Event-aware Hierarchical Routing with Differential Compression to Extend WSN Lifetime

You-Chiun Wang and Bo-Chun Pan
Department of Computer Science and Engineering,
National Sun Yat-sen University, Kaohsiung, 80424, Taiwan
Email: ycwang@cse.nsysu.edu.tw; m063040002@student.nsysu.edu.tw

***Abstract*—Wireless sensor networks are popularly used in IoT applications. It is essential to save energy of sensors on reporting data. Though many routing methods have been developed, how to well integrate packet routing with data compression is not much discussed. Hence, we propose an *event-aware hierarchical routing with differential compression (EHR-DC)* scheme. It groups sensors and selects a *cluster head (CH)* in each group to manage routing and compression. In normal times, sensors transmit data to their CHs, which are condensed by exploiting spatial correlation. When events appear, sensors adaptively forward data to nearby CHs to raise the efficiency of compression. Through simulations, we show that EHR-DC outperforms other methods in terms of network lifetime and the amount of sensing data retrieved by the sink.**

***Keywords*—compression, event, hierarchical routing, spatial correlation, wireless sensor network (WSN).**

I. Introduction

Wireless sensor network (WSN) is the core part of Industry 4.0 and IoT [1], which comprises many sensors capable of self-organization. Each sensor gathers data from the surroundings and reports them to the sink. Various WSN-based applications are developed, including visual surveillance [2], unmanned vehicles [3], health care [4], air-quality evaluation [5], interactive shopping [6], and agriculture applications [7].

In a large WSN, sensors report data to the sink via multihop relays. The closer to the sink a sensor is, the more data it has to relay. Due to their size limitation, sensors are usually powered by compact batteries. When recharging batteries is infeasible, how to save energy of sensors on relaying data is critical [8]. Thus, the routing method plays a key role on deciding network lifetime, as those sensors beside the sink will quickly run out of energy, which disconnects the sink from the WSN [9].

On the other hand, data compression is a popular technique to diminish the amount of data sent by sensors. This technique exploits the correlation of sensing data to merge and condense them, which can be *temporal* or *spatial* [10]. Temporal correlation indicates that the data produced by a single sensor may have just small fluctuation for a short while. Spatial correlation means that the sensors in close proximity would collect similar data. By discarding redundant information, multiple packets of sensing data can be compressed to cut the transmission cost.

This paper aims to combine packet routing with data compression to extend WSN lifetime. Considering that sensors are simple devices [11], we propose a lightweight scheme called *event-aware hierarchical routing with differential compression (EHR-DC)*. For ease of management, EHR-DC divides a WSN into grids and chooses a CH in each grid to do the routing job based on its estimated residual energy. Through a differential idea, the CH merges data sent from members by using spatial correlation. If an event occurs, some sensors in the event region change to send packets to neighboring CHs, so as to facilitate compression by raising correlation of data gathered by them. To balance loads of sensors, CHs will be reselected regularly. Simulation results show that EHR-DC greatly extends network lifetime and makes the sink get more data. Our contribution is to develop a lightweight routing scheme efficiently combined with data compression for performance improvement, which is fit for resource-constrained sensors and helps save their energy.

II. Related Work

Ad hoc on-demand distance vector (AODV) is a well-known routing method and it has many variations [12], where a sensor finds the sink by flooding route requests. Like AODV, a few routing methods build shortest paths to reach the sink. In [13], each sensor estimates the cost for sending packets to the sink. A sensor can relay packets if it is on the minimum-cost path. Gradient-based methods [14] assign a height to every sensor to reveal its hop count to the sink. The gradient of each link is the difference between heights of its endpoint sensors, and packets will be routed along large-gradient links. Given the locations of sensors, geographical routing methods [15] let each sensor forward its packets to a neighbor closest to the sink. However, sensors on popular shortest paths will exhaust energy quickly. Thus, some methods consider energy of sensors when finding paths. In [16], when a sensor gets a route request, it checks if residual energy is twice more than the energy spent to forward packets. If so, the request is sent to neighbors, or discarded otherwise. Fradj et al. [17] estimate the energy cost to forward one packet from a source to a target through every relay node, and choose the node with the lowest cost to relay packets. In the *residual energy based multipath routing (REBM)* method [18], a sensor asks the neighbor with more energy and fewer children to relay its packets. However, those sensors with more initial energy would be imposed with a heavy load on routing.

In hierarchical routing methods, parts of sensors are designated as CHs to handle packet routing. *Low energy adaptive clustering hierarchy (LEACH)* is one representative with many successors [19]. They ask each sensor deciding whether to act as a CH through a probability. Other nodes then send packets to their nearest CHs, which will be relayed to the sink directly or in a multihop manner. Liang et al. [20] organize a WSN into rings, and select one CH to gather data from sensors in each ring. Several methods consider energy of sensors in the selection of CHs. The work [21] picks CHs based on residual energy and link degree of each sensor. Kumar et al. [22] pick a sensor to be a CH if it has more energy than its neighbors. A *node ranking clustering algorithm (NRCA)* [23] divides the sensing field into grids and chooses a sensor with more energy and shorter distance to the sink as the CH in each grid. As can be seen, how to exploit data compression for CHs to decrease the amount of data transmissions is not addressed.

Few studies apply compression to existing routing methods. For example, [24] assumes that every k packets can be merged into a compressed packet and uses this compression scheme in AODV. Each sensor decides whether to encapsulate packets by itself or send them to a neighbor to speed up compression, with the aim of reducing packet latency. The study [25] combines LEACH with data reduction, where each sensor keeps a copy of the recently sent packet. If more than 50% of data bytes in a newly generated packet are the same with the previous packet, the sensor sends to the CH merely different bytes along with a code to show which bytes are different. However, both [24] and [25] disregard the impact of events. This motivates us to develop EHR-DC that can modify routes when events appear, so as to facilitate compression and extend network lifetime.

III. System Model

Let us consider a sensing field where sensors are uniformly deployed and form a connected WSN. All sensors are identical in hardware and battery capacity, which produce the same kind of sensing data allowed to be merged together and compressed. The sensing field is partitioned into grids beforehand. Through some positioning approaches [26], the locations of sensors can be obtained. Thus, every sensor knows which grid it resides in. Events could appear at anytime and anywhere. Once detecting events, sensors would generate sensing data whose values have large differences with those produced in normal times.

A sensor spends energy on generating, sending, and receiving data. When a sensor s_i produces an l-bit packet of sensing data, it consumes an amount of energy [27]:

$$\tilde{E}_{\mathbf{G}}(s_i, l) = (u_i^{\mathbf{G}} \times c_i^{\mathbf{G}} \times t_i^{\mathbf{G}}) \times l, \tag{1}$$

where $u_i^{\mathbf{G}}$, $c_i^{\mathbf{G}}$, and $t_i^{\mathbf{G}}$ denote the voltage, current, and time required by s_i to create the packet, respectively. When s_i sends the packet to a node s_j, s_i takes an amount of energy:

$$\tilde{E}_{\mathbf{S}}(s_i, s_j, l) = [\alpha_i^{\mathbf{T}} + \alpha_i^{\mathbf{A}} \times \hat{D}(s_i, s_j)^2] \times l. \tag{2}$$

Specifically, $\alpha_i^{\mathbf{T}}$ and $\alpha_i^{\mathbf{A}}$ are the power for s_i's transmitter and amplifier to send a bit, and $\hat{D}(\cdot,\cdot)$ is the distance function. Let β_j be the power for s_j's receiver to get a bit. Then, s_j should spend an amount of energy to take the packet:

$$\tilde{E}_{\mathbf{R}}(s_j, l) = \beta_j \times l. \tag{3}$$

Our objective is to develop a routing scheme along with data compression to maximize WSN lifetime, which is defined by the period since it starts working until the first sensor dies.

IV. The Proposed EHR-DC Scheme

The description of EHR-DC includes three parts as follows: Section IV-A addresses how to select a good CH in each grid and when to select it. Section IV-B discusses routing strategies in normal times and when events appear. Then, the *differential compression (DC)* scheme is proposed in Section IV-C.

A. CH Selection

Since the CH has to route packets on behalf of other sensors in a grid, we should pick a node s_i to act as the CH based on two conditions: 1) s_i has the most energy and 2) s_i can spend the least energy to do the routing job. Let T be the length of a period and r be the sensing rate of sensors. We compute the *next-period energy (NPE)* of each sensor s_i by

$$\begin{aligned}\tilde{E}_i = e_i - Tr \times \tilde{E}_{\mathbf{G}}(s_i, l) - [\textstyle\sum_{s_j \in \hat{\mathcal{S}} \setminus \{s_i\}} Tr] \times \tilde{E}_{\mathbf{R}}(s_i, l)\\ - \tilde{E}_{\mathbf{S}}(s_i, \mathrm{NH}_i, \sigma \textstyle\sum_{s_j \in \hat{\mathcal{S}}} Trl),\end{aligned} \tag{4}$$

where e_i is the current energy of s_i, l is the length of a normal packet, $\hat{\mathcal{S}}$ is the set of all sensors in the grid, NH_i is the next hop of s_i, and σ is the compression ratio. In Eq. (4), the term of $Tr \times \tilde{E}_{\mathbf{G}}(s_i, l)$ is the energy used by s_i to generate its sensing data in a period, the term of $[\sum_{s_j \in \hat{\mathcal{S}} \setminus \{s_i\}} Tr] \times \tilde{E}_{\mathbf{R}}(s_i, l)$ is the energy for s_i to collect sensing data from all other sensors, and the term of $\tilde{E}_{\mathbf{S}}(s_i, \mathrm{NH}_i, \sigma \sum_{s_j \in \hat{\mathcal{S}}} Trl)$ indicates the energy for s_i to send encapsulated data to its next hop.

By applying Eqs. (1), (2), and (3) to Eq. (4), we obtain that

$$\tilde{E}_i = e_i - \zeta_1[\zeta_2 + \sigma(\alpha_i^{\mathbf{T}} + \alpha_i^{\mathbf{A}} \times \hat{D}(s_i, \mathrm{NH}_i)^2)]. \tag{5}$$

Here, $\zeta_1 = Trl$ and $\zeta_2 = u_i^{\mathbf{G}} \times c_i^{\mathbf{G}} \times t_i^{\mathbf{G}} + (|\hat{\mathcal{S}}| - 1)\beta_i$, where $|\hat{\mathcal{S}}|$ is the number of sensors in the grid (including s_i). Since sensors are homogeneous, which means that all coefficients in Eqs. (1), (2), and (3) are the same for each sensor, ζ_1 and ζ_2 must be constants. Thus, both e_i and $\hat{D}(s_i, \mathrm{NH}_i)$ are the only variables in Eq. (5). For practical implementation, a sensor can use two constants ζ_3 and ζ_4 to simplify Eq. (5) as follows:

$$\tilde{E}_i = e_i - \zeta_3 - \zeta_4 \hat{D}(s_i, \mathrm{NH}_i)^2, \tag{6}$$

where $\zeta_3 = \zeta_1\zeta_2 + \sigma\zeta_1\alpha_i^{\mathbf{T}}$ and $\zeta_4 = \sigma\zeta_1\alpha_i^{\mathbf{A}}$. In Section IV-C, we will analyze the ratio σ. By keeping the values of ζ_3 and ζ_4 in the local memory of sensors, the calculation of NPE can be significantly reduced.

Furthermore, we have to decide the next hop (i.e., NH_i) for s_i. It is expected to be the CH in adjacent grids (including the diagonal ones) that is closest to the sink. However, this solution may not be feasible, as we have not known the CHs in adjacent grids yet. Instead, we consider using the *center* of an adjacent grid that is closest to the sink to be NH_i. As for the grid right beside the sink, NH_i will be the sink. In this way, each sensor can also store the value of $\hat{D}(s_i, \mathrm{NH}_i)^2$ beforehand, thereby further reducing the complexity of calculation in Eq. (6). Then, the sensor with the maximum NPE will be chosen as the CH.

Evidently, when a sensor serves as the CH, the expenditure of its energy will be raised. To avoid such sensors exhausting energy quickly, we should periodically reselect the CH in each

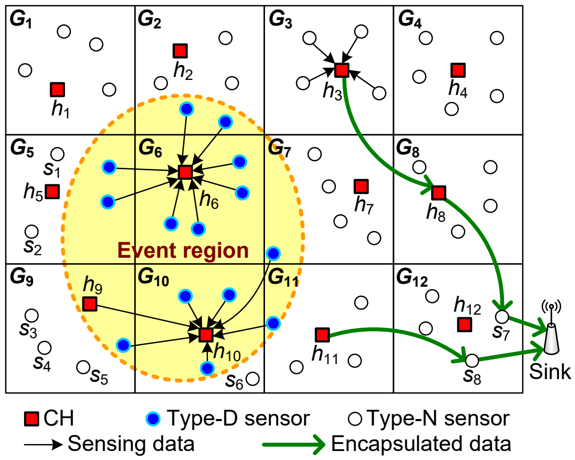


Fig. 1: Data forwarding in EHR-DC.

grid. One possible approach is to compare the CH's energy and the average energy of sensors in the grid. If the former is below a ratio of the latter, the CH should be reselected. Unfortunately, this approach requires the CH to acquire residual energy of all members in the grid, which incurs extra message overhead. To address the issue, we let a sensor s_i record its energy whenever it starts serving as the CH (denoted by $e_i^{\mathbf{B}}$). Once the current energy of s_i falls below $e_i^{\mathbf{B}}/\xi$, where $\xi > 1$, which means that s_i has consumed much energy on doing the routing job, it will ask to reselect the CH to conserve energy. The suggested value of ξ is within $[2, 3)$, so s_i can reserve at least one half of its energy (in terms of $e_i^{\mathbf{B}}$) while resulting in a low frequency of CH reselection. Notice that if s_i possesses very little residual energy (e.g., less than 5% of its initial energy) but s_i is chosen as the CH again, it implies that every sensor in the grid actually holds pretty low energy. In this case, we ask s_i to keep acting as the CH, until it uses up energy. Doing so can avoid wasting extra energy on CH reselection in this special case.

B. Data Forwarding

In EHR-DC, packets are routed in a grid-based manner. Let h_k be the CH of a grid G_k. In addition, we call the grid beside the sink the *proximal grid*. Fig. 1 gives an example, where the proximal grid is G_{12}. Evidently, the proximal grid will be the last step of each routing path in the WSN.

For a grid G_k where no events occur, each sensor directly transmits its sensing data to h_k. Afterwards, h_k combines and encapsulates these data into a compressed packet by the DC scheme discussed in Section IV-C, and forwards this packet to a neighboring CH, say, h_{k+1} that is closest to the sink. If the next grid of h_{k+1} is not the proximal grid, h_{k+1} will then pass the packet to its next-hop CH based on the above procedure. Otherwise, h_k arbitrarily picks a node (may not necessarily be the CH) in the proximal grid, and sends the packet to the sink via that node. Fig. 1 illustrates two examples. In grid G_3, all members send their data to h_3. After encapsulating them, h_3 forwards one compressed packet to the sink along the path $h_3 \to h_8 \to s_7 \to$ sink. On the other hand, h_{11} chooses sensor s_8 to relay its packet in the proximal grid. The reason why we randomly choose a node in the proximal grid to relay data is to avoid burdening the CH in the proximal grid with too heavy load (as the proximal grid is the last grid to the sink). Doing so helps achieve load balance in the proximal grid.

When an event occurs, a different routing strategy should be applied to those grids covered by the event region. Let us call a sensor which detects the event a *type-D sensor*. Otherwise, it is called a *type-N sensor*. Since there exists a remarkable gap between their values, it is inefficient to combine and compress the sensing data generated by different types of sensors. Thus, if a gird contains different types of sensors, we have to bypass their data to different CHs, so as to make sure that each CH can collect the sensing data with high correlation (i.e., produced by the same type of sensors).

Suppose that the event region covers N_c grids (for convenience, these grids are called *event grids*). We choose $\lceil N_c/4 \rceil$ CHs whose grids contain the most number of type-D sensors (called *leader CHs*). Then, each type-D sensor forwards its data to the nearest leader CH. An example is shown in Fig. 1, where G_2, G_5, G_6, G_7, G_9, G_{10}, and G_{11} are event grids. Since $N_c = 7$ and $\lceil 7/4 \rceil = 2$, we pick both h_6 and h_{10} to be leader CHs, which collect data from nearby type-D sensors.

As for each type-N sensor, say, s_x in event grids, there are three cases to be considered, depending on its CH h_y:

[Case 1] *h_y is a leader CH.* Since h_y collects data from type-D sensors, s_x has to forward its data to the closest non-leader CH. Grid G_{10} in Fig. 1 illustrates an example, where s_6 should send its data to h_{11} (instead of h_{10}).

[Case 2] *h_y is not a leader CH but it is a type-D sensor.* In this case, s_x can send its data to h_y for compression. However, h_y needs to forward its own sensing data to the nearest leader CH. In Fig. 1, gird G_9 gives an example. Sensors s_3, s_4, and s_5 send their data to h_9 for compression. When h_9 generates its own sensing data, the data should be forwarded to h_{10}.

[Case 3] *h_y is neither a leader CH nor a type-D sensor.* This case is similar to case 2, but h_y will combine its own sensing data with s_x's data. Grid G_5 in Fig. 1 presents one example, where h_5 will merge its sensing data with the data sent from s_1 and s_2 (as they all belong to type-N sensors).

Then, Theorem 1 analyzes the maximum length of routing paths (in hop counts) constructed by EHR-DC.

Theorem 1: Given $m \times n$ grid partition for the sensing field, the longest routing path contains $(\max\{m, n\} + 1)$ hops.

Proof: Without loss of generality, we assume that the sink locates at the bottom-right corner of the sensing field. Thus, the longest routing path must originate from the top-left grid. This path first passes through each diagonal grid, followed by some grids in the bottom row. Take Fig. 1 as an example, the longest routing path will pass through grid G_1, G_6, G_{11}, and G_{12}. Since it takes one hop to cross a grid, the aggregate hop count is thus $\max\{m, n\}$. However, this path could start from a sensor in G_1 (when G_1 is not an event grid) or a node in G_2, G_5, or G_6 (if they are event grids), we have to add one more hop in the length, thereby proving this theorem. ∎

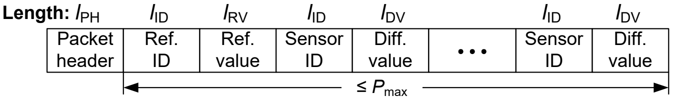


Fig. 2: Format of a compressed packet.

C. Differential Compression

Consider that a CH h_y collects the sensing data from a set $\hat{\mathcal{S}}$ of sensors, where $\hat{\mathcal{S}}$ may or may not include h_y (depending on the three data forwarding cases discussed in Section IV-B). Let us denote by $\hat{\mathcal{V}}$ the set of values of these data. Specifically, the value of the sensing data given by a sensor $s_i \in \hat{\mathcal{S}}$ will be $v_i \in \hat{\mathcal{V}}$. Then, h_y encapsulates data in $\hat{\mathcal{V}}$ into one compressed packet, so as to save the transmission overhead. To do so, we propose the format for compressed packets as given in Fig. 2. In particular, we choose the smallest value v_x in $\hat{\mathcal{V}}$ to be the reference (Ref.) value. The compressed packet will be "packet header, s_x, v_x, s_{i_1}, $v_{i_1} - v_x$, s_{i_2}, $v_{i_2} - v_x$, $\cdots$, s_{i_k}, $v_{i_k} - v_x$", where $s_{i_1}, s_{i_2}, \cdots, s_{i_k} \in \hat{\mathcal{S}}$. Here, since each difference (Diff.) value must be non-negative (as v_x is the minimum value), there is no need to add one signed bit in each Diff. value field. In this way, we can save the space in the packet's payload.

In Fig. 2, we denote by l_{PH}, l_{ID}, l_{RV}, and l_{DV} the length of packet header, Ref./Sensor ID, Ref. value, and Diff. value, respectively. Here, l_{PH} depends on the communication protocol. To reduce l_{ID}, we design condensed IDs for sensors (discussed later). The condition of $l_{\mathrm{RV}} \geq l_{\mathrm{DV}}$ must hold, as the Ref. value field keeps the *complete* value of sensing data while the Diff. value field stores merely a difference. Lemma 1 estimates l_{DV}. Given the maximum payload size $P_{\max}$, Theorem 2 analyzes the compression ratio σ.

Lemma 1: Let $D_{\max}$ be the maximum difference between values of any two sensing data. Then, l_{DV} is $\lceil \lg D_{\max} \rceil$ bits.

Proof: To avoid overflow, we have to guarantee that

$$2^{l_{\mathrm{DV}}} \geq D_{\max} \Rightarrow l_{\mathrm{DV}} \geq \lg D_{\max}. \tag{7}$$

Thus, we derive that $l_{\mathrm{DV}} = \lceil \lg D_{\max} \rceil$. ■

Theorem 2: The compression ratio is

$$\sigma = \frac{l_{\mathrm{PH}} + l_{\mathrm{ID}} + l_{\mathrm{RV}} + \varphi(l_{\mathrm{ID}} + l_{\mathrm{DV}})}{(\varphi + 1) \times (l_{\mathrm{PH}} + l_{\mathrm{RV}})}, \tag{8}$$

where $\varphi = \lfloor (P_{\max} - l_{\mathrm{ID}} - l_{\mathrm{RV}})/(l_{\mathrm{ID}} + l_{\mathrm{DV}}) \rfloor$.

Proof: Observing from the format in Fig. 2, the payload includes one Ref. value (together with the Ref. ID) and φ Diff. values (together with their Sensor IDs). Since the maximum payload size is $P_{\max}$, it is guaranteed that

$$l_{\mathrm{ID}} + l_{\mathrm{RV}} + \varphi(l_{\mathrm{ID}} + l_{\mathrm{DV}}) \leq P_{\max} \Rightarrow \varphi \leq \frac{P_{\max} - l_{\mathrm{ID}} - l_{\mathrm{RV}}}{l_{\mathrm{ID}} + l_{\mathrm{DV}}}$$

In other words, one compressed packet actually carries information of $(\varphi + 1)$ packets of sensing data. If we do not apply the DC scheme, the CH has to relay $(\varphi + 1)$ uncompressed packets, each with a length of $l_{\mathrm{PH}} + l_{\mathrm{RV}}$ (i.e., packet header + complete value of sensing data). In this case, the amount of data sent by the CH will be $\Psi_n = (\varphi + 1) \times (l_{\mathrm{PH}} + l_{\mathrm{RV}})$. On the other hand, the total length of one compressed packet is $\Psi_c = l_{\mathrm{PH}} + l_{\mathrm{ID}} + l_{\mathrm{RV}} + \varphi(l_{\mathrm{ID}} + l_{\mathrm{DV}})$. Thus, the compression ratio σ is Ψ_c/Ψ_n, which is exactly the same with Eq. (8). ■

TABLE I: Parameters used in the simulation.

parameter	value
area of the sensing field	$1000\,\mathrm{m} \times 1000\,\mathrm{m}$ (with 10×10 grids)
communication range	150 m
packet length	20 bytes (uncompressed)
battery capacity	6480 J
coefficients in Eq. (1)	$u_i^{\mathbf{G}}$ = 1.5V, $c_i^{\mathbf{G}}$ = 25mA, $t_i^{\mathbf{G}}$ = 0.25 ms
coefficients in Eq. (2)	$\alpha_i^{\mathbf{T}}$ = 50nJ/bit, $\alpha_i^{\mathbf{A}}$ = 100pJ/bit (per m^2)
coefficient in Eq. (3)	β_j = 50nJ/bit
time interval	sensing: 1800 seconds, event: 10800 seconds

A smaller σ value in Eq. (8) implies that the efficiency of data compression is better. From Lemma 1 and Theorem 2, it is apparent that σ decreases as the maximum difference $D_{\max}$ reduces. That explains why we need to bypass the sensing data generated by type-D and type-N sensors to different CHs for compression, as mentioned earlier in Section IV-B. In this way, we can improve the performance of our compression scheme.

The last issue is about IDs of sensors. To reduce l_{ID}, we design a *grid-based short addressing (GSA)* method. Consider that one sensor is referred as the jth sensor in a grid G_i. This information is known by the sink since sensor deployment and grid partition are conducted beforehand. Then, its ID will be the combination of i and j. For example, suppose that there are 100 grids and each grid contains at most 10 sensors. Given an ID of 01100101001, it means that this sensor is the 9th one (referring to the last 4 bits) in grid 50 (referring to the first 7 bits). As compared with some traditional solutions such as using 48-bit MAC addresses to be IDs, the GSA method can efficiently reduce l_{ID}. Theorem 3 estimates l_{ID} by using GSA.

Theorem 3: Given $m \times n$ grid partition, GSA guarantees that $l_{\mathrm{ID}} = \lceil \lg mn \rceil + \lceil \lg N_{\max} \rceil$ bits, where $N_{\max}$ is the maximum number of sensors in a grid.

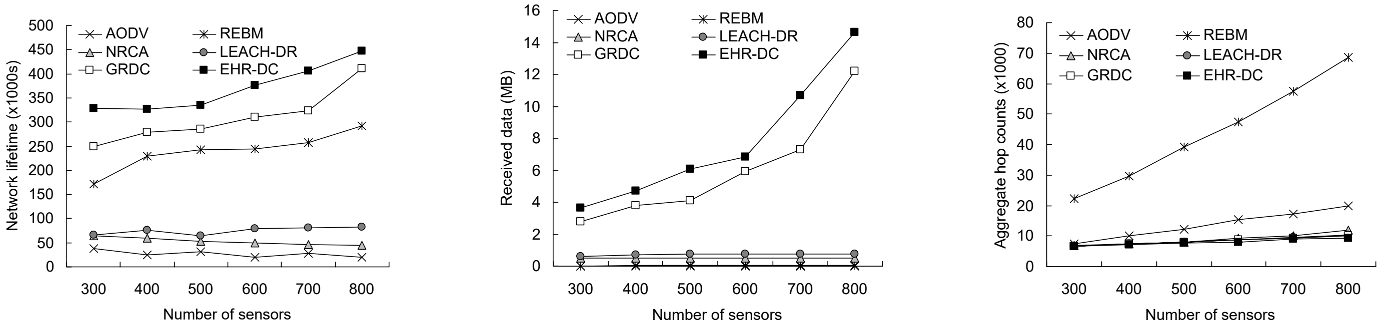
Proof: Based on Lemma 1, it requires $\lceil \lg \phi \rceil$ bits to store data whose maximum possible value is ϕ. Since there are mn grids and each of them contains no more than $N_{\max}$ sensors, l_{ID} will thus be $\lceil \lg mn \rceil + \lceil \lg N_{\max} \rceil$ in the GSA method. ■

V. EXPERIMENTAL RESULTS

In the simulation, we consider a sensing field where sensors are randomly deployed and form a connected WSN. The sink is placed on its bottom-right corner. Table I lists all parameters. Three routing methods mentioned in Section II are selected for comparison, including AODV [12], REBM [18], and NRCA [23]. Besides, we compare EHR-DC with two routing methods that involve data compression. The LEACH-DR method [25] applies data reduction to LEACH. The *grid-based routing with DC (GRDC)* method is similar to EHR-DC, except that sensors always send data to their CHs (even if there are events). Here, GRDC is used to highlight the superiority of EHR-DC in terms of dynamically changing routes when events occur.

A. Effect of the Number of Sensors

Fig. 3(a) gives network lifetime, where there are 300 to 800 sensors and two events appear in the sensing field. In AODV,



(a) Network lifetime (b) Sensing data gotten by the sink (c) Aggregate hop counts

Fig. 3: Performance evaluation by varying the number of sensors.

sensors spend more energy on flooding route requests. Besides, AODV always builds the shortest paths to the sink, so sensors on these paths would relay lots of data, thereby running out of energy quickly. Hierarchical routing methods including NRCA and LEACH-DR solve this problem by picking CHs to route packets, where LEACH-DR outperforms NRCA by using data compression. REBM selects sensors with more energy to relay data, so it has longer lifetime than the above methods. Through carefully choosing CHs and encapsulating packets by DC, both GRDC and EHR-DC can greatly raise lifetime. In fact, EHR-DC further extends 20% of lifetime than GRDC, which shows the benefit of adaptively changing routes as events occur.

Fig. 3(b) then presents the amount of sensing data received by the sink (in megabytes). If data compression is applied, we measure the amount of data after the sink extracts compressed packets. In general, the longer the lifetime is, the more data the sink gets. An exception is REBM, and we will give the reason later. Since GRDC and EHR-DC have much longer lifetime and they use DC to compress data, both of them substantially increase the amount of data retrieved by the sink. Moreover, as EHR-DC flexibly varies routes when events appear, it further increases 29.3% of received data than GRDC, which validates its effectiveness on routing packets.

In Fig. 3(c), we study the aggregate hop counts of all routing paths in the WSN. If a route is longer, the probability of packet loss (e.g., due to data collision in the MAC layer) will increase [28]. For hierarchical routing methods (i.e., NRCA, LEACH-DR, GRDC, and EHR-DC), since most sensors forward data to CHs directly, these methods can decrease routing paths than AODV. On the other hand, REBM prefers choosing a neighbor closer to each sender as its relay node, which makes a routing path contain many hops. This situation becomes more serious when there are more sensors. In this case, many packets would be actually lost in REBM. That is why REBM makes the sink get the least amount of data in Fig. 3(b).

B. Effect of the Number of Events

Fig. 4(a) compares network lifetime by varying the number of events, where there are 400 sensors. As can be seen, EHR-DC achieves the longest lifetime, followed by GRDC, REBM, LEACH-DR, NRCA, and AODV. When there are more events, the diversity of sensing data will increase accordingly. Thus, it becomes not easy to find adequate (nearby) CHs for sensors to send packets to improve data correlation. That is why network lifetime in EHR-DC decreases as the number of events grows. Evidently, the worst case occurs when each sensor has to send data to its CH even if there are events. Thus, EHR-DC will at least have the equal lifetime with GRDC (if we keep increasing events), which is still longer than that of other methods.

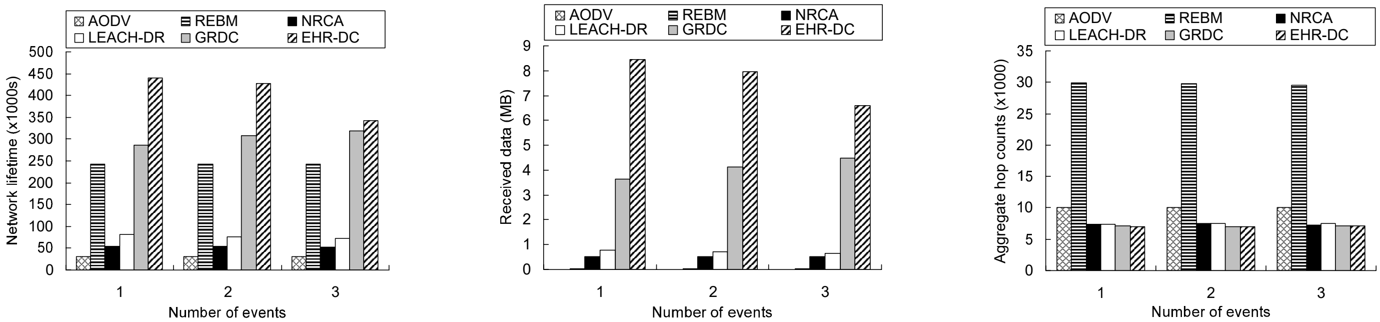
Fig. 4(b) shows the amount of sensing data received by the sink. Similarly, the received data in EHR-DC decreases when there are more events. As compared with other methods, EHR-DC allows the sink capturing many more data, which verifies that the routing strategy in EHR-DC can indeed work well with our compression approach proposed in Section IV-C. Finally, the result in Fig. 4(c) shows that EHR-DC also reduces lengths of routing paths than both AODV and REBM, just like other hierarchical methods (i.e., NRCA, LEACH-DR, and GRDC).

VI. Conclusion

Sensors have limited energy but they have to keep reporting data to the sink via multihop relays, so energy-efficient routing is imperative to lengthen WSN lifetime. This paper proposes a lightweight, efficient EHR-DC scheme to well integrate packet routing with data compression. In each grid, the sensor with the maximum NPE is designated as the CH to route packets for other nodes. It then encapsulates multiple packets to save the transmission cost by the DC method. When sensors detect an event, they can adaptively forward data to neighboring CHs to improve the efficiency of compression. Simulation results validate that EHR-DC not only prolongs network lifetime but also increases the amount of data sent to the sink, as compared with AODV, REBM, NRCA, LEACH-DR, and GRDC. As for the future work, we will consider the effect of obstacles in the sensing field [29]. Moreover, it deserves further investigation to take account of the mobility of sensors [30].

Acknowledgment

You-Chiun Wang's research is co-sponsored by the Ministry of Science and Technology under Grant No. MOST 108-2221-E-110-016-MY3, Taiwan.



(a) Network lifetime (b) Sensing data gotten by the sink (c) Aggregate hop counts

Fig. 4: Performance evaluation by varying the number of events.

REFERENCES

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: automation networks in the era of the Internet of Things and Industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[2] Y. C. Wang, Y. F. Chen, and Y. C. Tseng, "Using rotatable and directional (R&D) sensors to achieve temporal coverage of objects and its surveillance application," *IEEE Transactions on Mobile Computing*, vol. 11, no. 8, pp. 1358–1371, 2012.

[3] Y. C. Wang, "Mobile sensor networks: system hardware and dispatch software," *ACM Computing Surveys*, vol. 47, no. 1, pp. 12:1–12:36, 2014.

[4] A. Alaiad and L. Zhou, "Patients' adoption of WSN-based smart home healthcare systems: an integrated model of facilitators and barriers," *IEEE Transactions on Professional Communication*, vol. 60, no. 1, pp. 4–23, 2017.

[5] Y. C. Wang and G. W. Chen, "Efficient data gathering and estimation for metropolitan air quality monitoring by using vehicular sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7234–7248, 2017.

[6] Y. C. Wang and C. C. Yang, "3S-cart: a lightweight, interactive sensor-based cart for smart shopping in supermarkets," *IEEE Sensors Journal*, vol. 16, no. 17, pp. 6774–6781, 2016.

[7] M. Ayaz, M. Ammad-uddin, I. Baig, and E. M. Aggoune, "Wireless sensor's civil applications, prototypes, and future integration possibilities: a review," *IEEE Sensors Journal*, vol. 18, no. 1, pp. 4–30, 2018.

[8] Y. C. Wang, W. C. Peng, and Y. C. Tseng, "Energy-balanced dispatch of mobile sensors in a hybrid wireless sensor network," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 12, pp. 1836–1850, 2010.

[9] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[10] Y. C. Wang, "Data compression techniques in wireless sensor networks," in *Pervasive Computing*. Hauppauge: Nova Science Publishers, 2012.

[11] Z. Sheng, H. Wang, C. Yin, X. Hu, S. Yang, and V. C. M. Leung, "Lightweight management of resource-constrained sensor devices in Internet of Things," *IEEE Internet of Things Journal*, vol. 2, no. 5, pp. 402–411, 2015.

[12] D. J. Persis and T. P. Robert, "Review of ad-hoc on-demand distance vector protocol and its swarm intelligent variants for mobile ad-hoc network," *IET Networks*, vol. 6, no. 5, pp. 87–93, 2017.

[13] W. D. Henderson and S. Tron, "Verification of the minimum cost forwarding protocol for wireless sensor networks," in *IEEE Conference on Emerging Technologies and Factory Automation*, 2006, pp. 194–201.

[14] H. Yoo, M. Shim, D. Kim, and K. H. Kim, "GLOBAL: a gradient-based routing protocol for load-balancing in large-scale wireless sensor networks with multiple sinks," in *IEEE Symposium on Computers and Communications*, 2010, pp. 556–562.

[15] F. Cadger, K. Curran, J. Santos, and S. Moffett, "A survey of geographical routing in wireless ad-hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 621–653, 2013.

[16] R. Vidhyapriya and P. T. Vanathi, "Energy aware routing for wireless sensor networks," in *IEEE International Conference on Signal Processing, Communications and Networking*, 2007, pp. 545–550.

[17] H. B. Fradj, R. Anane, M. Bouallegue, and R. Bouallegue, "A range-based opportunistic routing protocol for wireless sensor networks," in *IEEE International Wireless Communications and Mobile Computing Conference*, 2017, pp. 770–774.

[18] R. Singh, S. K. Gupta, and P. Khatri, "Residual energy based multipath routing protocol (REBM) in wireless sensor network," in *IEEE International Conference on Computing, Communication & Automation*, 2015, pp. 420–426.

[19] S. K. Singh, P. Kumar, and J. P. Singh, "A survey on successors of LEACH protocol," *IEEE Access*, vol. 5, pp. 4298–4328, 2017.

[20] Y. Z. Liang, A. L. Zhang, and Y. Z. Li, "An energy-effective routing protocol efficiently constructs cluster topology for WSN," in *International Conference on Instrumentation, Measurement, Computer, Communication and Control*, 2013, pp. 1097–1100.

[21] Z. Ullah, L. Mostarda, R. Gagliardi, D. Cacciagrano, and F. Corradini, "A comparison of HEED based clustering algorithms – introducing ER-HEED," in *IEEE International Conference on Advanced Information Networking and Applications*, 2016, pp. 339–345.

[22] P. Kumar, M. K. Banga, U. Rani, B. M. Thippeswamy, and K. R. Venugopal, "EBDRA: energy balanced dynamic cluster routing approach for WSN," in *IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology*, 2016, pp. 141–145.

[23] M. Alnuaimi, K. Shuaib, K. Alnuaimi, and M. Abed-Hafez, "An efficient clustering algorithm for wireless sensor networks," *International Journal of Pervasive Computing and Communications*, vol. 11, pp. 302–322, 2015.

[24] Y. C. Wang and C. T. Wei, "Lightweight, latency-aware routing for data compression in wireless sensor networks with heterogeneous traffics," *Wireless Communications and Mobile Computing*, vol. 16, no. 9, pp. 1035–1049, 2016.

[25] S. Misbahuddin, M. Tahir, and S. Siddiqui, "An efficient lossless data reduction algorithm for cluster based wireless sensor network," in *International Conference on Collaboration Technologies and Systems*, 2014, pp. 287–290.

[26] R. M. Buehrer, H. Wymeersch, and R. M. Vaghefi, "Collaborative sensor network localization: algorithms and practical issues," *Proceedings of the IEEE*, vol. 106, no. 6, pp. 1089–1114, 2018.

[27] Y. C. Wang and K. C. Chen, "Efficient path planning for a mobile sink to reliably gather data from sensors with diverse sensing rates and limited buffers," *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1527–1540, 2019.

[28] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The evolution of MAC protocols in wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 101–120, 2013.

[29] Y. C. Wang, C. C. Hu, and Y. C. Tseng, "Efficient deployment algorithms for ensuring coverage and connectivity of wireless sensor networks," in *IEEE Wireless Internet Conference*, 2005, pp. 114–121.

[30] Y. C. Wang, F. J. Wu, and Y. C. Tseng, "Mobility management algorithms and applications for mobile sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 1, pp. 7–21, 2012.