Low-Latency Geographic Routing for Asynchronous Energy-Harvesting WSNs

Donggeon Noh, Ikjune Yoon, Heonshik Shin
School of Computer Science and Engineering, Seoul National University, Seoul, Korea
Email: {dgnoh, ijyoon}@mobisys.snu.ac.kr, shinhs@snu.ac.kr

***Abstract*— Research on data routing strategies for wireless sensor networks (WSNs) has largely focused on energy efficiency. However rapid advances in WSNs require routing protocols which can accommodate new types of energy source and data of requiring short end-to-end delay. In this paper, we describe a duty-cycle-based low-latency geographic routing for asynchronous energy-harvesting WSNs. It uses an algorithm (D-APOLLO) that periodically and locally determines the topological knowledge range and duty-cycle of each node, based on an estimated energy budget for each period which includes the currently available energy, the predicted energy consumption, and the energy expected from the harvesting device. This facilitates a low-latency routing scheme which considers both geographic and duty-cycle information about the neighbors of a node, so that data can be routed efficiently and delivered to the sink as quickly as possible. Simulation results confirm that our routing scheme can deliver data to the sink with high reliability and low latency.**

***Index Terms*— wireless sensor network, duty cycle, geographic, routing, low-latency, energy harvesting**

I. INTRODUCTION

Sensor networks are typically composed of hundreds to thousands of small collaborating wireless sensor nodes that have limited computation and communication capabilities. The need for ease of deployment and the requirement for many nodes to be deployed makes it difficult, or impossible, to provide an external power supply or to replace depleted batteries. The life of the battery at each sensor node then determines how long the entire network can remain active. Consequently, research on data routing strategies for wireless sensor networks (WSNs) has largely focused on energy efficiency [1], [2]. But rapid advances in WSNs mean that routing protocols are now required for timely delivery of sensing data with the help of semi-permanently deployed sensor. For example, applications such as critical condition monitoring and security surveillance require durable power source and are highly sensitive to delay. In order to efficiently overcome these challenges, WSNs need new types of power source and low-latency routing schemes which can accommodate them.

Energy harvesting, or scavenging, is a technology that has recently been introduced to overcome the short running time available from batteries. By generating electricity from environmental energy, such as solar energy, temperature variations, kinetic energy, or vibrations, the dependency on batteries can be reduced or even eliminated [3]–[5]. However, little attention has yet been given to routing within a network of nodes running on environmental energy.

Once a sensor has been deployed, it must operate autonomously as far as possible. Thus, each sensor node in an energy-harvesting WSN should be aware that the amount of environmental energy it can gather will depend on the time, and its location and surroundings; and this awareness needs to be reflected in its pattern of operation. If a WSN is powered by energy-harvesting devices, knowledge of their characteristics should be incorporated in its routing scheme. So far, however, relatively few studies [6], [7] have been devoted to this topic.

We will propose low-latency routing schemes based on a harvested energy model. We have been concerned with these problems in earlier work [8]–[11]. We are now able to expand and elaborate on the ideas behind our earlier work. Our target network and a basic routing scheme are described below:

- **Asynchronous energy-harvesting WSN**: The quantity of harvested energy from the environment can be very different from node to node. Therefore, each node must determine appropriate duty cycle (DC) [1] with which a sensor node can operate perpetually. Therefore each node wakes up at a different time due to the different DC.
- **Geographic routing scheme**: It has been shown [12] that routing protocols should use geographic location information to support scalability and mobility. And recent research [13], [14] confirms that geographic routing algorithms are more suitable for power-constrained networks. In this scheme, routes are determined locally by each node based on information about its neighbors and the sink. The topological extent of this information is the knowledge range (KR), and determining the KR of each node is the most important problem in geographic routing: a larger KR is likely to produce a more nearly optimal path, but gathering and maintaining more topological information requires more energy.

With based on these technologies, we developed duty-

This paper is based on "Priority-based routing for solar-powered wireless sensor networks," and "Low-latency routing for energy-harvesting sensor networks" by D. Noh, H. Kwon, J. Kim, J. Lee, D. Lee and H. Shin, which appeared in the ISWPC'07 and UIC'07 respectively

[1]Duty cycle means the proportion of time for which a node is active.

cycle-based low-latency routing scheme for asynchronous energy-harvesting WSNs. The major contributions of our work are summarized below:

- **Maximum utilization of harvested energy** The proposed D-APOLLO algorithm periodically updates both the duty cycle and the KR of a node so as to route data efficiently, while keeping within the expected energy budget of the energy-harvesting device.
- **Low-latency routing protocol** A routing algorithm which considers both KR and DC can reduce the end-to-end delay to the sink. We have therefore proposed low-latency routing protocol, based on a geographic routing scheme, which take account of DC information of the neighbors.

The rest of this paper is organized as follows. The next section contains an analysis of existing routing protocols for energy-harvesting WSNs. Section III explains how D-APOLLO chooses the KR and DC to maximize utilization of the harvested energy and Section IV describes our low-latency routing strategies based on geographic information and DC information of the neighbors. We then give an overview of our simulation environment and evaluate our strategies in Section V. Finally, conclusions are drawn in Section VI.

II. RELATED WORK

A. Routing for Energy-Harvesting WSNs

In the literature, most studies of energy-aware routing focus on residual battery status and do not take into account the environmental energy availability at the nodes. It is all that Willig et al. have developed a routing protocol that considers nodes with permanent power supply [15].

Moreover, even though a lot of works have been put into the design and development of solar-powered sensor nodes, only a few makeshift topologies and routing protocols have been implemented.

At a time when energy-harvesting techniques were less attractive, there was some research [16], [17] on integrating a small number of solar-powered nodes into existing routing protocols. A small number of harvesting nodes were added to an otherwise battery-powered sensor network. The well-known routing protocols, LEACH [18] and Directed Diffusion [19] were modified to prolong network activity by placing heavier workloads on to energy-harvesting nodes. But such a network cannot, of course, keep operating indefinitely, because most of the nodes in the network are battery-powered. Gradients are used to provide additional information about the state of neighbors, which may be running on solar power or relying on their batteries.

The first full-scale approach to the utilization of environmental energy for routing [20] demonstrated that environmentally aware decisions improve performance compared to decisions based only on battery status, although the application scenario was limited.

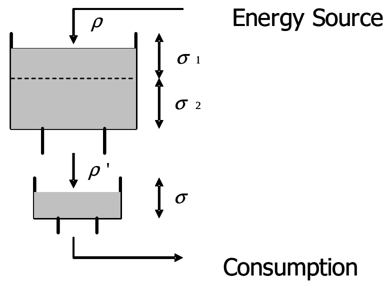

Figure 1: Energy flow from source to consumer

Then, in the Heliomote [21] project, perpetuity of operation was considered in the context of task management, network topology and the routing protocol. The UCLA team's prototype harvesting node, itself called 'Heliomote', suggested ways to model energy harvesting and consumption numerically, and resulted in a scheme for indefinite operation.

B. Duty Cycle for Perpetual Operation

The quantity of energy harvested from the environment can be very different from node to node due to the diversity of harvesters, the locations of the nodes, the deployment policy, the rate of harvesting, and so on.

Kansal et al. [6] suggested a mathematical condition which would express the conditions under which a sensor node can operate perpetually, through an analysis of the relationship between harvested and consumed energy as shown in Fig. 1.

Suppose $E(t)$ is the power output of any energy source at time t, then the energy harvested during any finite time T can be estimated as follows:

$$\rho T - \sigma_1 \leq \int_T E(t)dt \leq \rho T + \sigma_2. \tag{1}$$

Therefore, if there is enough historical data, we can make a good estimate of the energy that will be supplied to a node during a given period.

$E_c(t)$ is the power consumption of a node at time t and satisfies the following constraint for any value of T:

$$\int_T E_c(t)dt \leq \rho' T + \sigma. \tag{2}$$

Each node in an energy-harvesting network must have a sensible plan to control the energy that it consumes by means of an energy budget based on adequate historical data. That means that a node must have an energy storage capacity of more than $\sigma + \sigma_1 + \sigma_2$ and satisfy $\rho' < \rho$ in order to operate perpetually. Since the energy consumption of each active mode is fixed, the duty cycle d of a perpetually operating node completely depends on the calculated value of ρ' [6].

C. Determination of KRs in geographic routing

PRADA [12] is the well-known KR-tracing algorithm. It determines KRs which allow each node to minimize its energy consumption. It has been shown [22] that the

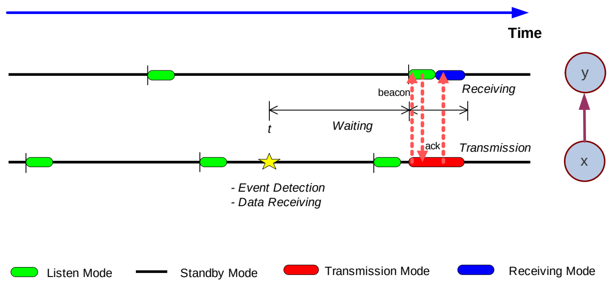

Figure 2: Data transfer sequence in asynchronous WSNs

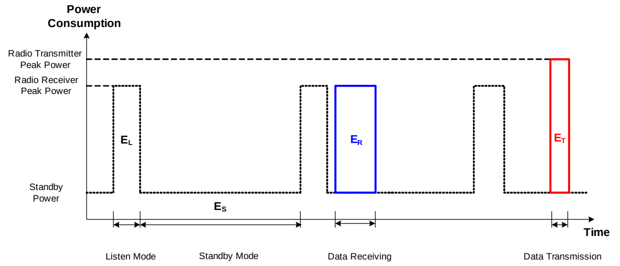

Figure 3: Power consumption of a node in various modes

determination of KRs is an NP-hard problem, so PRADA deploys a distributed real-time algorithm to achieve an approximate solution, using the following procedure: Each node that changes its KR transmits a probe packet to the sink; nodes receiving the probe packets estimate the energy required to process them, and then transmit that information to the source along the reverse path; and the source uses this information to calculate the energy needed to deliver a packet to the sink with that KR value. By varying its KR, each node can determine a near-optimal value which approximates the lowest energy consumption.

This method concentrates on minimizing the energy drain on each battery, and it does not consider any QoS metric, including end-to-end delay. It also incurs a large overhead whenever the KRs must be recalculated due to topological changes. Furthermore, since each node uses information from other nodes to determine its own KR, PRADA cannot support large-scale WSNs composed of hundreds of nodes.

III. Determining KR and DC for Low Latency

Correct determination of the knowledge range (KR) of each node is very important in geographic routing. A larger KR is likely to produce a more nearly optimal path, but gathering and maintaining more topological information requires more energy. The duty cycle (DC) is also important in low-latency routing for asynchronous WSNs. If a node has a large duty cycle, it can route data more quickly, but it also needs more energy to wake up frequently. Because the quantity of periodically harvested energy is limited, nodes must determine KR and DC prudently. In this section, we introduce D-APOLLO (Duty-cycle-based Adaptive toPOLogical KR aLgOrithm).

Before looking at the determination of the KR and DC, we will consider the data transfer sequence in asynchronous WSNs. As shown in the Fig. 2, when a node x senses a new event or receives data, it chooses the appropriate neighbor node (in Fig. 2, it is node y) to which the data will be delivered. Then it waits for node y to wake up. When that happens, it sends a beacon message to node y, and starts to transmit data. So we see that data is received immediately after the receiving node wakes, but the transfer of data is performed when the target node wakes.

A. Choosing the KR and DC to Maximize Utilization of the Harvested Energy

The first step taken by D-APOLLO is to find (KR, DC) pair which will maximize utilization of the harvested energy.

Like APOLLO [8], D-APOLLO decides whether the available energy E_{avail}[2] is more than the threshold E_{allow} at the beginning of each interval, as follows:

$$E_{avail} = E_P + E_C - E_{allow}, \tag{3}$$

where E_{avail} is the energy available during the next period, E_P is the amount of energy currently remaining, E_C is the energy expected from the harvesting device over the next period, and E_{allow} is an allowance for incorrect assumptions.

As we can see in Fig. 3, the power consumption can be classified into that required for four different tasks; receiving data (E_R), transferring data (E_T), listening for a beacon E_L and remaining in standby mode (E_S). The only activities taking place in standby mode are background tasks: low-speed timers, sensor monitoring, and node synchronization. The available energy for the next period T' must satisfy the following inequality:

$$E_{avail}^{T'} \geq (E_R^{T'} + E_T^{T'}) + (E_L^{T'} + E_S^{T'}), \tag{4}$$

where $E_R^{T'}$, $E_T^{T'}$, $E_L^{T'}$ and $E_S^{T'}$ are the values of E_R, E_T, E_L and E_S for the next period T'.

The value of $E_R^{T'} + E_T^{T'}$ for node i equals $E_i^{T'}(R)$ which is the amount of energy consumed by node i when KR is R during period T' as explained in [8]. In other words, if each node remembers the amount of data traffic and the energy used in gathering information during the preceding period T, then $E_R^{T'} + E_T^{T'}$, which is the amount of energy required by node i for data transmission and reception when the KR is R during period T', can be estimated as follows:

$$\begin{aligned} E_R^{T'} + E_T^{T'} &= E_i^{T'}(R) \\ &= \frac{(\frac{L_N T}{T_N} + L_D P_i^T)(R)^\alpha + \frac{L_U T}{T_N} \cdot \sum_{j \in U_i^T} (d_{ij})^\alpha}{(\frac{L_N T}{T_N} + L_D P_i^T)(r)^\alpha + \frac{L_U T}{T_N} \cdot \sum_{j \in U_i^T} (d_{ij})^\alpha} \cdot E_i^T(r). \end{aligned} \tag{5}$$

This means that $E_R^{T'} + E_T^{T'}$ can be expressed by the function $f(KR)$, which is dependent on the knowledge range.

[2] In this chapter, the units of energy are always Joules [J].

We expect the peak transmitter power in Fig. 3 to be around 2 to 3mW [23], depending on its efficiency, and the receiver should not consume more than 1mW. Including the dissipation of the sensors and peripheral circuitry, a maximum peak power of 5mW is quite reasonable. Moreover, using advanced wake-up radio techniques or semi-asynchronous beaconing techniques, the average standby power of the node can be limited to 50 μW, although these values are quite system-dependent. The energy used in listening mode during T', which is denoted by $E_L^{T'}$, can be calculated as follows:

$$\begin{aligned} E_L^{T'} &= Power_{peak} \cdot T' \cdot DC[mWH] \\ &= Power_{peak} \cdot T' \cdot DC \cdot 3.6[Joule], \end{aligned} \quad (6)$$

where DC is the length of the wake-up period determined by the duty cycle of the node.

Similarly, the energy consumption in standby mode during T' can be obtained as follows:

$$\begin{aligned} E_S^{T'} &= Power_{standby} \cdot T' \cdot (1-DC)[mWH] \\ &= Power_{standby} \cdot T' \cdot (1-DC) \cdot 3.6[Joule]. \end{aligned} \quad (7)$$

We can now see that $E_L^{T'} + E_S^{T'}$ can be expressed as a function of the duty cycle $g(DC)$, and the energy available for the next period T' must satisfy the following inequality:

$$E_{avail}^{T'} \geq f(KR) + g(DC). \quad (8)$$

We want to find the pair values (KR, DC) for which $f(KR)+g(DC)$ is closest to $E_{avail}^{T'}$, so as to maximize the utilization of the next period's energy budget. Note that $E_{avail}^{T'}$ is constant, and $f(KR)$ and $g(DC)$ are increasing functions. Thus, by changing the KR[3], we can obtain the DC for which $f(KR)+g(DC)$ is closest to $E_{avail}^{T'}$. Therefore, there are many (KR, DC) pairs, which can maximize the utilization of the next period's energy budget.

B. Determining the KR and DC to Reduce Latency

In self-sustaining energy-harvesting sensor networks, each node wakes up at a different time due to the different duty cycle d. Therefore, the sender must know the duty cycle of the receiver in advance, and can then wait for the receiver to be ready to receive the packet.

As shown in Fig. 4, for a node x to send b bits of data to node y at time t involves a waiting time $T_{wait}(y,t)$ and a transmission time $T_{send}(b)$, assuming that node x already knows that the duty cycle of node y has a period T_y. The transmission time is composed of two parts: T_{beacon} for two-phase handshaking using a beacon, and T_{data} for link-layer transmission. Assuming that R is the data bandwidth of the channel between the nodes, the expected latency between node x and node y at time t can be calculated as follows:

$$\begin{aligned} L(x,y,t) &= T_{wait}(y,t) + T_{send}(b) \\ &= \{T_y - (t - t_{setup})\%T_y\} + \left(T_{beacon} + \frac{b}{R}\right). \end{aligned} \quad (9)$$

[3]The resolution of the KR is determined by the granularity of power control at each node.

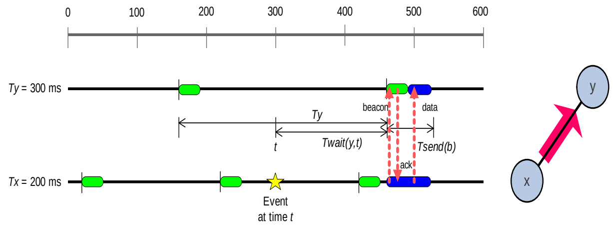

Figure 4: Latency of two nodes which have different duty cycles

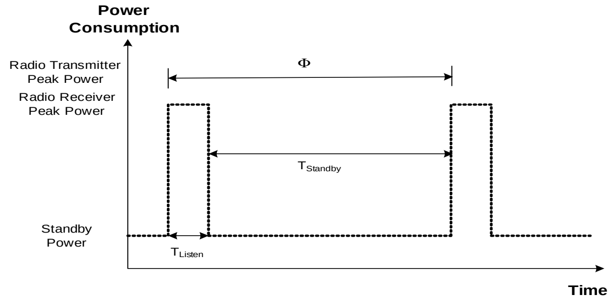

Figure 5: Duty cycle of a node

Among the candidate (KR, DC) pairs, a node chooses the pair which allows it to route data to the sink with the shortest end-to-end delay. This can be done by evaluating a heuristic function which approximates the expected end-to-end delay to the sink for a given (KR, DC) pair. This function has two elements; one is the expected time required to send the data to a one-hop neighbor, and the other is the expected hop count to the sink. Using Equation (9), it can be formulated as follows:

$$\begin{aligned} Eval(KR,DC) &= \frac{1}{\Phi}\int_{\Phi} T_{wait}(x,t)dt \cdot \{\frac{d}{KR}\} \\ &= \frac{1}{\Phi}\int_{\Phi}\{\Phi - (t - t_{setup}^{x})\%\Phi\}\, dt\,\{\frac{d}{KR}\}, \end{aligned} \quad (10)$$

where $T_{wait}(x,t)$ denotes the waiting time for node x to send data at time t; Φ is the period for which node x wakes up, as shown in Fig. 5; t_{setup}^{x} is the time required to set up the duty cycle of node x; and d is the distance from node x to the sink. Note that Φ has the value of T_{Listen}/DC.

D-APOLLO evaluates the function for each candidate pair (KR, DC), and finds the pair for which its value is the smallest. The node on which D-APOLLO is running uses this pair as the KR and DC for the next period, and in doing so it can expect to maximize its utilization of the next period's energy budget and to minimize the end-to-end delay of data sent to the sink.

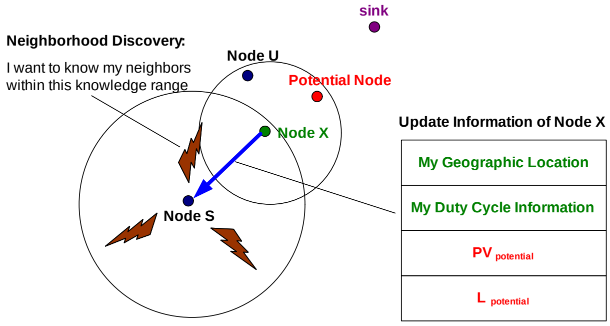

Figure 6: Information received by a node from its neighbor nodes

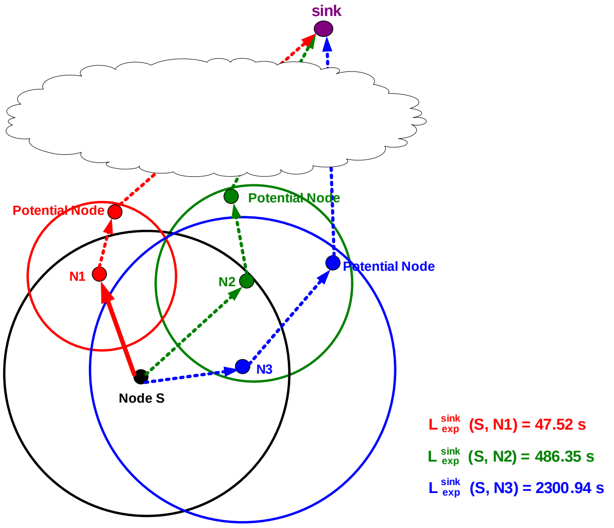

Figure 7: An example of a route selected by our low-latency geographic routing scheme

IV. Duty-Cycle-Aware Geographic Routing for Low Latency

Equation (9) allows us to estimate the expected delay for node x to send data to the node u as follows:

$$L^{hop}_{avg}(x,u) = \frac{1}{T_u}\int_{T_u} L(x,u,t)dt. \tag{11}$$

This is the expected latency per hop when data is sent to node u.

If $d(x,sink)$ is the distance between node x and the sink, then the expected number of hops from node x to the sink when the data is routed through node u is given by:

$$H^{sink}_{avg}(x,u) = \frac{d(x,sink)}{PV(u)}. \tag{12}$$

We recall that *PV(u)* is the distance from the source node x to the target node u in the direction of the vector from x to the sink. The details are clarified in [8].

Therefore, we can calculate the expected latency from node x to the sink via node u as follows:

$$L^{sink}_{avg}(x,u) = L^{hop}_{avg}(x,u)\cdot H^{sink}_{avg}(x,u). \tag{13}$$

Each node evaluates this equation for all progressive nodes[4], whenever its information about them is updated. So a node x evaluates $L^{sink}_{avg}(x,u)$ for every progressive node u, and chooses the node with the minimum value of $L^{sink}_{avg}(x,u)$ as its potential node.

$$PotentialNode(x) = \{u \mid \min_{u\in N(x)}\{L^{sink}_{avg}(x,u)\}\}, \tag{14}$$

where $N(x)$ is the set of neighbors of node x.

Every node has its own potential node for which it maintains values of PV and L^{hop}_{avg}, which we will call PV_{pot} and L_{pot} respectively. As shown in Fig. 6, when a node sends information about itself to a requesting node, it sends PV_{pot} and L_{pot} as well as its location and duty cycle.

When a node receives data or senses a new event, it uses the information received from its neighbors to calculate $L^{sink}_{exp}(S,N_i)$ for each progressive node N_i. This is the expected latency from node S to the sink when the data is routed through N_i, and is calculated as follows:

$$\begin{aligned} L^{sink}_{exp}(S,N_i) &= L^{Hop}_{exp}\cdot H^{sink}_{exp} \\ &= \frac{L(S,N_i,t)+L_{pot}(N_i)}{2}\cdot\frac{d(S,sink)}{\{PV(N_i)+PV_{pot}(N_i)\}/2}. \end{aligned} \tag{15}$$

$L(S,N_i,t)$ and $PV(N_i)$ can easily be calculated using information about the duty cycle and location of node N_i. And $L_{pot}(N_i)$ and $PV_{pot}(N_i)$ are in the message received from node N_i, as shown in Fig. 6. Therefore, the computational overhead is very small.

Among the progressive nodes, node S chooses the node N_i which has the minimum value of $L^{sink}_{exp}(S,N_i)$ as the next node.

$$NextNode(S) = \{N_i \mid \min_{N_i\in N(S)}\{L^{sink}_{exp}(S,N_i)\}\}, \tag{16}$$

where $N(S)$ is the set of neighbors of node S.

The solid arrow from node S in Fig. 7 represents the route determined by our low-latency geographic routing scheme. This is the shortest path that corresponds to the minimum end-to-end delay. It is determined by comparing the value of $L^{sink}_{exp}(S,N_i)$ for each progressive node.

As shown in Fig. 7, node S calculates $L^{sink}_{exp}(S,N_i)$ for each of its progressive nodes (nodes $N1$, $N2$ and $N3$) using all the information it has. In this case, $N1$ has the smallest value of this function. This means that the path to the sink which includes node $N1$ can be expected to have the shortest end-to-end delay out of all the paths which include a progressive node of node S. So node S chooses node $N1$ as the next node.

This algorithm can avoid the geographically restricted flooding called the 'void' problem, which can occur with a greedy geographic routing algorithm when a recipient node has no neighbor in the direction of the sink node. Our algorithm avoids this situation by setting a flag in the PV_{pot} field, which tells other nodes not to route data to it.

[4]We call a node in the positive direction towards the sink a progressive node.

TABLE I.: Main parameters used in our solar-cell energy model

Parameter	Value
Output voltage of solar cell	3.3V
AA-sized NiMH battery capacity	1800mAh
Amount of energy from solar cell over 1 day	200∼300mWh
$Power_{peak}$ in Equation (6)	
$Power_{standby}$ in Equation (7)	3mW

TABLE II.: Environmental parameters used in the simulation of D-APOLLO

Environmental Parameter	Value
E_{allow} in Equation (3)	300mWh
α in Equation (5)	2
L_N and L_U in Equation (5)	64bit
L_D in Equation (5)	1Kbit
T_N in Equation (5)	60s
T (Period of D-APOLLO) in Equation (5)	1 day
Traffic	1Kbit/sec/node

In addition, each node sets this flag when its own traffic load exceeds a known capacity. This prevents data being routed to a congested node.

V. Performance Evaluation

We designed a simulation to evaluate the performance of our low-latency geographic routing scheme and D-APOLLO. C++ was used to develop the simulator which is based on NS2.

A. Simulation Environment

We derived an energy model of a solar-powered sensor from the specification of the Heliomote [6] solar-powered sensor platform. Table I shows the important parameters of the energy model used in our simulation.

Table II summarizes the environmental parameters used in simulating D-APOLLO, which runs on each node and uses these parameters to determine the pair of KR and DC values for the next period. The traffic of each node is 1Kbit/sec[5] and the period used by D-APOLLO is 1 day.

We simulated a WSN containing between 50 and 400 solar-powered nodes, spread randomly over an area of $100\times100m^2$. The simulated period of operation is 7 days. The resolution of the KRs is 1m and the maximum radio range is 10m. The maximum data-rate of a node is 100Kbit/s. Table III summarizes the environmental parameters used in simulating the proposed routing scheme.

[5]This number is based on real radio prototypes and a realistic smart home scenario [23].

TABLE III.: Environmental parameters used in the simulation of the proposed routing scheme

Environmental Parameter	Value
Maximum radio range	20m
Data bandwidth	100Kbit/s
T_{beacon} for two-phase handshaking in Equation (9)	0.5s
Size of a data packet in Equation (9)	1Kbit
t_{setup} in Equation (9)	0s
Number of nodes	50∼400
Terrain	100m×100m
Node placement	Random
Φ in the Fig. 5	D-APOLLO
T_{Listen} in the Fig. 5	1s
Knowledge range of each node	D-APOLLO
Granularity of knowledge range	1m

We have compared our scheme to: (i) a random selection scheme in which a node routes data to one of its progressive nodes, chosen at random; (ii) greedy PV selection, in which a node routes data to its most progressive node; (iii) greedy DC selection, in which a node routes data to the progressive node which has the shortest waiting time; and lastly (iv) a greedy PV + DC scheme, in which a node sends data to the node which is expected to have the shortest end-to-end delay to the sink, based only on information from neighbor nodes, and not considering information about the potential node of each neighbor.

B. Simulation Results

Table IV shows the failure rate for each algorithm in routing data to the sink. The channel error-rate of each node was set to 0.01. As we can see in Table IV, our proposed scheme outperforms the others in terms of successful delivery of data. We suggest that this is largely due to the way in which our scheme circumvents the 'void' situation. It can escape this situation in advance by setting a specific flag in the information message, which tells other nodes not to route data to the source of the message.

The network with a lower density of nodes shows a higher failure-rate with each scheme. We suggest that this is because of increased probability of a void situation and isolation due to the sparsity of the nodes. We see that the failure-rate of our scheme improved more quickly with increasing density than the other schemes.

Fig. 8 compares the average end-to-end delays for different geographic routing algorithms. This figure shows that our scheme achieves the shortest end-to-end delay for every node density. Its latency is 56% less than that of the random selection scheme when there are 50 nodes. As the number of nodes increases, that differential grows. Moreover, our scheme shows the least variation in latency with changes in node density.

TABLE IV.: Failure-rate in routing to the sink (Percentage)

Routing Scheme	Number of Nodes					
	50	100	150	200	300	400
Proposed Scheme	42.4	21.1	3.4	1.5	0.6	0.02
Random Selection	56.1	53.0	27.8	13.7	3.3	0.98
Greedy PV	54.9	49.3	23.6	14.2	2.6	0.9
Greedy DC	50.8	49.9	25.2	13.0	2.8	0.7
Greedy PV+DC	54.9	50.2	22.8	13.5	1.7	0.9

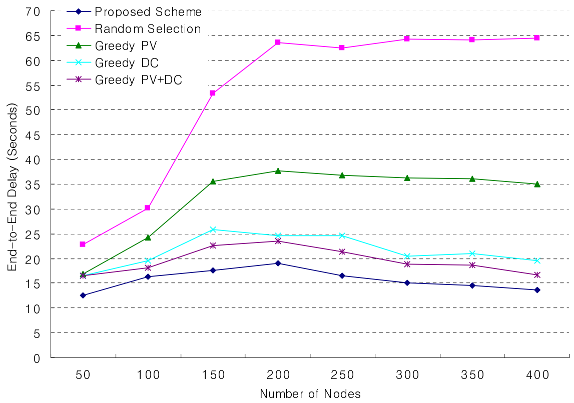

Figure 8: Comparative end-to-end delays for different geographic routing schemes

For every scheme, the average delay rises as the number of nodes increases from 50 to 200. This is because the curves in Fig. 8 only relate to trials in which a packet was successfully delivered to the sink. Moreover, the KR and DC are reduces according as the density of nodes rises, because more energy is required to maintain the topology. Therefore, there is a tendency for latency to rise until the failure-rate approaches saturation. However, when these are more than 200 nodes, the latency of each scheme saturates, or even decrease somewhat. We suggest that this is because the failure-rate is relatively static when the network has more than 200 nodes, and because there are usually more ways to route data when the number of nodes is larger, even though the KR and DC tend to fall off somewhat.

The greedy PV+DC scheme has a relatively short end-to-end delay. But this algorithm has a much lower success rate than our scheme, as shown in Table IV. Our scheme persistently tries to deliver data to the sink, avoiding a 'void' situation, even though the data has to be diverted, while greedy PV+DC gives up.

At the end of the simulation, the average KR of all nodes is 13.5m and the average duty cycle is 0.067 when the number of nodes is 200.

VI. CONCLUSION

The demand for energy-harvesting WSNs is growing because they can overcome the short running time of battery-based WSNs. Additionally, interest of WSN applications is increasingly centered on time-critical data which requires low-latency to the sink. Therefore we have developed a low-latency routing scheme to support energy-harvesting WSNs.

If a sensor is operated by the battery, its knowledge range (KR) and duty cycle (DC) are fixed values corresponding to the characteristic of a sensor or an application. On the other hands, when using AC-power, the sensor can always operate in active mode with the maximum radio range. The harvested energy, however, can be gathered permanently, but its quantity during a certain period is limited. Therefore, in order to utilize stored energy effectively, we developed D-APOLLO algorithm which periodically determines KR and DC values using information about the local energy budget for the next period, based on an energy model of a harvesting device such as a solar cell. The purpose of this algorithm is to maximize the utilization of hoarded energy for low latency: it is not appropriate for a node in an energy-harvesting WSN to focus solely on minimizing its energy consumption.

Under this regime, each node executes duty-cycle-based geographic routing protocol to produce a low-latency path. This protocol takes into account geographic information and DC values about neighbor nodes within the KR in order to reduce latency. It can therefore find a faster path than a protocol which only considers one of them. The geographic element increases scalability, reduces memory requirements, and supports the mobility of the nodes.

ACKNOWLEDGMENT

This work was supported by BK21 Research Fellowship from the Ministry of Education and Human Resources Development in Korea, and also by Institute of Computer Technology in Seoul National University.

REFERENCES

[1] R. Min, M. Bhardwaj, S. Cho, E. Shih, A. Sinba, A. Wang, and A. Chandrakasan, "Low-power wireless sensor networks," in *Proceedings of the Fourteenth International Conference on VLSI Design*, January 2001.

[2] R. C. Shah and J. Rabaey, "Rabaey: Energy aware routing for low energy ad hoc sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, March 2002.

[3] S. Mahlknecht and M. Roetzer, "Energy supply considerations for self-sustaining wireless sensor networks," in *Proceedings of the Second European Workshop on Wireless Sensor Networks*, February 2005.

[4] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, "Design considerations for solar energy harvesting wireless embedded systems," in *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks*, April 2005.

[5] J. P. M. Torregoza, I. Kong, W. Hwang, and J. Kim, "Battery model for wireless networks using photovoltaic cells," in *Proceedings of the Eighth International Conference on Advanced Communication Technology*, February 2006.

[6] A. Kansal, D. Potter, and M. B. Srivastava, “Performance-aware tasking for environmentally powered sensor networks,” in *Proceedings of the ACM Conference of Special Interest Group for the Computer/Communication System Performance Community*, June 2004.
[7] M. Minami, T. Morito, H. Morikawa, and T. Aoyama, “Solar Biscuit: a battery-less wireless sensor network system for environmental monitoring applications,” in *Proceedings of the Second International Workshop on Networked Sensing Systems*, June 2005.
[8] D. Noh, J. Kim, J. Lee, D. Lee, H. Kwon, and H. Shin, “Priority-based routing for solar-powered wireless sensor networks,” in *Proceedings of the IEEE International Symposium on Wireless Pervasive Computing*, February 2007.
[9] D. Noh, D. Lee, and H. Shin, “Qos-aware geographic rouiting for solar-powered wireless sensor networks,” 2007, under review in IEICE Transaction on Communication.
[10] ——, “Mission-oriented selective routing for wireless sensor networks,” August 2007, under review in IEEE International Conference on Communications and Networking in China.
[11] H. Kwon, D. Noh, J. Kim, J. Lee, D. Lee, and H. Shin, “Low-latency routing for energy-harvesting wireless sensor networks,” July 2007, to appear in the 4th International Conference on Ubiquitous Intelligence and Computing.
[12] R. Jain, A. Puri, and R. Sengupta, “Geographical routing using partial information for wireless ad hoc networks,” *IEEE Personal Communications*, vol. 8, no. 7, pp. 48–57, February 2001.
[13] I. Stojmenovic, “Position-based routing in ad hoc networks,” *IEEE Communications Magazine*, vol. 40, no. 7, pp. 128–134, July 2002.
[14] T. Melodia, D. Pompili, and I. Akyildiz, “Optimal local topology knowledge for energy efficient geographical routing in sensor networks,” in *Proceedings of the 23th IEEE Conference on Computer Communications*, March 2004.
[15] A. Willig, R. Shah, J. Rabaey, and A. Wolisz, “Altruists in the picoradio sensor network,” in *Proceedings of the forth IEEE International Workshop on Factory Communication Systems*, August 2002.
[16] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, and J. Schiller, “Solar-aware clustering in wireless sensor networks,” in *Proceedings of the Ninth IEEE Symposium on Computers and Communications*, June 2004.
[17] T. Voigt, H. Ritter, and J. Schiller, “Utilizing solar power in wireless sensor networks,” in *Proceedings of the IEEE Conference on Local Computer Networks*, October 2003.
[18] W. Heinzelman, “Application-specific protocol architectures for wireless sensor network,” vol. Ph.D. Thesis, Massachusetts Institute of Technology, USA, June 2000.
[19] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: a scalable and robust communication paradigm for sensor networks,” in *Proceedings of the Sixth ACM International Conference on Mobile Computing and Networking*, August 2000.
[20] A. Kansal and M. B. Srivastava, “An environmental energy harvesting framework for sensor networks,” in *Proceedings of the International Symposium on Low Power Electronics and Design*, August 2003, pp. 481–486.
[21] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, “Power management in energy harvesting sensor networks,” 2007, to appear in ACM Transactions on Embedded Computing Systems.
[22] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless micro sensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, October 2002.
[23] I. Stojmenovic, *Handbook of Sensor Networks*. Wiley-Interscience, 2005.

Donggeon Noh received B.S., M.S., and Ph.D. degrees in Electronic Engineering and Computer Science from Seoul National University, Seoul, Korea in 2000, 2002, and 2007, respectively. He is currently a visiting scholar in Department of Computer Science at the University of Illinois at Urbana-Champaign. His current research interests are in the field of mobile computing and communication with emphasis on routing protocol, QoS control and service discovery architecture on wireless ad-hoc and sensor network.

Ikjune Yoon received the B.S. degree in Computer Science from Chonbuk University, Jeonju, Korea in 2006. He is currently a Ph.D. student in School of Electronic Engineering and Computer Science at Seoul National University. His research interests include sensor network and multimedia system.

Heonshik Shin received the B.S. degree in applied physics from Seoul National University, Korea, in 1973 and Ph.D. degree in computer engineering from the University of Texas at Austin in 1985. He is currently a professor in School of Computer Science and Engineering at Seoul National University. His research interests include mobile system, embedded system and multimedia system.