Characterization of Hello Message Exchange for Estimating Distribution of Network Residual Energy

Shudong Fang
Department of Electrical and Computer Engineering
University of Auckland, New Zealand
sfan028@aucklanduni.ac.nz

Stevan Mirko Berber
Department of Electrical and Computer Engineering
University of Auckland, New Zealand
s.berber@auckland.ac.nz

Akshya Kumar Swain
Department of Electrical and Computer Engineering
University of Auckland, New Zealand
a.swain@auckland.ac.nz

ABSTRACT
This paper investigates the practicability that a sensor node develops the probability density function (pdf) of its local network energy via exchanging hello messages with its neighboring nodes in the context of dense node deployment. The pdf is proven to approach Gaussian and can be used to decentralize a recent clustering algorithm. To alleviate the broadcast storm problem, a node is considered broadcasting hello messages to its neighboring nodes without immediate feedback from the receiving ones. Thus the broadcasting node cannot be guaranteed that its neighboring nodes have received its messages which are at high risk of channel collision. Characterizing hello message exchange becomes non-trivial, as the discovery ratio, which measures the effectiveness and the sufficiency of the message exchange, is identified having decisive effect on the precision of the developed pdf. A set of time asynchronous and slot-based channel access rules is presented for the sufficient and fast exchange of hello messages.

Categories and Subject Descriptors
C.2.1 [Computer Systems Organization]: Network Architecture and Design - *Wireless communication*;

General Terms
Algorithms, theory.

Keywords
Wireless sensor networks, Hello message, energy distribution.

1. INTRODUCTION
Ref [1] reports a wireless sensor networking algorithm SWEET, which uses the *distribution of network residual energy* to organize densely deployed sensor nodes into clusters. Such distribution is centrally calculated at the base station. This study is motivated to enhance the decentralization of SWEET using the *distribution of local network energy* independently developed by every node.

Regarding the nodes' energy dissipations as independent stochastic processes, this study proves that the *distribution of local network energy* developed by a node approaches the Gaussian distribution. Then we investigate the practicability that a node develops the empirical *pdf* of Gaussian distribution using the well-known hello message exchange [2, 3]. The idea seems simple: a node locally broadcasts the short hello messages, which contain the value of its residual energy, to its neighboring nodes; by reading the values of residual energy in received messages from neighboring nodes, this node may compute the parameter estimates for the *pdf* of the Gaussian distribution. The neighboring nodes are defined as a group of nodes that reside in the reliable radio range of each other.

However, the execution of hello message exchange is a non-trivial issue for dense sensor networks. The dense node deployment presents fundamental difficulties to effectively exchange the messages, due to the intrinsic issues of the broadcast channel access [4]: data (message) collision rate arises from the contention among many neighboring nodes which tend to concurrently send messages into the channel. We consider that the neighboring nodes which have received the message do not immediately acknowledge the sending node; otherwise, the collision may be aggravated in analogues to the broadcast storm problem [5]. Problem appears that the sending node cannot be sure that its messages have been received by its neighboring nodes since there is no feedback from the receiving ones. This problem closely resembles the issue of collision channel without feedback reported in wireless networking systems [6].

In [7], hello message exchange using a slot and time synchronous-based neighbour discovery protocol, i.e., Birthday protocol [8], which requires no feedback from receiving nodes, is documented. Findings show that the Birthday protocol can effectively conduct the message exchange at the expense of long time duration and large node energy to achieve high *discovery ratio* in dense node deployments. The *discovery ratio* is defined as a ratio of the number of neighboring nodes which have received the message from the sending node to the total number of neighboring nodes that this sending node has.

In this paper, the discovery ratio is identified having decisive effect on the precision of the parameter estimates of the needed *pdf*. To accomplish a high discovery ratio fast and energy-efficiently, this paper presents a set of channel access rules which are modified from the solution for the initialization problem [9]. The proposed rules have these properties: (i) they are based on slotted time intervals; (ii) they are time asynchronous-based: each node senses the channel for a random period, and then broadcasts hello message with a probability which is configured to minimize the time duration of the message exchange; (iii) feedback to the sending node is not sent immediately by multiple receiving nodes,

IWCMC'09, June 21–24, 2009, Leipzig, Germany.

but piggybacked in the hello message of the next sending node; (iv) a node may broadcast its message multiple times.

The above rules are abbreviated as CSMS. This study also contributes the characterization of hello message exchange using CSMS. The duration of message exchange and energy that a node spends in the course of message exchange are analyzed as functions of key attributes, to understand the time and energy required to achieve arbitrary discovery ratio in various network settings. Simulations of CSMS are carried out to show that the time and energy expenses on achieving a high discovery ratio can be significantly reduced, in comparison to message exchange using Birthday protocol. The decentralized SWEET algorithm is evaluated with respect to imperfect discovery ratios (<100%).

2. PRELIMINARIES

2.1 Network model

There are N-number of identical sensor nodes that are randomly and uniformly deployed in a square area A., which yields the network density $\lambda=N/A$. Nodes are battery-powered. All the nodes have the same initial energy e_0. The residual energy of node n_i, i=1,2,…,N, at time t is denoted by $e_i(t)$, $0 \le e_i(t) \le e_0$. Every node is equipped with an Omni-directional antenna and a transceiver which work in half-duplex mode. In the center of A sits one powerful data sink, to which all the nodes report data.

The transmission range of the transceiver is denoted as d_{TR}. The transceiver can transit among three states: transmit (T), listen/receive (L) and sleep (S). Powers of the transceiver in state T, L and S are denoted by P_t, P_l and P_s, respectively. The value of P_s is often disregarded ($P_s \approx 0$), for P_s is much smaller than P_t and P_l. A transmitting node adjusts its transmitting power P_t sufficiently large, e.g., adding enough fading margin, such that receiving nodes on the edge of d_{TR} has high availability of receiving the data, provided that no channel collision occurs during the data transmission.

2.2 Timeline of system operation

The timeline of system operation is shown in Fig. 1. The system operates in rounds. Each round constitutes the following sequence of events: node synchronization, hello message exchange, SWEET algorithm, and data uploading. The associated time intervals for processing each of these events are denoted by T_{syn}, T_{nd}, T_{delay_frame} and T_{steady} in turn.

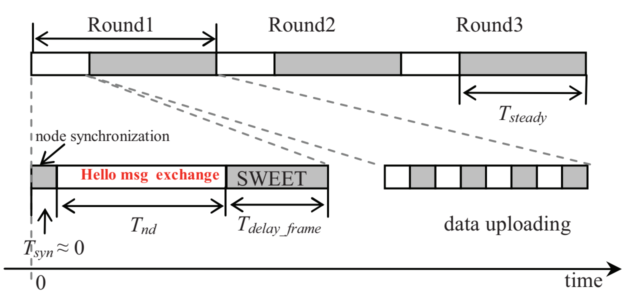


Fig. 1. Timeline of system operation

Node synchronization: after T_{syn}, all the N nodes are assumed to be synchronized to the beginning of T_{nd}.
Hello message exchange: every node, n_i, locally broadcasts hello messages which contain the value of its residual energy, $e_i(t)$, to neighboring nodes within its radius d_{TR}. Node n_i also receives the hello messages from neighboring nodes to calculate the empirical *pdf* of the local network energy distribution. Node n_i does not know the exact amount of neighboring nodes, but it is assumed to know the network density λ prior. Execution of the hello message exchange using the proposed channel access rules in the time interval T_{nd} will be explained in Section IV.
SWEET and data uploading: the decentralized SWEET algorithm using the distribution of local network energy is conducted by nodes in T_{delay_frame}. Running SWEET algorithm, nodes are organized in the form of clusters and upload data to the data sink. Due to the limited space of the paper, issues related to the decentralization of SWEET and data uploading are omitted. Yet, effects from the periodic course of message exchange on network nodes running the decentralized SWEET will be presented in the simulation results in Section VI.

3. LOCAL NETWORK ENERGY

3.1 Node's local network energy

Because the energy dissipation of each node depends on multiple random factors [10], the residual energy of node n_i, $e_i(t)$, may be regarded as a stochastic process which drops down from e_0 to zero. Hence $e_i(t)$ at time t is a random variable (RV) denoted by $E_i(t)$, where $0 \le E_i(t) \le e_0$, t may takes values as, e.g., the time point when a round begins.
Definition: node n_i's *distribution of local network energy* is defined as a sample mean of the residual energy of n_i's neighboring nodes at time t, and can be expressed as:

$$E^i_{nb}(t) = (E_1(t)+ E_2(t)+\ldots+ E_n(t))/n,$$

where $n = \lfloor \pi\lambda d^2_{TR} \rfloor \le N$ is the number of n_i's neighboring nodes including node n_i itself. The sign $\lfloor x \rfloor$ means the smallest integer equal to or larger than x.
Proposition: $E^i_{nb}(t)$ is a RV that approaches Gaussian distribution, provided that n is sufficiently large.
Proof: Because the correlations of energy consumption among nodes are minor, the residual energies of nodes may be regarded independent. At time t, $E_i(t)$, for i=1,2,…,N, may have different distributions in nature. The expectations and variances of these $E_i(t)$, which are denoted by $\mathrm{E}[E_i(t)]$ and $\mathrm{D}[E_i(t)]$, respectively, may also be different. Since the variances $\mathrm{D}[E_i(t)]$ (i=1,2,…,N) are all finite, according to the generalized Central Limit Theorem (CTL) [11], $E^i_{nb}(t)$ is a RV converging to the Gaussian distribution, provided that n is large. $E^i_{nb}(t)$ has mean $\mathrm{E}[E^i_{nb}(t)] = \sum_{i=1}^{n} \mathrm{E}[E_i(t)]/n$, and variance $\mathrm{D}[E^i_{nb}(t)] = \sum_{i=1}^{n} \mathrm{D}[E_i(t)]/n^2$.

Node n_i may independently calculate the estimates of $\mathrm{E}[E^i_{nb}(t)]$ and $\mathrm{D}[E^i_{nb}(t)]$, using values of the residual energy of n_i'neighboring nodes via message exchanges as explained next.

3.2 Parameter estimation

The unbiased estimations of $\mathrm{E}[E^i_{nb}(t)]$ and $\mathrm{D}[E^i_{nb}(t)]$ may be calculated using the following formulas.

$$\mathrm{E}[E^i_{nb}(t)] = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} e_i(t) \,, \tag{1}$$

$$\mathrm{D}[E^i_{nb}(t)] = \sum_{i=1}^{\tilde{n}} (e_i(t) - \mathrm{E}[E^i_{nb}(t)])^2 / (\tilde{n} - 1) \,. \tag{2}$$

where $i \in [1, \tilde{n}]$, $\tilde{n} = \lfloor p_{dr}\, n \rfloor$, $p_{dr} \le 1$ denotes the *discovery ratio* which is defined as a ratio of the number of neighboring nodes from which node n_i has received messages to the total number of n_i's neighboring nodes, i.e., n.

One can observe from (1, 2) that p_{dr} has decisive effect on the accuracy of E[$E^i_{nb}(t)$] and $\mathrm{D}[E^i_{nb}(t)]$. The right-hand sides of (1) and (2) converge to the theoretical mean and variance, provided that $\tilde{n} \to \infty$. Note that in practice, n is finite. To increase the accuracy of the estimated mean and variance, hello messages need to be effectively exchanged to make p_{dr} approximate 1. Next, we will explain the proposed channel access rules which are designed to render fast, effective and sufficient message exchange.

4. CHANNEL ACCESS RULES

4.1 Time interval for channel access rules

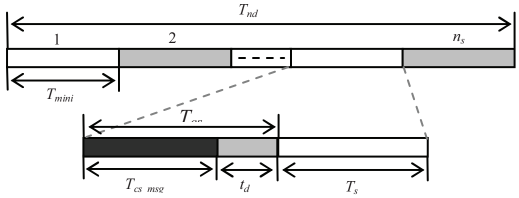


Fig. 2 Slotted time interval for the course of Hello message exchange

The structure of the slotted time interval for the course of hello message exchange is shown in Fig. 2. The time interval T_{nd} is divided into n_s-number of mini-slots. Each mini-slot lasts for T_{mini}. A slot T_{mini} consists of a carrier sensing period T_{cs} and a message transmission period T_s. Interval T_{cs} consists of T_{cs_msg} and t_d, where T_{cs_msg}=$w_t t_d n$, w_t is a weighting factor, t_d is the signal transmission delay. The message transmission interval T_s is equal to l_{msg}/R_b, where l_{msg} is the length of a Hello message and R_b is the data speed. The numerical relationship of these parameters is expressed as

$$T_{\text{mini}} = T_{cs} + T_s = (T_{cs\_msg} + t_d) + l_{msg} / R_b . \qquad (3)$$

4.2 Structure of a hello message

The structure of the hello message in use has three sections, i.e., ID of the sending node, the residual energy of the sending node and a feedback (ACK). The ACK is used to piggyback the ID of the latest node whose message has been received.

4.3 Channel access rules

The proposed channel access rules for exchanging hello messages are summed up in the following five steps within one mini-slot time interval T_{mini}.

Step 1: The n-number of neighboring nodes are assumed to be synchronized at the beginning of every T_{mini}. We will show that this assumption is reasonable based on the simulation results, which confirms that the required T_{nd} is very short under CSMS. Thus, the clocks of nodes may have no significant drift in T_{nd} and may keep being synchronized in every T_{mini}, provided that nodes have been synchronized to the beginning of T_{nd}.

Step 2: Each node, say nodes n_i, launches a waiting timer $t^i_{cs\_msg}$. Node n_i assigns a random value to $t^i_{cs\_msg}$. The random value is uniformly selected from an interval [0, T_{cs_msg}]. Node n_i listens to the channel for a period of $t^i_{cs\_msg}$.

Step 3 (case 1): if node n_i detects the message on channel in $t^i_{cs\_msg}$, it will receive this message and then do the following: (i) update the content of a buffer called AP with the ID of node whose message has been just received; (ii) read out the value of residual energy; (iii) check the content of ACK section. If the ID in ACK matches the ID of node n_i, node n_i will not broadcast any message in the rest period of T_{nd}. Then n_i turns the radio down and waits for the next mini-slot T_{mini}.

Step 3 (case 2): If no message is detected at the expiry of $t^i_{cs\_msg}$, node n_i decides to releases its hello message with a probability p_t.

Step 4 (case 1): If node n_i decides to broadcasts it messages with the probability p_t, it fills the message with its ID, its $e_i(t)$ and the content of buffer AP, and then sends this message. When the transmission is completed, node n_i shuts down the radio front and wait for the next mini-slot T_{mini}.

Step 4 (case 2): If node n_i decides not to broadcast with probability (1-p_t), it continues to sense the channel in the rest time period of (T_{cs_msg} - $t^i_{cs\_msg}$).

Step 5: If a message is detected during the period (T_{cs_msg} - $t^i_{cs\_msg}$), node n_i receives the message by following the same procedure as described in **Step 3 (case 1)**. If no message is detected during the interval (T_{cs_msg} - $t^i_{cs\_msg}$), node n_i shuts down its radio front and wait for the next mini-slot T_{mini}.

The proposed set of channel access rules is termed the Carrier Sensing Mini-Slot access (CSMS). Note that if a node successfully broadcasts its message in one T_{mini}, all of its neighboring nodes can sense and receive this message. Hence, in step 3 if a node finds its ID matches the content of ACK in the currently received message, this node is advised that its message have been received by all the neighboring nodes. Then, this node does not need to broadcast its message any more. In the next section we explain the configuration of the probability p_t to minimize the duration of T_{nd}. The time and energy costs on exchanging Hello message using CSMS are also analyzed.

5. ANALYSIS OF CSMS

5.1 Duration of message exchange by CSMS

The operation of CSMS in essence is a Bernoulli trial in a binominal distribution. Let $t^1_{cs\_msg}, t^2_{cs\_msg}, \ldots, t^i_{cs\_msg}, \ldots,$ $t^n_{cs\_msg}$ denote the random delays that n-number of nodes spend on sensing the carrier from the start of a mini-slot. The values of these delays are put into a non-decreasing order $\{t^{1:n}_{cs\_msg}, t^{2:n}_{cs\_msg}, \ldots, t^{i:n}_{cs\_msg}, \ldots, t^{n:n}_{cs\_msg},\}$, where $t^{1:n}_{cs\_msg}$ denotes the smallest delay, $t^{2:n}_{cs\_msg}$ is the second smallest delay, $t^{n:n}_{cs\_msg}$ is the longest delay. In this mini-slot, the probability that $\hat{n}$-number of nodes ($\hat{n} \le n$) decide to send their messages with the probability p_t can be formulated as $\Pr(\hat{n}) = \binom{n}{\hat{n}} p_t^{\hat{n}} (1 - p_t)^{n - \hat{n}}$. Assume node n_i is one of these nodes. n_i can succeeds in broadcasting its message, provided that: (i) $t^i_{cs\_msg}$ of n_i is equal to $t^{1:n}_{cs\_msg}$; (ii) the shortest delay among the rest $\hat{n}-1$ nodes is $t^{2:n}_{cs\_msg}$ which satisfies $t_{cs\_msg}^{2:n} - t_{cs\_msg}^{1:n} \ge t_d$. In such a way, if node n_i decides to broadcast its message, the neighbor nodes of n_i can sense the message in t_d. In a mini-slot, all of its neighbor nodes can detect its message. If node n_i succeeds in broadcasting its message in a mini-slot, all of its neighbor nodes can receive it message. The probability that a node successfully broadcasts its message in a mini-slot can be expressed as

$$\Pr(p_t, n, t_d) = \sum_{\hat{n}=1}^{n} \binom{n}{\hat{n}} p_t^{\hat{n}} (1-p_t)^{n-\hat{n}} \Pr(y - x \ge t_d). \quad (4)$$

where $x = t_{cs\_msg}^{1:n}$ and $y = t_{cs\_msg}^{2:n}$. Note that both x and y are RVs in uniform distribution in interval [0, T_{cs_msg}]. The function $\Pr(y - x \ge t_d)$ can be computed by using the joint *pdf* of x and y, $f_{1,2:n}(x, y)$, in the following expression

$$f_{1,2:n}(x, y) = \hat{n}(\hat{n}-1)(1 - y / T_{cs\_msg})^{\hat{n}-2} / T_{cs\_msg}^2. \quad (5)$$

The probability of $y - x \ge t_d$ can be computed as below:

$$\Pr(y - x \ge t_d) = \int_0^{T_{cs\_msg} - t_d} \int_{x+t_d}^{T_{cs\_msg}} f_{1,2:n}(x, y) dy dx$$

$$= \begin{cases} 1, & \hat{n} = 1; \\ (1 - 1/(w_t n))^{\hat{n}}, & \hat{n} \in [2, n). \end{cases} \quad (6)$$

Substituting (6) into (4) yields

$$\Pr(p_t, n, t_d) = \sum_{\hat{n}=2}^{n} \binom{n}{\hat{n}} p_t^{\hat{n}} (1-p_t)^{n-\hat{n}} (1 - \frac{1}{w_t n})^{\hat{n}} + n p_t (1-p_t)^{n-1}. \quad (7)$$

The event that a node successfully broadcasts message in a mini-slot follows a binomial distribution $B(n_s, \Pr(p_t, n, t_d))$. Let p_{desire} denote the confidence that the prescribed p_{dr} can be attained by the end of T_{nd}. Then $T_{nd}=n_s T_{mini}$ where n_s can be calculated using the Chernoff Bound [5]:

$$n_s = \frac{p_{dr} n - \log(1-p_{desire}) + \sqrt{\log^2(1-p_{desire}) - 2 p_{dr} n \log(1-p_{desire})}}{\Pr(p_t, n, t_d)}. \quad (8)$$

5.2 Optimization of p_t

Parameter p_t is configured to minimize the duration of $T_{nd}=n_s T_{mini}$. It can be observed from (3), (4) and (8) that T_{nd} is dependent on p_t, n, l_{msg}, R_b, t_d, p_{dr}, p_{desire}, and w_t. Suppose the parameters of network settings, λ, d_{TR}, l_{msg}, R_b, t_d, p_{dr}, p_{desire} and w_t are known by nodes *a priori*. Node n_i can independently optimize p_t to minimize the duration of message exchange T_{nd}, using numerical methods.

5.3 Analysis of energy expense

During T_{nd}, node n_i broadcasts its message $1/\Pr(p_t, n, t_d)$ times. The total energy that n_i spends on broadcasting these hello messages is denoted by e_{tmsg}^i and computed as

$$e_{tmsg}^i = P_{tx} T_s / \Pr(p_t, n, t_d). \quad (9)$$

The total energy that n_i spends on receiving messages and sensing the channel is denoted by e_{rmsg}^i and computed as

$$e_{rmsg}^i \approx P_{rx}(\sum_{n_s} t_{cs\_msg}^{1:n} + n_s T_s) \Pr(p_t, n, t_d) + P_{rx} n_s T_{cs} (1-p_t)^n +$$
$$P_{rx}(\sum_{n_s} t_{cs\_msg}^{1:n} + n_s T_s)(1 - \Pr(p_t, n, t_d) - (1-p_t)^n), \quad (10)$$

where $\sum_{n_s} t_{cs\_msg}^{1:n} < n_s T_{cs\_msg}$ stands for the total amount of delay for carrier sensing in all the successful mini-slots. Thus

$$e_{rmsg}^i \le P_{rx} n_s (T_{cs\_msg} + T_s) \Pr(p_t, n, t_d) + P_{rx} n_s T_{cs} (1-p_t)^n +$$
$$P_{rx} n_s (T_{cs\_msg} + T_s)(1 - \Pr(p_t, n, t_d) - (1-p_t)^n). \quad (11)$$

Thus the total energy consumption of node n_i denoted by e_{msg}^i is a sum of these two energy costs: $e_{msg}^i = e_{tmsg}^i + e_{rmsg}^i$.

6. NUMERICAL RESULTS

A comparative study of hello message exchange using CSMS and Birthday protocol [6], is conducted via simulations to investigate the performances of each method. Then, CSMS and Birthday protocol are integrated to decentralize SWEET.

6.1 CSMS and the Birthday protocol

To message exchange using CSMS and Birthday protocol, the same transceiver power consumption model is used. The transmission power P_t and the listening/receiving power P_l are calculated as follows

$$P_t = (E_{elec} + E_{amp} d_{TR}^2) R_b / 10^6, \; P_t = E_{elec} R_b / 10^6. \quad (12)$$

Eq. (12) is adopted from the energy consumption model in [1]. This model is also used by the decentralized SWEET algorithm, so that the entire system is evaluated with a consistent energy consumption model. E_{elec} is the energy consumed by the circuitry of the transceiver to transmit one bit at the data rate 1Mbps. E_{amp} is the energy consumed by the amplifier of the transmitter to transmit one bit to a unit distance at the data rate 1Mbps. The values of E_{elec} and E_{amp} are set to 5×10^{-8} and 9×10^{-12}, respectively[1]. d_{TR} takes the value of 40 meters.

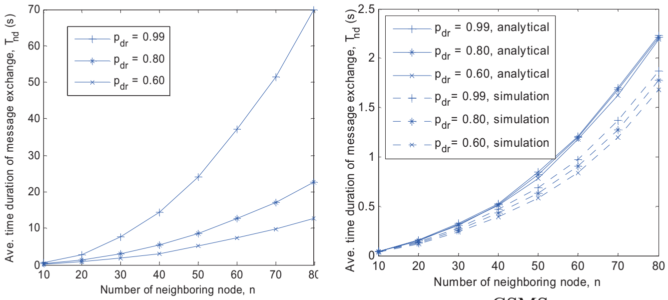
According to Eq. (3), (8), and (11), T_{nd} and e_{msg}^i are dependent on several key parameters: n, p_{dr}, $p_{dersire}$, l_{msg}, R_b and w_t. To investigate the dependence of T_{nd} and e_{msg}^i on each of these parameters, three groups of simulations are conducted. In each group, two parameters alter and others stay constant.

In the first group, n is increased from 10 to 80, p_{dr} is increased from 0.50 to 0.99, p_{desire} is set to 0.99, R_b is set to 1Mbps and l_{msg} is set to 150 bytes. In the second group, n is increased from 10 to 80, l_{msg} is increased from 100 bytes to 200 bytes, R_b is set to 1Mbps, p_{dr} and p_{desire} are set to 0.99. In the third group, n is increased from 10 to 80, R_b is increased from 256 kbps to 1Mbps, l_{msg} is set to 150 bytes, p_{dr} and p_{desire} are set to 0.99.

For the brevity of presentation, the graphic results of the first group of simulations are shown in Fig. 3. These results are the averaged outcomes of 1000 repeated simulations. In the sequel, the observation of these results, together with the observations of simulation results of the rest two groups are summarized.

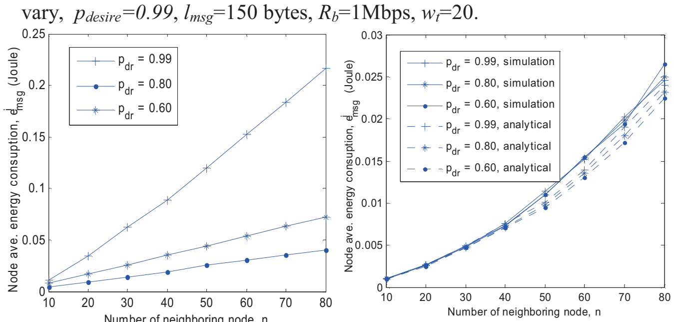
Figure 3 shows that message exchange using CSMS needs much shorter time and charges a node less energy to attain given p_{dr} at the investigated n. For example, for p_{dr}=0.99 and n=40, T_{nd} and e_{msg}^i needed by CSMS is less than 0.6 second and 0.007 Joule, while T_{nd} required by Birthday protocol is about 15 seconds and

0.09 Joule. When n is increased to 80, T_{nd} needed by CSMS is less than 2.5 seconds. Similar observations can be concluded from the studies of the second and the third groups that CSMS changes much less energy and time than Birthday protocol does.



Birthday protocol CSMS

Fig. 3 (a). Time duration of the message exchange T_{nd} in one round, n and p_{dr} vary, $p_{desire}=0.99$, l_{msg}=150 bytes, R_b=1Mbps, w_t=20.



Birthday protocol CSMS

Fig. 3 (b). Node n_i's energy consumption e^i_{msg} for the message exchange in one round, n and p_{dr} vary, $p_{desire}=0.99$, l_{msg}=150 bytes, R_b=1Mbps, w_t=20.

6.2 Performances of decentralized SWEET

Hello messages periodically are exchanged for a node to develop the distribution of local network energy for the execution of the decentralized SWEET algorithm. We investigate the network lifetime influenced by the node's energy consumption during the course of an imperfect hello message exchange, i.e., p_{dr}<1.

Simulations are carried out using the following network settings: N=200 nodes are randomly and uniformly deployed in 10^4 square meter area. Each node is initialized with e_0=2J energy. The transmission range for Hello message exchange is set to d_{TR}=40 meters, so that every node on average has n=40 neighboring nodes. The length of hello message is set to 150 Bytes. The parameters for the decentralized SWEET algorithm take corresponding values from [1].

Fig. 4 presents the lifetime of network nodes running the decentralized SWEET after imperfect message exchange that employs Birthday protocol and CSMS, respectively. Every result is the average of 30 repeated simulations. In literature, the lifetime is usually defined as the last round when half of the nodes die (HND) or all the nodes die (AND). One can observe that the network lifetime is influenced by the course of message exchange, with respect to p_{dr}<1. When p_{dr} increases, the network lifetime is reduced, because node has to spend more energy on message exchange that employs either Birthday protocol or CSMS. The lifetime of a network employing CSMS to exchange hello message is much longer than the lifetime of a network using Birthday protocol for all the studied p_{dr}.

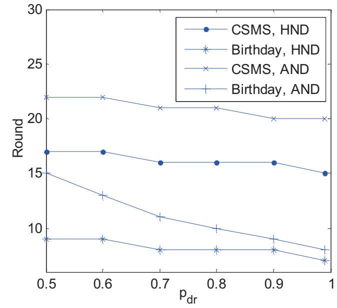


Fig. 4. Lifetime of network running decentralized SWEET imperfect message exchange, with respect to the increasing discovery ratio p_{dr}<1. The data uploading phase in a round lasts for 15 seconds.

7. CONCLUSIONS

The hello message exchange is an effective approach to develop the *pdf* of local network residual energy distribution. We proved that such distribution approaches the Gaussian distribution in dense node deployments. To find the empirical *pdf*, we propose a set of channel access rules (CSMS) to render fast and energy efficient hello message exchange. Characterizations of CSMS, in particular the time and energy costs, are analyzed as functions of the number of neighbor nodes, the discovery ratio, the message length and the data rate. Simulation results confirm the theoretical analysis and show that CSMS outperforms Birthday Protocol in terms of shorter time duration and smaller energy cost for given discovery ratio. Simulations show that the lifetime of network running the decentralized SWEET is reduced if the hello message exchange, which utilizes either CSMS or Birthday protocol, needs to accomplish high discovery ratios. The results suggest that the decentralized SWEET is capable of organizing densely deployed sensor nodes at fairly low costs of time and energy in the hello message exchange if CSMS is employed.

8. REFERENCES

[1] Fang, S., Berber, S. M., and Swain, A. K., "An overhead-free clustering algorithm for wireless sensor networks", in *Proc. GLOBECOM'07,* Washington DC, USA, 2007.

[2] Younis, O., and Fahmy, S., "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks", *IEEE Trans. on Mobile Computing,* vol. 3, pp. 366-79, 2004.

[3] Hua, C., and Yum,T., "Data aggregated maximum lifetime routing for wireless sensor networks", *Ad Hoc Network*s, vol.6, no.3, pp. 380-92, 2008.

[4] Cai, Z., Liu, M., and Wang, X., "Randomized Broadcast Channel Access Algorithm for Ad Hoc Networks", in *Proc. ICPP'*02, 2002.

[5] Tseng, Y., , Ni, S., Chen,Y., and Sheu, J.,"The Broadcast Storm Problem in a Mobile Ad Hoc Network", *Wireless Networks*, vol. 8, no. 2-3, pp 153-167, 2002.

[6] Massey, J., and Mathys, P., "The Collision Channel Without Feedback", *IEEE Trans. on Infor. Theory*, vol. IT-31, no. 2, 1985.

[7] Fang, S., Berber, S. M., and Swain,A. K., "Analysis of Neighbor Discovery Protocols for Energy Distribution Estimations in Wireless Sensor Networks", in *Proc. ICC'08*, Beijing, China, 2008.

[8] McGlynn, M. J., and Borbash,S. A., "Birthday *protocols* for low energy deployment and flexible neighbor discovery in ad hoc wireless networks", in *Proc. MOBIHOC'01*, 2001, pp. 137-45.

[9] Cai, Z., Lu, M. and Wang, X., "Distributed Initialization Algorithm for Single-Hop Ad Hoc Networks with Minislotted Carrier Sensing", *IEEE Trans. on Parallel and Distributed Systems*, vol. 14, no. 5, pp. 516-28, 2003.

[10] Zhu, J., and Papavassiliou, S.,"On the Energy-Efficient Organization and the Lifetime of Multi-hop Sensor Networks", *IEEE Communi. Letters*, vol. 7, no. 11, pp. 537-9, 2003.

[11] Spiegel, M. R., *Theory and Problems of Probability and Statistics*. New York: McGraw-Hill, pp. 112-113, 1992.