHT*, Phase I calculates the hop-count along the path with the minimum number of hops. If multiple paths having the minimum hop-count, the path with the minimum cost is selected. In Phase II, a neighbor of $CH_i$ with the minimum hop-count to BS is denoted as $mht\_p(i)$. Assume a distributed demon randomly selects a subset of privileged vertices (i.e., *CH*s) and moves simultaneously. Fig. 7 shows five configurations that illustrate the execution of the algorithms. The shaded nodes represent the privileged nodes, and shaded nodes with a bold circle denote the privileged nodes selected by the distributed demon to make a *system move* [29].

The proofs of Phase I and Phase I are similar to those in [29], Phase II is as follows, and Phase II can be proved similarly.

**Lemma 1.** *PhaseIstabilizes in finite steps.*

**Proof.** In [29], Huang has already proved this Lemma. □

**Lemma 2.** *After PhaseI stabilizes, PhaseII can stabilize in finite time.*

**Proof.** For node $s$, it spends finite time to set the variable $p(s)$ to itself. For any node $i \neq s$, we assume that node $j$ has the minimum identifier of the neighbors of $i$ and satisfies $d(i) = d(j) + w(i,\ j)$. Due to finite number of neighbors of $i$, node $i$ can choose $j$ and set the variable $p(i)$ to $j$ in finite time. Since each node $k$ spends finite time to set the variable $p(k)$ of itself and at most waits for all the other nodes finish the same kinds of action (the number of nodes is finite), the lemma follows. □

### 4.3. Convey aggregate data in hop-constrained fashion

In cluster-based WSN, data packets can be classified into two categories. The sensing data (*SD*) packets presented in Fig. 8(a) are produced by member SNs. The aggregate data (*AD*) packet shown in Fig. 8(b) produced by *CH* $s_i$ is denoted as $AD_i$, where $s_i$ denotes its source *CH*. When $AD_i$ are produced by a *CH* $s_i$, they will be assigned a new value of $AT_i$. Before constructing the routes between *CH*s and the BS, *CH*s have to recognize the direction to BS and the sets of SN on the individual *cluster border* toward this direction. The notion of *cluster border* for *mcb_B* and *mhb_B* is illustrated in Fig. 9.

To transmit *AD* packets to the BS in a number of hops not greater than their individual *AT*, we propose three transfer modes to adapt their QoS requirements. Moreover, a packet routing between *CH*s and BS can apply those transfer modes interchangeably. The first mode is called *hitchhike* mode, with an example route shown in Fig. 9(a) denoted by dotted lines. *CH A* transmits an *AD* packet to BS by attaching it on a sensing data packet that will be transmitted to *CH B*. Due to asymmetric transmission in opposite directions, SN $y$ can reach SN $x$ but not necessarily vice versa. SNs always send their data packets in the direction of *CH* or BS while packet transfer to the opposite direction is infrequent and usually energy-inefficient. Therefore, in the *hitchhike* mode, *CH A* first conveys *AD* packets to the SNs in the set on the *border* between itself and its parent *CH* on *MHT* or *MCT* spanning trees. Those SNs are called "*bridges*" of border $mht\_B_{A,B}$ or $mct\_B_{A,B}$ between clusters of $A$ and $B$, which are stochastically selected SNs with higher residual energy. The hops required by *CH A* using *hitchhike* is modeled as

$$d(A) + (d(A) - 2) \times (H_{lower} - 1), \tag{14}$$

where $d(A)$ denotes the depth of *CH A* in an *MHT* or *MCT* tree.

The second mode is called *infrastructure* mode which enables a *CH* to send aggregate data directly to its parent *CH* in an *MHT* or *MCT* tree. By using self-stabilizing algorithms *InterCluster_MHT* and *InterCluster_MCT* in Section 4.2, each *CH A* knows its parent, and its distance to the BS takes $d(A)$ hops. In Fig. 9(b), the infrastructure route is presented as dashed lines. The third mode is called *prompt* mode, in which each *CH* transmits aggregate data directly to the BS. As shown in Fig. 9(b), a bold solid arc denotes the route in prompt mode. When an *AD* packet arrives at or is produced by *CH A*, it can be transferred directly to the BS when its $AT \leq 1$, or a hybrid of those modes can be applied to the BS depending on its current *AT* value. Hybrid transmission due to routing decision is performed by lines 4–17 and lines 18–26 in *InterCluster_Routing*.

Algorithm *InterCluster_Routing* executed by SN $s_i$.

```
On receiving aggregate data packet $AD_x$ with:
1.  Let $CH\ s_k = mct\_p(j)$ or $s_k = mht\_p(j)$.
2.  IF $s_i$ is a member of CH $s_k$ and $\{s_i | \exists s_b \in mht_{B\,j,k} \vee mct_{B\,j,k},$
    $s_b \in MsgQ_i\} \neq \emptyset$
3.      THEN store the packet in the cache.
4.  ELSE IF $i = j$,
5.          IF $AT_x \geq H_{lower} + d(i)$,
6.              THEN transmit the aggregate packet to a SN in $mct\_B_{j,k}$.
7.          ELSE IF $AT_x \geq H_{lower} + \ell(i)$,
8.              THEN transmit the aggregate packet to a SN in $mht\_B_{j,k}$.
9.          ELSE IF $AT_x \geq d(i)$,
```

```
10.            THEN transmit the aggregate packet to CH $mct\_p(i)$.
11.         ELSE IF $AT_x \geq \ell(i)$,
              THEN transmit the aggregate packet to CH $mht\_p(i)$.
12.         ELSE transmit the aggregate packet direct to BS with $r_y \leq r_{max}$.
13.  ELSE IF $i = k$,
14.       THEN transmit the packet to a SN in $mht\_B_{k,m}$, $k \neq m$.
15.  ELSE discard $AD_x$.
16.  $AT_x --$.
```

**On reporting aggregate data to BS**

```
17.  Generate aggregate data packet and determine its hop count limit
     $AT_x$ to BS
18.  IF $AT_x \geq H_{lower} + d(i)$,
19.       THEN transmit the aggregate packet to a SN in $mct\_B_{j,k}$.
20.  ELSE IF $AT_x \geq H_{lower} + \ell(i)$,
21.       THEN transmit the aggregate packet to a SN in $mht\_B_{j,k}$.
22.  ELSE IF $AT_x \geq d(i)$,
23.       THEN transmit the aggregate packet to CH $mct\_p(i)$.
24.  ELSE IF $AT_x \geq \ell(i)$,
         THEN transmit the aggregate packet to CH $mht\_p(i)$.
25.  ELSE transmit the aggregate packet direct to BS with $r_y \leq r_{max}$.
26.  END IF
```

According to the locations of SN $s_i$, there are three cases discussed in *InterCluster_Routing*.

Case 1. $s_i \notin mht_{B\,j,k} \vee mct_{B\,j,k}$, and $s_i$ are members of $s_j$: In Fig. 10(a), $s_j$ transmits $AD_x$ directly to the members of $mht_{B\,j,k} \vee mct_{B\,j,k}$, and $s_i$ discards $AD_x$ if it receives $AD_x$. There are two reasons for $s_i$ not forwarding $AD_x$. Firstly, in the WSN, all SNs are designed to transmit their data to their *CH*, and transmission in the opposite direction from a *CH* to its SNs is undesirable in that it would increase the design complexity of SNs and the WSN protocol. Secondly, if those $s_i$ must forward $AD_x$ from *CH*, coordination among $s_i$s would incur additional network overhead.

Case 2. $s_i$ is a member of CH $s_k$: In Fig. 10(b), $s_i$ can fully bring its function that hitchhikes $AD_x$ with sensing data to their *CH*.

Case 3. $s_i$ is a CH: In Figs. 10(c) and (d), $s_i$ forwards $AD_x$ to the SNs in $mht_{Bi,k} \vee mct_{Bi,k}$ so as to *hitchhike* $AD_x$ to *CH* $s_k$ or the next *CH* toward the BS.

When a bridge SN has energy level $e$ lower than a threshold $Th > 0$, it signals an *Energy_Urgent* to the SNs in the same set. Those SNs that receive the *Energy_Urgent* and have an energy level of $1 \geq e \geq Th$ will bid for the bridge of $mht\_B_{j,k}$ or $mct\_B_{j,k}$. In line 5 of algorithm *IntraCluster_transmission*, an SN becomes a bridge in a probability of $\frac{i}{|mht\_B_{j,k}|} \times e$ or $\frac{i}{|mct\_B_{j,k}|} \times e$.

Algorithm *IntraCluster_transmission* executed by SN $s_i$.

**On forwarding a sensing data packet $SD_k$**

```
1.  IF CHID of $SD_k$ is not the CH of $s_i$ OR $AT_k < 1$,
2.     THEN discard $SD_k$.
3.  ELSE,
      forward $SD_k$ together with $SD_i$ produced by $s_i$.
```

**On reporting a sensing data to *CH*:**

```
4.  IF $s_i \in mht_{B\,j,k}(mct\_B_{j,k})$ and received an Eergy_Urgent message,
5.       THEN bid for a bridge of $mht\_B_{j,k}$ in probability of
         $\frac{i}{|mht\_B_{j,k}|} e$ ($\frac{i}{|mct\_B_{j,k}|} \times e$).
6.  IF $s_i$ is a bridge and its energy is lower than threshold $Th$,
7.       THEN multicast an Eergy_Urgent message to $mht\_B_{j,k}$ ($mct\_B_{j,k}$).
8.  IF cache is not empty,
9.       THEN hitchhike the cache content with sensing data to the next
         SN.
10. ELSE
11.      Send sensing data to the next SN.
END IF.
```

In Fig. 8(a), the *SD* packets produced by SN $s_i$ are $SD_i$ in which field *source* and *CHID* respectively represent the starting SN and destination *CH*. In *IntraCluster_transmission*, each *SD* packet is forwarded by SNs if their producers and *FWD*s are in the same cluster. In other words, a *SD* packet is kept alive in the network as long as it stays in the range of its original cluster.

In accordance with the proposed inter-cluster routing algorithm, the energy dissipation model for conveying aggregate data packets should be modified so as to derive a precise number of required *BCH*s and a better network lifetime. In Eq.(9), each *CH* receives a total $m$ SD packets per hour and receives $|Q_c|$ packets in each round, where $|Q_c| \leq m$ is the number of member SNs in cluster $c$. The number of clusters in the network is $\frac{n}{|Q_c|}$ and we derive

$$m_{AD} = \frac{n}{|Q_c|} \times \frac{m}{|Q_c|} = \frac{n \times m}{|Q_c|^2} \tag{15}$$

*AD* packets per hour. The average height of *MCT* and *MHT* is $\mathcal{L}$. Notations $\delta_{prompt}$, $\delta_{pig}$ and $\delta_{infra}$ respectively denote the proportion of *AD* packets applying *prompt*, *hitchhike* and *infrastructure* mode to $m$ *AD* packets, such that $\delta_{prompt} + \delta_{pig} + \delta_{infra} = 1$. The energy required by *CH* to transmit an *AD* packet in the prompt mode is derived from Eq.(11) and

$$E_{prompt} = \lambda E_{elec} + \lambda \varepsilon_{fs} r_{max}^2 \tag{16}$$

where $\lambda$ is the length of the packet in bits. Let the square of the average distance between $CH_i$ and $mct\_B_{i,k}$ or $mht\_B_{i,k}$ be

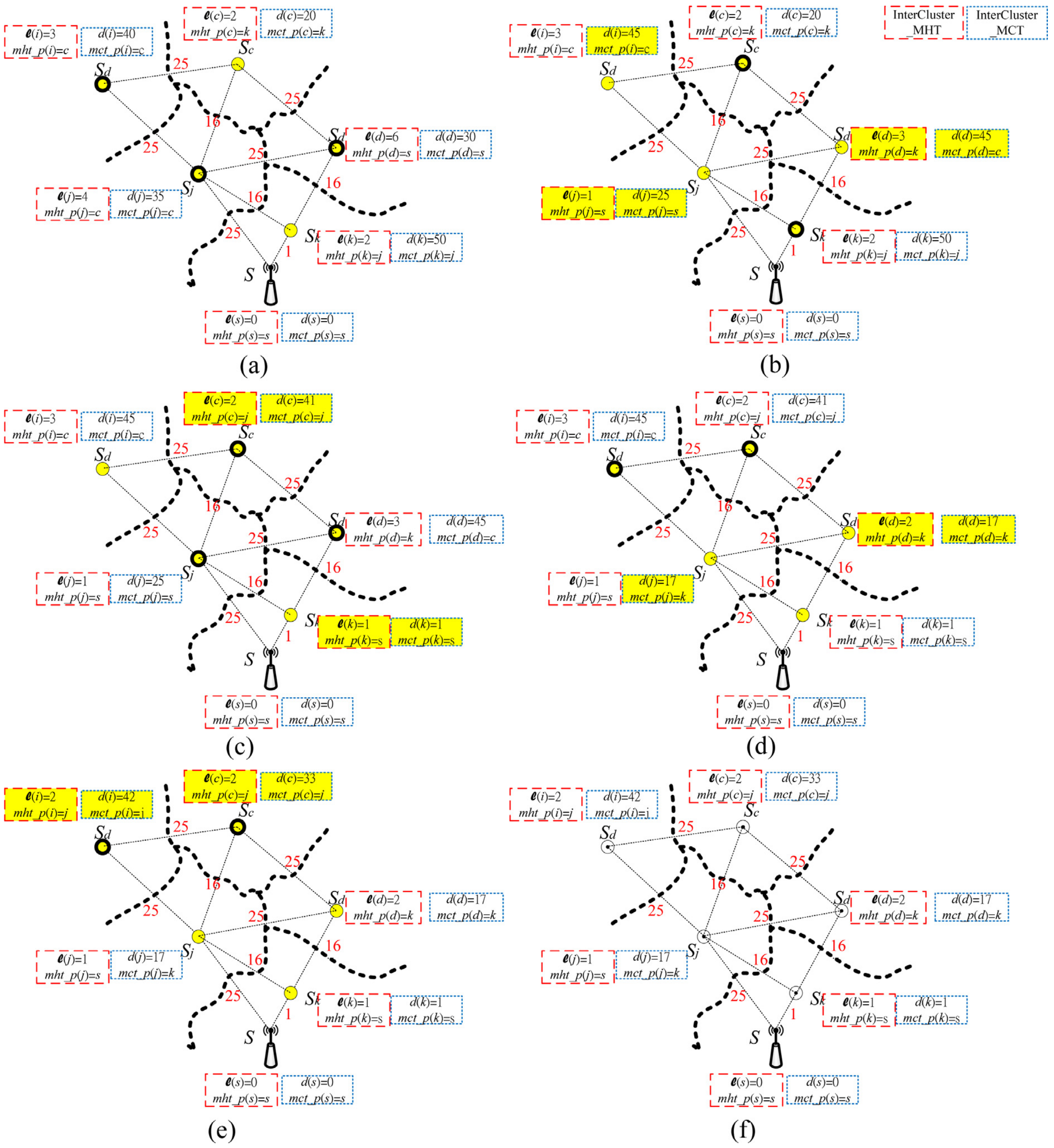
$$r_{pig}^2 = \begin{cases} \dfrac{\sum_{\forall e_{i,k} \in \alpha_{mct}} \left( \frac{W(e_{i,k})}{2} \right)^2}{E_{mct}}, & e_{i,k} \text{ in } MCT \\ \dfrac{\sum_{\forall e_{i,k} \in \alpha_{mht}} \left( \frac{W(e_{i,k})}{2} \right)^2}{E_{mht}}, & e_{i,k} \text{ in } MHT. \end{cases} \tag{17}$$

where $\alpha_{mct}$ and $\alpha_{mht}$ respectively denote the sets of edges in the *MCT* and *MHT*. In accordance with Eqs. (11), (14) and (17), the energy required for transmitting a $\lambda$-bit *AD* packet from *CH* $s_i$ to BS by *hitchhike* mode is derived as

$$E_{pig}(i) = \begin{cases} \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_{pig}^2\right) \times d(i) \\ \quad + (d(i) - 2) \times H_{lower} \times \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_c^2\right), \\ \quad s_i \text{ use } MCT \\ \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_{pig}^2\right) \times \ell(i) \\ \quad + (\ell(i) - 2) \times H_{lower} \times \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_c^2\right), \\ \quad s_i \text{ use } MHT. \end{cases} \tag{18}$$

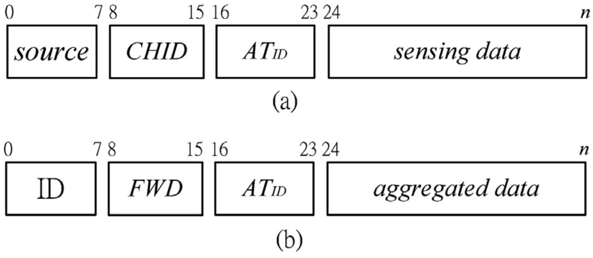
Let the edge $e_{u,v}$ induce *CH*s $s_u$ and $s_v$, and $W(s_u, s_v)$ be the length of $e_{u,v}$, the square of average length between adjacent *CH*s is

$$r_{infra}^2 = \begin{cases} \dfrac{\sum_{\forall e_{u,v} \in \alpha_{mct}} (W(e_{u,v}))^2}{E_{mct}}, & e_{u,v} \text{ in } MCT \\ \dfrac{\sum_{\forall e_{u,v} \in \alpha_{mht}} (W(e_{u,v}))^2}{E_{mht}}, & e_{u,v} \text{ in } MHT. \end{cases} \tag{19}$$



(a) (b) (c) (d) (e) (f)

**Fig. 7.** The execution example of self-stabilizing algorithms *InterCluster_MCT* and *InterCluster_MHT*.



(a)

(b)

**Fig. 8.** The format of (a) *SD* and (b) *AD* packets.

In the infrastructure mode, the energy required to transmit a $\lambda$-bit *AD* packet from *CH* $s_i$ to BS is denoted as

$$E_{infra}(i) = \begin{cases} \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_{infra}^2\right) \times d(i), & s_i \text{ use } MCT \\ \left(\lambda E_{elec} + \lambda \varepsilon_{fs} r_{infra}^2\right) \times l(i), & s_i \text{ use } MHT. \end{cases} \tag{20}$$

Eqs. (16), (18) and (20) give the average energy for transmitting an $\lambda$-bit *AD* packet. When the proportions of $\delta_{prompt}$,

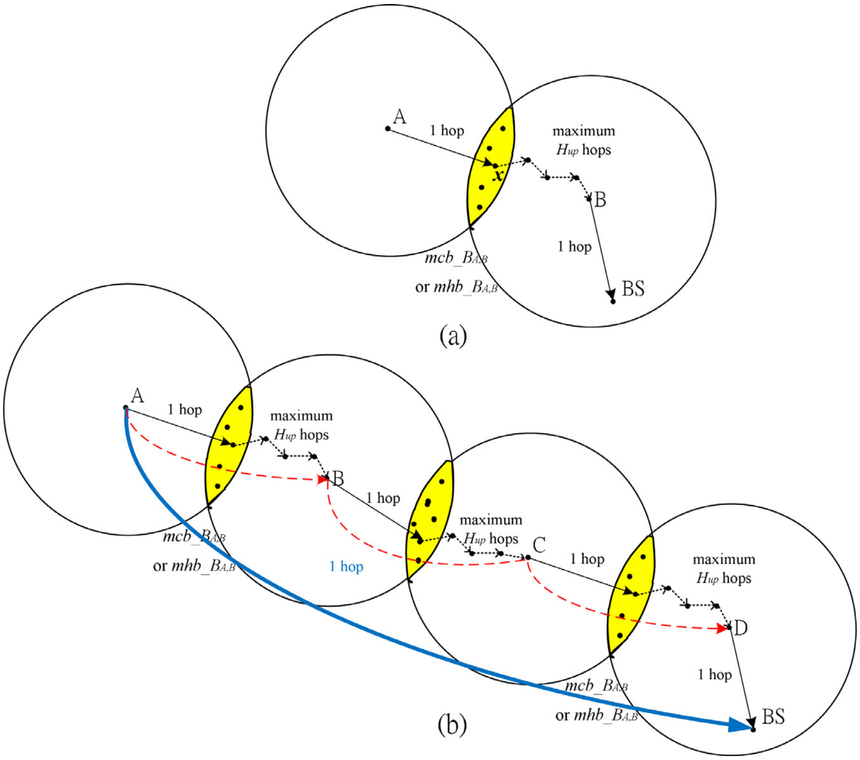


**Fig. 9.** The (a) hitchhike and (b) a hybrid transfer modes.

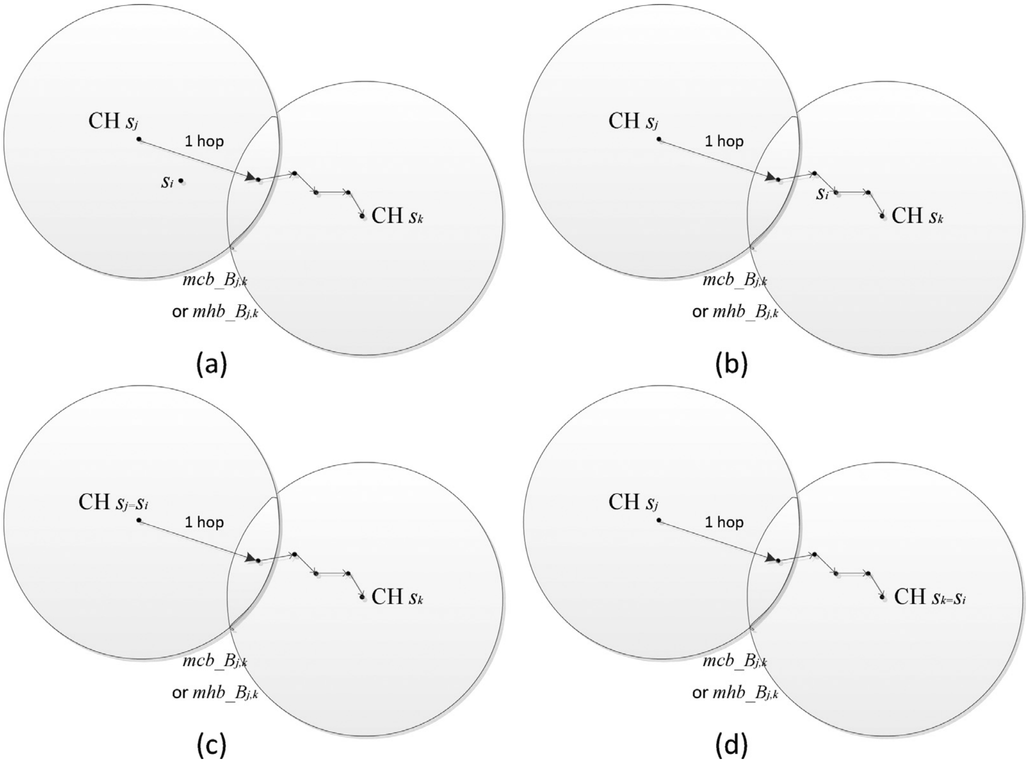


**Fig. 10.** The location of SN $s_i$.

$\delta_{pig}$ and $\delta_{infra}$ are given, we can derive

$$E_{CHT}(R) = (\delta_{prompt} \times E_{prompt}) + \left(\delta_{pig} \times E_{pig}(R)\right) + \left(\delta_{infra} \times E_{infra}(R)\right) \tag{21}$$

where *CH* $s_R$ denotes a reference point having the average distance $\mathcal{L}$ from the BS. Therefore,

$$E_{CH\_new} = (mE_{elec} + mE_{DA}) + (m_{agg} \times E_{CHT}(R)) \tag{22}$$

denotes the energy consumption of a *CH* in an hour. When we use the proposed inter-cluster routing protocol, the expected lifetime of a *CH* is

$$CHL_{new} = \frac{E_{bat}}{E_{CH\_new}}, \tag{23}$$

and can replace the value of $|CHL(z)|$ in Eq. (13).

We prove the packet hop constrains are met by the proposed protocol as follows.

**Lemma 3.** *For all sensing data packets* $\boldsymbol{SD_i}$ *with* $\boldsymbol{AT_i} > \boldsymbol{H_{lower}} + 1$ *in SN* $\boldsymbol{s_i}$*, when they arrive at CH, then* $\boldsymbol{AT_i} > 1$.

**Proof.** Before sending the $SD_i$ packet, $s_i$ has already multicasted an exploring message $EM_i$ with $AT_i = H_{lower} - 1$ in the *offline* stage. Given that $EM_i$ reaches SN $s_k$ with $AT_i = 1$, it can be recorded by $s_k$. If $s_k$ becomes a *CH*, it also receives $SD_i$. By definition, $AT_i$ must be greater than 1. In accordance with line 1 of *IntraCluster_transmission*, $SD_i$ is not discarded by $s_k$, and this completes the proof. □

**Lemma 4.** *Given an aggregate data packet* $\boldsymbol{AD_k}$ *with* $\boldsymbol{AT_k} > 1$*, it will arrive at the BS with* $\boldsymbol{AT_k} > 0$.

**Proof.** The proof is achieved in accordance with lines 5–12 in algorithm *InterCluster_routing*, and divided into four cases.

**Case 1.** When $AD_k$ is forwarded by *CH* $s_x$ to a border of $mct\_B_{x,y}$ with $AT_k > H_{lower} + d(x)$, it takes at most $H_{lower}$ hops to reach *CH* $s_y$ because, by definition and Lemma 3, the border is a member SN of $s_y$. When $AD_k$ arrives at $s_y$, we have $AT_k = d(x)$. Because $s_x$ and $s_y$ are adjacent to *CH* on the *mct* tree, we derive $d(x) - d(y) = 1$. Therefore, $AD_k$ can be transferred solely using the infrastructure mode to the BS and $AT_k = 1$.
**Case 2.** When $AD_k$ is forwarded by $s_x$ to a border of $mht\_B_{x,y}$ with $H_{lower} + \ell(x)$, this proof is similar to Case 1.
**Case 3.** The cases of $AT \geq d(x)$ and $AT_k \geq \ell(x)$ is already proved by Case 1 and Case 2, respectively.
**Case 4.** When $AT_k = 2, AD_k$ is transferred by $s_y$ to BS by *prompt* mode, resulting in $AT_k = 1$ and this completes the proof. □

**Theorem 5.** *For all sensing data packets with hop constrains greater than* $\boldsymbol{H_{lower}} + 1$*, the aggregate data packet can arrive at the BS before their* $AT = 0$.

**Proof.** The theorem can be proved by Lemma 3 and Lemma 4. □

## 5. Performance |evaluation

### 5.1. Execution time complexities

The algorithms of the SHE framework shown in Fig. 3 are classified in Table 1. The *online* algorithms are performed during the network operation time, while the *offline* algorithms are performed only once at the beginning of network initialization. The *offline* algorithms can be further divided into *distributed* and *centralized* algorithms. The *distributed* algorithms are performed by all SNs while the *centralized* algorithm is performed only by the BS.

**Table 1**
Time complexities of SHE.

| Location | Duration | |
|---|---|---|
| | Offline | Online |
| Centralized | *CHs_selection* $O(n^2)$ | |
| Distributed | *MultiHopCasting* $O(n+t)$ | *InterCluster_routing* $O(1)$ |
| | *Cluster_formation* $O(n)$ | *IntraCluster_transmission* $O(1)$ |

**Lemma 6.** *The time complexity of the centralized offline algorithm is O(n), where n denotes the number of SNs in the network.*

**Proof.** In Eq. (13), $|BCH_i|$ is associated with the value of $CHL_i$ depending on the battery and energy overheads of the SNs. The overhead arising from the distribution of the SNs dominates the energy consumption in the network. The first part of *CHs_selection* finds suitable SNs as formal *CH*s taking $O(|PCH|)$ time complexity and $|PCH| < n$ where $n$ denotes the number of SNs in the network. The second part of the algorithm computes the number of *BCH*s depending on the number of neighbors of the *BCH*s. In the worst case, the number of backup *CH*s and their neighbors is as many as $n$, and thus its complexity is $O(n)$. □

**Lemma 7.** *The time complexity of the distributed offline algorithm is* $O(n+t)$ *where t denotes the number of available transmission radii.*

**Proof.** For *MultiHopCasting*, the time complexity is analyzed as follows: Line 1 searches for a suitable radius for broadcasting *EM* messages and takes $O(t)$ time, where $t$ is the number of available transmission ranges. In addition, the insertion of *MsgQ* in line 9 requires $O(n)$, because the size of *MsgQ* is as large as the number of nodes in the network in the worst case. In line 14, the search in *MsgQ* and its neighbors both requires $O(n)$. Therefore, the time complexity of *MultiHopCasting* is $O(n+t)$. For *Cluster_formation*, the analysis is divided into two parts. Firstly, Phase I of the self-stabilizing algorithm requires $O(n)$ time complexity [29]. In the worst case, each node in the network is privileged when executing the rules of Phase II, and the distributed demon always chooses one privileged node to move. Therefore, Phase II will stabilize after performing $O(n)$ received messages in *MsgQ* and this completes the proof. □

**Lemma 8.** *The time complexity of the distributed online algorithms is O(1).*

**Proof.** The dominant actions performed by algorithms *InterCluster_routing* and *IntraCluster_transmission* are composed of a series of simple IF-ELSE statements about transmitting packets. They take $O(1)$ time complexity. □

In each set-up stage of the previous protocols, the cluster formation process has to be performed repeatedly during the lifetime of the network. In SHE, the higher-complexity protocols *MultiHopCasting*, *CHs_selection* and *Cluster_formation*

**Table 2**
Experimental settings.

| Experiment parameters | Values |
|---|---|
| Square | 100 m × 100 m, 200 m × 200 m |
| Total SNs | 50, 200 |
| Position of base station | (200, 200) |
| Initial energy capacity per SN | 18720 J (two AA batteries) |
| Baseline energy ($E_0$) | 0.5 J |
| Communication energy ($E_{elec}$) [24] | 50 nJ/bit |
| Free space energy ($\varepsilon_{fs}$) [24] | 10 pJ/bit/m$^2$ |
| Multipath energy ($\varepsilon_{mp}$) [24] | 0.0013 pJ/bit/m$^2$ |
| Packet length | 1000 bits (max.) |
| Beacon length | 80 bits |
| Proportions of $\delta_{prompt}$, $\delta_{infra}$, $\delta_{pig}$ | 20%, 30%, 50% |
| Energy threshold for bridge SNs ($Th$) | 30% |

are performed only once at the network initiation, while the online algorithms, *InterCluster_routing* and *IntraCluster_transmission*, requires only O(1) time. Therefore, the *distributed online* algorithms incur low computational overhead in the networks.

### 5.2. Simulation results

In this section, we compare the performance of SHE with those of LEACH [23], LEACH-C [24], M-HEED [12,15], PEGASIS [25] and HFB [15]. Traditional performance metrics, *network lifetime* and *energy consumption*, are used for comparison. The *network lifetime* is defined as the duration from network initialization to the instant when a given amount or percentage of SNs die. The *energy consumption* is defined as the sum of the residual energy of all of the SNs. In addition, new performance metrics are also introduced. The *number of clusters* defines the average number of clusters required by the network. The metric shows the performance of the clustering decisions made by underlying methods. A multi-hop method having a lower number of required clusters provides better energy efficiency and relatively undisturbed service due to fewer cluster re-configurations. *Hop-constrained missed packets* are defined as the number of constraint-missed packets transmitted by the underlying methods. Section 5.1 proves that it can be applied to various hop-constrained packets in the manner of QoS. In simulations, a *round* is defined as the BS receiving data from all of the SNs.

The experiments are performed by C++ and OMNet++® [33], and primary assumptions for simulation environment are as follows:

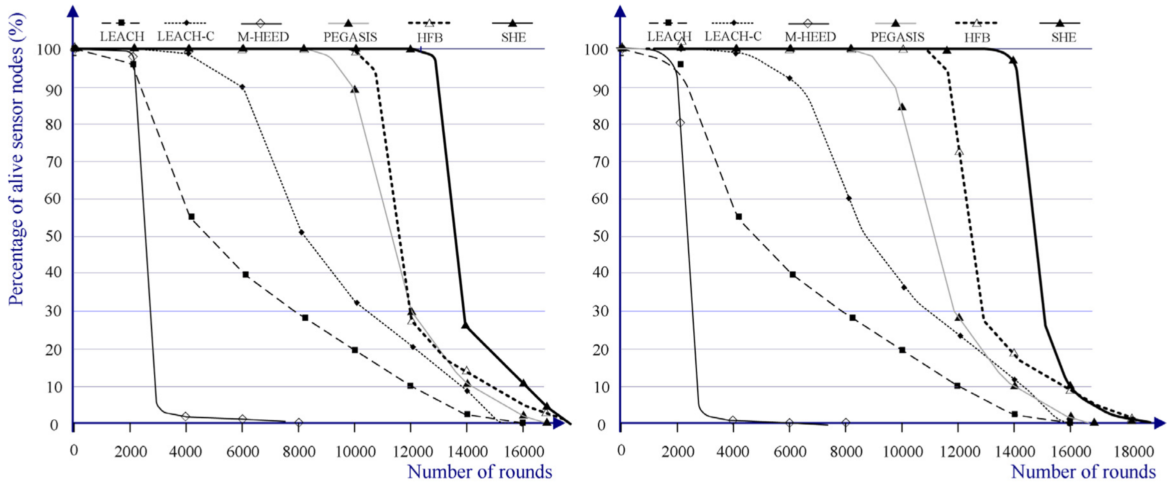
1. All SNs are deployed in a square using a random uniform distribution.
2. BS equipped with hi-speed CPU, enough memory space and power source is immobile and located at a corner of the square.
3. Each SN can send data to the BS.
4. Each *CH* can gather, compress and forward data to the BS or another *CH*.
5. Radio model is assumed to comply with the specification of CC2420 [28] providing 8 possible output power settings at a 1.8 V DC supply voltage and 2.54 GHz.

Tx in dBm: 0 −1 −3 −5 −7 −10 −15 −25
Tx in mW: 1 0.794 0.501 0.316 0.199 0.1 0.031 0.003

The parameter values used in the simulations are shown in Table 2.

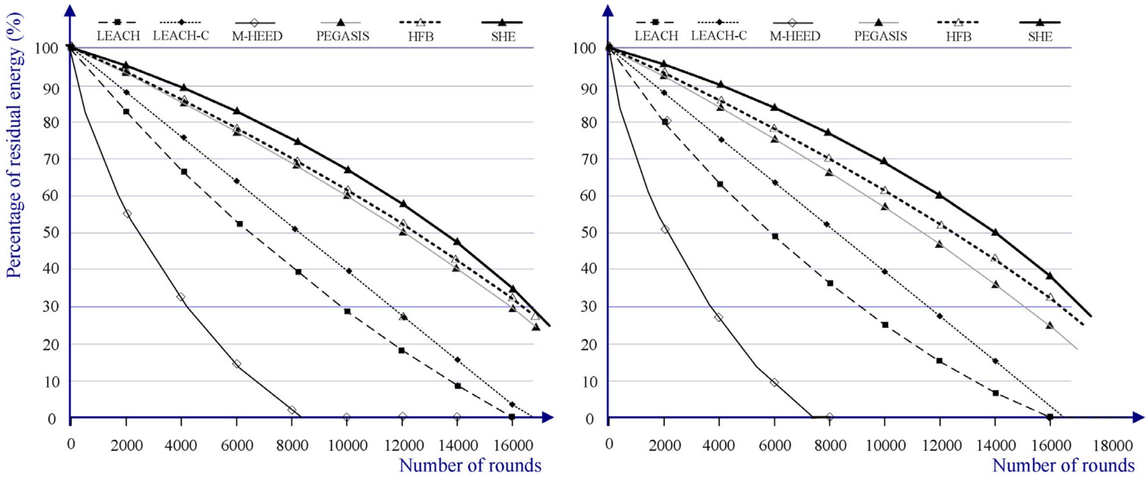
Fig. 11(a) compares the network lifetime of the proposed method with LEACH, M-HEED, PEGASIS and HFB give 50 SNs. We see from the figure that SHE outperforms other previous methods in terms of when the first SN died, and outperforms PEGASIS by approximately 20%. In addition, in Fig. 11(b) we extend the sensing area to a 200 m × 200m square in the simulation setting. FBR outperforms other previous methods because PEAGASIS is a chain-based protocol. That is, the packet forwarding overhead increases significantly as the length of its routing paths increases. Fig. 11 shows that SHE achieves a longer network lifetime than previous methods.

Fig. 12 compares the energy dissipation of the proposed method with that achieved by previous methods. We can see from Fig. 12(a) that the residual energy of PAGASIS is



(a) (b)

**Fig. 11.** Network lifetimes of various protocols in (a) 50 and (b) 200 SNs.



(a)

(b)

**Fig. 12.** Residual energy of various protocols in (a) 50 and (b) 200 SNs.

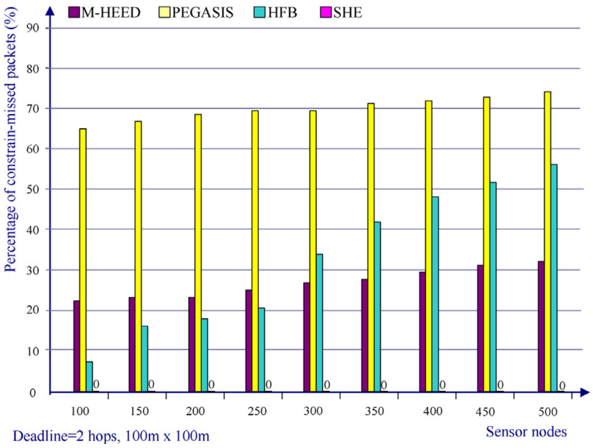


**Fig. 13.** Constrain-missed packets in various number of SNs.

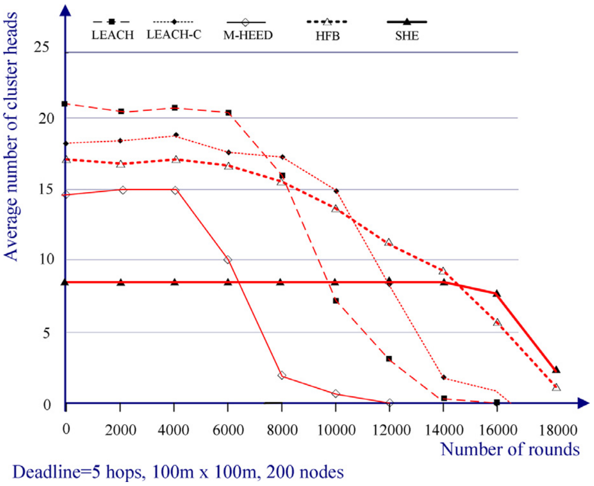


**Fig. 14.** The number of required *CH*s.

higher than other previous methods while in Fig. 12(b) HFB outperforms PAGASIS in a larger-scaled WSN. This finding occurs because HFB provides shorter routing paths between SNs and the BS resulting in less overhead for forwarding packets than that in PEGASIS. Furthermore, it is evident that the proposed method has a higher residual energy than other methods due to constrained multi-hop and *hitchhike* routing strategies. Figs. 12(a) and (b) show that most multi-hop methods such as PEAGSIS, HFB and SHE outperform 2-hop methods in terms of network lifetime and energy consumption.

Fig. 13 shows the percentage of constrain-missed packets of the protocol under consideration given varying numbers of SNs in the network. LEACH and LEACH-C are excluded from the simulation because they are 2-hop clustering protocols. A *hop-count* of a packet in the simulation counts both the number SNs and *CH*s that the packets traverse. In other words, it counts a packet from its source SN to BS. For example, when the hop-count constraint on each packet is 5, HFB produces 21% constraint-missed packets. The proposed method in the simulation can be a yardstick against other methods because it is a hard hop-constrained routing in hops.

SHE primarily elects *CH*s according to their *MsgQ*, which is independent of node distribution because the *MsgQ* of each node is composed of a set of *EM* messages sent by the reachable SNs in a predefined hop constraint. Based on this, BS determines the sets of *CH*s and backup *CH*s for the clusters and reduces the number of required clusters. In addition, previous distributed clustering methods elect *CH*s according to their neighbors and residual energy, resulting in *CH*s which may be quite close to each other within the cluster coverage. Fig. 14 shows that the number of required *CH*s is much lower than in previous methods.

## 6. Conclusions

This paper proposes a self-stabilizing hop-constrained energy-efficient (SHE) clustering and routing protocol for multi-hop wireless SN networks. SHE is hybrid: *CH*s are deterministically elected by an offline algorithm performed by the BS and routing decisions are made by online and distributed algorithms. The important features of SHE are as follows. First, the real-time packet routing is multi-hop in both intra-cluster and inter-cluster transmission. On the basis of the distributed network architecture, the number of required clusters and transmission power of SNs can be reduced. Secondly, data packets can be transferred in the most energy efficient way while guaranteeing their hard deadlines (TD) in hops by using the transfer modes of hitchhike, *infrastructure* and *prompt* interchangeably. Therefore, it provides a QoS feature in terms of hop constraints of real-time data packets. Simulation results show that SHE achieves lower power dissipation and longer network lifetime than previous methods. In future, the proposed model will be extended to accommodate the heterogeneous sensor nodes with various energy capacities, RF model, power consumptions and communication capabilities. In addition, we plan to extend our method with fault diagnosis and fault tolerance capabilities.

## Acknowledgments

The authors would like to thank the Ministry of Science and Technology of Republic of China, Taiwan, for financially supporting this research under MOST 102-2221-E-025-003 and MOST 104-2221-E-025-002.

## Reference

[1] D. Li, K.D. Wong, Y.H. Hu, A.M. Sayeed, Detection, classification, and tracking of targets, Signal. Proc. Mag. IEEE 19 (2) (2002) 17–29.

[2] Y. Durmus, A. Ozgovde, C. Ersoy, Distributed and online fair resource management in video surveillance sensor networks, Mobile Comput. IEEE Transac. 11 (5) (2012) 835–848.

[3] H. Modares, R. Salleh, A. Moravejosharieh, Overview of security issues in wireless sensor networks, in: Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM), 2011, IEEE, 2011, September, pp. 308–311.

[4] P.N. Reddy, P.I. Basarkod, S.S. Manvi, Wireless sensor network based fire monitoring and extinguishing system in real time environment, Int. J. Adv. Netw. Appl. 3 (02) (2011) 1070–1075.

[5] F. Wang, J. Liu, Networked wireless sensor data collection: issues, challenges, and approaches, Commun. Surv. Tutor. IEEE 13 (4) (2011) 673–687.

[6] J.Y. Chang, P.H. Ju, An efficient cluster-based power saving scheme for wireless sensor networks, EURASIP J. Wirel. Commun. Netw. 2012 (1) (2012) 1–10.

[7] K.S. Prabh, Real-time Wireless Sensor Networks, Doctoral Dissertation, University of Virginia, 2007.

[8] P. Suriyachai, J. Brown, U. Roedig, Time-critical data delivery in wireless sensor networks, Distributed Computing in Sensor Systems, Springer, Berlin Heidelberg, 2010, pp. 216–229.

[9] P. Suriyachai, U. Roedig, A. Scott, A survey of MAC protocols for mission-critical applications in wireless sensor networks, Commun. Surv. Tutor. IEEE 14 (2) (2012) 240–264.

[10] Y. Gu, T. He, Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks, Mobile Comput. IEEE Transac. 10 (12) (2011) 1741–1754.

[11] Ossama Younis, Sonia Fahmy, Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach, in: INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies., 1, IEEE, 2004.

[12] Seema Bandyopadhyay, EdwardJ. Coyle, An energy efficient hierarchical clustering algorithm for wireless sensor networks, in: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies., 3, IEEE, 2003.

[13] V. Rajendran, K. Obraczka, J.J. Garcia-Luna-Aceves, Energy-efficient, collision-free medium access control for wireless sensor networks, Wirel. Netw. 12 (1) (2006) 63–78.

[14] I. Rhee, A. Warrier, M. Aia, J. Min, M.L. Sichitiu, Z-MAC: a hybrid MAC for wireless sensor networks, IEEE/ACM Transac. Netw. (TON) 16 (3) (2008) 511–524.

[15] P. Suriyachai, J. Brown, U. Roedig, Time-critical data delivery in wireless sensor networks, Distributed Computing in Sensor Systems, Springer, Berlin Heidelberg, 2010, pp. 216–229.

[16] K.S. Prabh, T.F. Abdelzaher, On scheduling and real-time capacity of hexagonal wireless sensor networks, IEEE, 2007, July, pp. 136–145.

[17] K.G. Shin, HARTS: a distributed real-time architecture, Computer 24 (5) (1991) 25–35.

[18] P. Santi, Topology control in wireless ad hoc and sensor networks, ACM Comput. Surv. (CSUR) 37 (2) (2005) 164–194.

[19] L. Sehgal, V. Choudhary, REEH: Residual Energy Efficient Heterogeneous Clustered Hierarchy Protocol for Wireless Sensor Networks, Int. J. Sci Eng Res 2 (12) (2011) 1–5.

[20] R. Saravanakumar, S.G. Susila, J. Raja, Energy efficient constant cluster node scheduling protocol for wireless sensor networks, WSEAS Transac. Commun. 10 (4) (2011) 119–128.

[21] A.R. Dutta, B.S. Saha, C.A.K. Mukhopadhyay, Efficient clustering techniques to optimize the system lifetime in wireless sensor network, in: International Conference on Advances in Engineering, Science and Management (ICAESM), 2012, IEEE, 2012, March, pp. 679–683.

[22] M. Noori, M. Ardakani, Lifetime analysis of random event-driven clustered wireless sensor networks, Mobile Comput., IEEE Transac. 10 (10) (2011) 1448–1458.

[23] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: International Conference on System Sciences, 2000. Proceedings of the 33rd Annual Hawaii, IEEE, 2000, January, p. 10.

[24] W.B. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, Wirel. Commun. IEEE Transac. 1 (4) (2002) 660–670.

[25] Stephanie Lindsey, Cauligi Raghavendra, KrishnaM. Sivalingam, Data gathering algorithms in sensor networks using energy metrics, Parall. Distrib. Syst. IEEE Transac. 13 (9) (2002) 924–935.

[26] Y. Tao, Y. Zhang, Y. Ji, Flow-balanced routing for multi-hop clustered wireless sensor networks, Ad hoc Netw 11 (1) (2012) 541–554.

[27] A.A. Abbasi, M. Younis, A survey on clustering algorithms for wireless sensor networks, Comput. Commun. 30 (14) (2007) 2826–2841.

[28] Instruments, Texas. "CC2420: 2.4 GHz IEEE 802.15. 4/ZigBee-ready RF Transceiver." Available at http://www.ti.com.cn/cn/lit/ds/symlink/cc2420.pdf, 2006.

[29] T.C. Huang, A self-stabilizing algorithm for the shortest path problem assuming the distributed demon, Comput. Math. Appl. 50 (5) (2005) 671–681.

[30] T.C. Huang, J.C. Lin, C.Y. Chen, C.P. Wang, The worst-case stabilization time of a self-stabilizing algorithm under the weakly fair daemon model, Int. J. Artif. Life Res. (IJALR) 1 (3) (2010) 45–52.

[31] E.W. Dijkstra, Self-stabilizing systems in spite of distributed control, Commun. ACM 17 (11) (1974) 643–644.

[32] J.E. Burns, Self-stabilizing Ring Without Demons, Technical Report GIT-ICS-87/36, Georgia Tech., 1987

[33] András. Varga, The OMNeT++ discrete event simulation system, in: Proceedings of the European Simulation Multiconference (ESM'2001), 9, 2001.

[34] Md Abdur Razzaque, et al., QoS-aware distributed adaptive cooperative routing in wireless sensor networks, Ad Hoc Netw. 19 (2014) 28–42.

[35] L. Mokdad, et al., Performance evaluation tools for QoS MAC protocol for wireless sensor networks, Ad Hoc Netw. 12 (2014) 86–99.

[36] T. Du, et al., A high efficient and real time data aggregation scheme for WSNs, Int. J. Distrib. Sens. Netw. 501 (2015) 261381. APA.

[37] R. Rosen, Internet control message protocol (ICMP), Linux Kernel Networking Apress (2014) 37–61.

[38] Chen, Da-Ren. A link-and hop-constrained clustering for multi-hop wireless sensor networks. Computer Communications (2015). http://dx.doi.org/10.1016/j.comcom.2015.05.009.

[39] Limin Hu, Distributed code assignments for CDMA packet radio networks, in: INFOCOM'91. Proceedings. Tenth Annual Joint Conference of the IEEE Computer and Communications Societies. Networking in the 90s, IEEE, 1991.

[40] Instruments, Texas. CC1000 Single chip very low power RF transceiver data sheet. Available at Available at http://www.ti.com/lit/ds/symlink/cc1000.pdf, 23. AS, C, 2004.

[41] Marion Souil, Bouabdallah Abdelmadjid, A.E. Kamal, Efficient QoS provisioning at the MAC layer in heterogeneous wireless sensor networks, Comput. Commun. 43 (2014) 16–30.

[42] M Yigitel, Ozlem Aykut, Durmaz Incel, Cem Ersoy, Design and implementation of a QoS-aware MAC protocol for Wireless Multimedia Sensor Networks, Comput. Commun. 34 (16) (2011) 1991–2001.

[43] Shensheng Tang, Wei Li, QoS supporting and optimal energy allocation for a cluster based wireless sensor network, Comput. Commun. 29 (13) (2006) 2569–2577.

[44] Babar Nazir, Halabi Hasbullah, Energy efficient and QoS aware routing protocol for clustered wireless sensor network, Comput. Electric. Eng. 39 (8) (2013) 2425–2441.

[45] Liangshan Jiang, et al., Lifetime maximization through dynamic ring-based routing scheme for correlated data collecting in WSNs, Comput. Electric. Eng. 41 (2015) 191–215.

[46] C. Santhi, D. Sharmila, A self-organized location aware energy efficient protocol for wireless sensor networks, Comput. Electric. Eng. 41 (2015) 265–274.

[47] Guang Ye, et al., Energy Balanced Redeployment Algorithm for Heterogeneous Wireless Sensor Networks, Math. Prob. Eng. 2015 (2015) Article ID 608121, doi:10.1155/2015/608121.

[48] RC Suganthe, P. Balasubramanie, Improving QoS in delay tolerant mobile ad hoc network using multiple message ferries, Netw. Protoc. Algor. 3 (4) (2011) 32–53.

[49] Muhsin Atto, Chris Guy, Routing protocols and quality of services for security based applications using wireless video sensor networks, Netw. Protoc. Algor. 6 (3) (2014) 119–147.

[50] JuanR. Diaz, Jaime Lloret, JoseM. Jimenez, JoelJ.P.C. Rodrigues, A QoS-based wireless multimedia sensor cluster protocol, Int. J. Distrib. Sens. Netw. 2014 (2014) Article ID 480372, 17, doi:10.1155/2014/480372.

[51] K. Benkic, M. Malajner, P. Planinsic, Z. Cucej, Using RSSI value for distance estimation in wireless sensor networks based on ZigBee, in: Conference on 15th International Systems, Signals and Image Processing, June 2008, pp. 303–306.

[52] Liqiang Tao, Yu. Fengqi, Delay-Jitter aware slot assignment for real-time applications in wireless multimedia ad hoc sensor networks, Comput. Commun. 35 (16) (2012) 1967–1982.

[53] Flávio Vieira, Henrique Teles, J.A. Santos, A.A. Cardoso, Estimation of backlog and delay in OFDM/TDMA systems with traffic policing using Network Calculus, Comput. Electric. Eng. 39 (8) (2013) 2507–2520.

**Da-Ren Chen** received his B.E., M.E. and Ph.D. degrees from National Taiwan University of Science and Technology, Taipei, Taiwan, in 1999, 2001 and 2006, respectively. Presently, he is an associate professor in the Department of Information Management at National Taichung University of Science and Technology. His research interests include real-time systems, cyber-physical systems, cloud computing, parallel and distributed computing and wireless sensor networks.