Controlled Potential-Based Routing for Large-Scale Wireless Sensor Networks

Daichi Kominami
Graduate School of Information Science and Technology
Osaka University
1-5 Yamadaoka, Suita
Osaka 565-0871, Japan
d-kominami@ist.osaka-u.ac.jp

Masashi Sugano
School of Comprehensive Rehabilitation
Osaka Prefecture University
3-7-30 Habikino, Habikino
Osaka 583-8555, Japan
sugano@rehab.osakafu-u.ac.jp

Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
1-5 Yamadaoka, Suita
Osaka 565-0871, Japan
murata@ist.osaka-u.ac.jp

Takaaki Hatauchi
Fuji Electric Co., Ltd.
1 Fuji, Hino
Tokyo 191-8502, Japan
hatauchi-takaaki@fujielectric.co.jp

ABSTRACT

Improving the scalability of wireless sensor networks is an important task, and toward this end, much research on self-organization has been conducted. However, the problem remains that much larger networks based on pure self-organization cannot be guaranteed to behave as desired. In this paper, we propose a controlled potential-based routing protocol. This protocol is based on a novel concept: a "controlled self-organization scheme", which is a self-organization scheme accompanied by control from outside the system. This scheme ensures desired network behavior by controlling a portion of nodes operated in self-organization. Through simulation experiments with a multi-sink network, we show that traffic loads can be equalized among heterogeneously distributed sink nodes, and moreover, that load balancing among the relay nodes can bring about a 138% extension of network lifetime.

Categories and Subject Descriptors

C.2.2 [**COMPUTER-COMMUNICATION NETWORKS**]: Network Protocols—*Routing protocols*

General Terms

Algorithm, Design, Performance, Reliability

Keywords

Controlled Self-Organization, Potential-Based Routing, Diffusion Equation, Load Balancing

MSWiM'11, October 31–November 4, 2011, Miami, Florida, USA.

1. INTRODUCTION

Recent advances in wireless and micro-electromechanical system (MEMS) technologies have made it possible to develop extremely small sensor nodes that have wireless communication facilities, and as a result, considerable attention has been directed at wireless sensor networks (WSNs). Wireless sensor networks consist of a large number of sensor nodes and sink nodes, and these networks can be used for a wide range of applications free of any need for a fixed network infrastructure, such as a monitoring of phenomena over a wide area.

Many challenges must be met to realize a WSN, and one of these is to improve scalability of routing. When dealing with thousands of sensor nodes, the wireless channel is occupied by the exchange of routing information, which consumes more energy and more bandwidth. Within such networks, it is impractical to give numerous nodes unique identifiers (IDs) and exchange all routing information among them. This stands in contrast to IP networks, where an ability to access an arbitrary node is required. Another scalability problem is that heavy loads are placed on nodes neighboring the sink node, as those nodes, as a consequence of many-to-one (or many-to-some) communication, must forward data from other sensor nodes in addition to their own data.

In order to reduce the amount of routing information that needs to be exchanged throughout the network, the emphasis is on self-organization, that is, entirely local decision-making by each node based only on local information. Self-organization has good scalability, adaptability, and robustness [4], which are important properties for sensor networks. These properties are brought about by numerous interactions among the local-level components of a system without external or centralized control process. Conversely, pure self-organization has the following problems because of its bottom-up design [12] in which only local information is used.

- Guaranteeing an optimal operation is difficult.
- Confirming an operation on the entire network is difficult.
- Convergence speed after an environmental change is slow.

Since each entity in self-organization makes a decision on the ba-

sis of local information, local optimization is easy to achieve, while a global optimal operation of the network is hard to guarantee. This local decision-making based on the local interaction in large systems derives emergence, however as is generally known, controlling emergence is not realistic. Then, once a self-organized system converges to act undesirably, it remains unchanged. In [12], the author suggests use of an observer/controller architecture, where an observer and a controller are responsible for correcting the system behavior. In the case of optimal operation in self-organized routings, since each node selects its next hop based on only local information, it is not possible to deal with network problems from a macro viewpoint; one would not be able to alleviate excess concentrations of communication load induced by an irregular node distribution density.

To resolve these problems, we propose a controlled self-organized routing, which is based on a novel scheme called "controlled self-organization" [2]. This scheme rests on a controller which is responsible for ensuring the desired behavior of the system, for guaranteeing a high degree of system perform, and for encouraging a convergence of the state of the system. That is, the controller makes the self-organized system *manageable* by controlling some of the components within the self-organization scheme to accomplish the above objectives.

In this paper we adopt a potential-based routing [1, 5, 6, 8–11, 13, 15, 16] as a self-organized routing and apply controlled self-organization scheme to it. In other words, we propose a *controlled potential-based routing (CPBR)*. In potential-based routing schemes, nodes have a scalar value called "potential", and a next hop is based on only a node's potential and its neighbors' potentials. Here, a node calculates its own potential from the potentials, the number of hops from the sink node, or the remaining energy of itself or its neighbors. Basically, the smaller a hop count to the sink node is, the lower is the potential value assigned to it. Therefore, if a node simply transmits data toward one of its neighbor nodes having a smaller potential than its own, the data will eventually reach the sink node.

On the reduction of the load on nodes neighboring the sink node, multiple sink nodes are deployed over the network, with data obtained by the sink nodes transmitted to a data server (users or applications can then access the data from the server as necessary). Here, each sensor node does not necessarily select only one sink node as its destination. When potential-based routing is applied to multi-sink networks, each node only has to forward data according to potentials: the data will eventually reach a sink node. Thus, potential-based routing can be straightforward to use on multi-sink sensor networks.

Because the potential-based routing is based on self-organization, it has above-described some problems, such as concentrations of communication load provoked by an inhomogeneous distribution density of sensor or sink nodes. In CPBR, sink nodes act as controllers, according to network manager's requests, that adjust their own potentials to construct a desired potential field. We assume that multiple sink nodes are connected to one another on a high-bandwidth wired network so that they can exchange various information periodically and instantaneously; for example, the number of received data packets and the remaining energy of their neighbor nodes. Sink nodes control their own potential on the basis of this information so that even as sensor nodes decide their own potential from local information, a preferable potential field is constructed over the entire network.

In order to realize CPBR, we begin by introducing a potential-field construction from the diffusion equation in Section 2. In the section, we first show how sensor nodes construct a potential field and select a next hop according to it. We then extend this method of construction to include local optimization. In Section 3, we describe controlled potential-based routing intended for overall optimization; then in the following section, we present the results of a simulation. Finally, we conclude our paper in Section 5.

2. POTENTIAL-BASED ROUTING

2.1 Related Work

Various works related to potential-based routing have been conducted [1, 5, 6, 8–11, 13, 15, 16]. Such efforts are classified into following two types.

1. Physics-knowledge based schemes
2. Hop-count based schemes

As for physics-knowledge based schemes, the focus has been on analogies between electrostatic fields and sensor networks [5, 6] and on analogies between electrical circuits and sensor networks [10]. In these studies, potentials are assigned at sensor nodes by solving a Poisson equation. With the solution obtained from the equation, a load-balancing potential field is constructed. However, to solve the equation, certain location information is needed for the nodes. However, obtaining and exploiting location information assume the availability of GPS receivers or some other means, and thus the cost of producing such nodes becomes much higher. By extension, the cost for deploying sensor nodes also rises, which is a potentially critical problem for constructing a large scale network. In a sense, this is also a scalability problem. A related difficulty is that it might not be feasible to provide GPS receivers indoors, in rooms below ground level, within heavily forested areas, or at other such locations with limited or obstructed satellite coverage.

As for hop-count based schemes, nodes calculate their own potential essentially from their hop count from the sink nodes [8, 11, 15, 16]. In references [8, 11, 15], nodes also use their own remaining energy and that of their neighbor nodes for load balancing. An effective data aggregation mechanism supported by potential-based routing was proposed in [16], where local queue-length information is used to calculate potentials. Although the proposed routing schemes exhibit good performance, location information is required in the schemes of references [9, 11]. Also, the parameters used to calculate potentials were insufficiently examined and evaluated in references [8, 15], and thus the difference between their proposals and mere minimum hop routing along with remaining energy information is not clear. Moreover, the most important problem is that all the above-mentioned studies offer no mechanism for guaranteeing intended network operation. To resolve this critical problem, we propose controlled potential-based routing (CPBR). Our CPBR utilizes a physics-knowledge based scheme inspired by thermal diffusion phenomenon as well as [13], however, our CPBR does not require location information. We focus on an analogy between the heat conduction from a heat source and the potential conveyance from a sink node. In CPBR, sensor nodes have to change their own potential according to the potential of the sink nodes. Making use of the diffusion equation that describes heat conduction, CPBR allows diffusion of the potentials of sink nodes throughout the entire sensor network.

2.2 Potential Field Construction from the Diffusion Equation

The diffusion equation is shown by the partial differential equation (1), which provides magnitude ϕ of the diffusing quantity at

time t at position $\mathbf{x}$.

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = D \triangle \phi(\mathbf{x},t), \quad (1)$$

where D is the diffusion rate and takes a positive value. By discretizing this equation and regarding ϕ as a potential, it becomes possible to construct a potential field based on self-organization where the behavior is governed by only local information.

2.2.1 Discrete Diffusion Equation

Node n calculates its own potential at time step $t+1$ (denoted by $\phi(n,t+1)$) based on the discrete diffusion equation (2). In equation (2), $N_a(n)$ denotes a set of nodes neighboring node n. It can be noted from the equation, that location $\mathbf{x}$ is cleared and the potential of node n is obtained from the latest potentials of $nb(n)$ and its own last potential. In order to calculate potentials, nodes must periodically inform their neighbor nodes of their own potentials.

$$\phi(n,t+1) = \phi(n,t) + D(n) \sum_{k \in N_a(n)} \{\phi(k,t) - \phi(n,t)\}. \quad (2)$$

In discrete equation (2) (derived from continuous equation (1)), it can be thought that $D(n)$ is a parameter that changes the magnitude of influences by potentials of the neighbor nodes. An important point should be noted: potentials may oscillate when $D(n)$ is large. For the solution of the problem, we consider the case where node n has an only one neighbor node m. Equation (2) can thus be replaced by $\phi(n,t+1) = D(n)\phi(m,t) + (1-D(n))\phi(n,t)$, which represents an internal/external division of the points on the number line. In the following, we consider the case of $\phi(n,t) < \phi(m,t)$.

In the case $0 < D(n) < 1$**:**

After node n receives the potential of node m, the following inequality is satisfied: $\phi(n,t) < \phi(n,t+1) < \phi(m,t)$. Repeating this procedure, potentials of node n and node m approach and converge between $\phi(n,t)$ and $\phi(m,t)$. In this case, node n's potential remains smaller than node m's potential.

In the case $1 \leq D(n) < 2$**:**

After node n receives the potential of node m, the following inequality is satisfied: $\phi(m,t) < \phi(n,t+1) < 2\phi(m,t) - \phi(n,t)$. Repeating this procedure, the potentials of node n and node $m$approach and converge, but the relationship between the magnitude of node n's potential and node m's potential is indefinite.

In the case $2 \leq D(n)$**:**

After node n receives the potential of node m, the following inequality is satisfied: $2\phi(m,t) - \phi(n,t) \leq \phi(n,t+1)$. Repeating this procedure, potentials of node n and node m remain unchanged or diverge. Moreover, the magnitude relationship between node n's potential and node m's potential is indefinite.

For the diffusion of potentials, it is preferable that $D(n)$ satisfies the following expression: $0 < D(n) < 1$. In the general case (i.e., when there exist multiple neighbor nodes), we set $D(n)$ to $\frac{\alpha}{|N_a(n)|}$, where $|N_a(n)|$ is the number of neighbor nodes of node n. As a result, it can be considered that each node has been influenced by the potential of essentially only one node. We then set α to the value between 0 and 1 in order to keep the potential from oscillating.

2.2.2 Boundary Condition

As an initial condition, the potentials of sensor nodes are set to zero. To construct a potential field from equation (2), we utilize a Dirichlet boundary condition to specify the sink nodes' potentials:

$$\forall d \in N_s,\ \phi(d,t) = \Phi(d), \quad (3)$$

where N_s is a set of sink nodes and $\phi(d,t)$ is the potential of sink node d at time step t. $\Phi(d)$ (≤ 0) is the potential control function, which we discuss later in Section 3. By the nature of the diffusion equation, this boundary condition is insufficient because the potentials of all nodes will arrive at much the same value as the potential of the sink node eventually. We thus define another boundary condition that has to be satisfied by nodes at the edge of the network:

$$\forall n \in N_{edge},\ \phi(n,t) = 0, \quad (4)$$

N_{edge} is a set of nodes at the edge of the network, and node n is an element of the set that satisfies any of the following conditions (5) or (6):

$$\forall k \in N_a(n),\ H(n) > H(k), \quad (5)$$

$$\forall k \in N_a(n),\ \{H(n) \geq H(k)\} \cap \{D_{id}(n) = D_{id}(k)\}. \quad (6)$$

Here, $H(n)$ is the minimum hop count of node n from a nearby sink nodes and $D_{id}(n)$ is the ID of the sink node. Nodes append H and D_{id} into their ID, which is transmitted periodically to let their neighbor know their H and D_{id}. Sink nodes set their H to zero and D_{id} to their own ID. When node n receives an ID from node m, node n updates $H(n)$ and D_{id}. If $H(m)+1$ is larger than $H(n)$, node n sets $H(n)$ to $H(m)+1$ and also sets $D_{id}(n)$ to $D_{id}(m)$. When $H(m)+1$ equals $H(n)$, node n changes $D_{id}(n)$ to $D_{id}(m)$ with a probability of 0.5.

The expression (5) cannot define the potentials of nodes at the network edge when two or more nodes with the same hop count exist at the network edge. Instead, we use expression (6) for that case. By using D_{id}, nodes in the middle portion of two sink nodes are prevented from mistakenly deciding that they are at the edge of the network. Because D_{id} represents the closest sink node, D_{id} does not coincide among neighboring nodes in the middle portion of two sink nodes.

2.2.3 Local Optimization

In this section, we present the construction of a potential field where nodes can select the best next hop locally. To do this, we add a term ρ on the right-hand side of the discrete diffusion equation (2).

$$\phi(n,t+1) = \phi(n,t) + D(n) \sum_{k \in N_a(n)} \{\phi(k,t) - \phi(n,t)\} + \rho(n,t), \quad (7)$$

where $\rho(n,t)$ is a variable indicating the influence of node n on the potential field at time step t (a larger ρ is associated with lower probability that node n is selected as a next hop and vice versa). Here, we show load balancing based on the residual energy with $\rho(n,t)$.

Node n increases $\rho(n,t)$ when the residual energy of node n is smaller than the average of that of the neighbor nodes whose hop count equals $H(n)$. Residual energy is contained in an ID and transmitted periodically.

The algorithm for deciding $\rho(n,t)$ is as follows, and is executed each time a potential is received.

1. Node n extracts the average residual energy of the neighbor nodes that have the same hop count as node n at time step t (denoted by $E_{avg}(n,t)$) and compares $E_{avg}(n,t)$ with own residual energy at time step t (denoted by $E_{rem}(n,t)$).
   - If $E_{rem}(n,t) > E_{avg}(n,t)$, $\rho(n,t)$ is set to zero.
   - If $E_{rem}(n,t) \leq E_{avg}(n,t)$, it follows step 3..
2. $E_{dif}(n,t)$ is the difference of energy between node n and its neighbors at time step t and assigned to $E_{avg}(n,t) - E_{rem}(n,t)$.

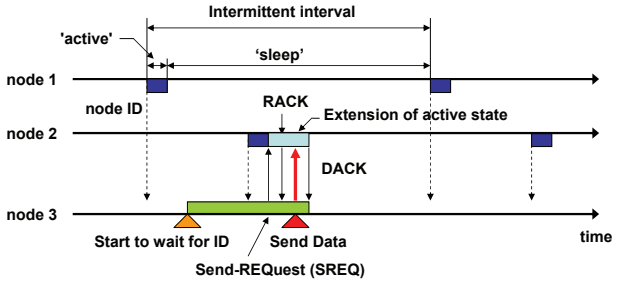


Figure 1: Intermittent receiver-driven data transmission (IRDT) protocol, which is one of the receiver-driven duty cycling MAC protocols such as RI-MAC [14]; we apply this protocol to the potential-based routing

- If $E_{dif}(n,t) < E_{dif}(n,t-1)$, $\rho(n,t)$ is unchanged.
- If $E_{dif}(n,t) \geq E_{dif}(n,t-1)$, $\delta(n,t)$ is added to $\rho(n,t)$, where $\delta(n,t)$ is the difference between $\phi(n,t)$ and average potential of the neighbor nodes whose hop count is same as node n at time step t.

3. Finally, $\rho(n,t)$ is set to $\frac{\rho(n,t)}{|N_a(n)|}$.

The procedure 3. is to suppress the dependence for the number of neighbor nodes on potential. This reduces dependency from our routing on the density of the network.

2.2.4 MAC Layer Protocol and Potential Dissemination

Another major challenge in wireless sensor network research is energy efficiency. For energy efficiency in wireless sensor networks, consideration of duty cycling MAC in which wireless nodes sleep and wake up periodically is required. Then, we use an *intermittent receiver-driven data transmission (IRDT)* protocol for MAC layer, which aims to save energy and get high reliability [7]. Note that our routing protocol is not limited to IRDT and it is applicable also in other underlying protocols. In IRDT, each receiver sends its own ID to inform other nodes that it is ready to receive a data packet (Figure 1). A sender node waits for a receiver ID, and when it acquires an ID from an appropriate receiver, it establishes a link with it by returning an SREQ message. After obtaining an acknowledgement for the SREQ (RACK), the sender transmits a data packet and terminates the communication upon receipt of an acknowledgement for the data (DACK). In this way, a sender node can communicate with one or more receivers flexibly, which can improve the communication reliability and save considerable energy. Because IRDT is an asynchronous MAC protocol that does not require synchronization, it is considered to have scalability.

By having the potential and node ID be periodically transmitted in IRDT, a node can inform its neighbor nodes of its potential. This simple modification produces little overhead. Note that because IRDT uses a duty cycling mechanism, where each node periodically cycles between an awake state and a sleep state, it is not necessarily that a transmitted potential be received by a node within the range of the communication. Each node wakes up and waits to receive potentials for a period of T_p at intervals of T_i. We refer to this period as the "sampling period", and to this interval as the "sampling interval". A sampling period should be longer than the interval of the periodic ID transmission in IRDT to ensure neighbor potentials are known by a node.

Potentials of neighbor nodes are managed in a soft-state manner. In other words, if a node receives a potential from a neighbor node during a sampling period, the node stores the potential; otherwise, the node deletes the information about the neighbor node. The procedure for calculating a potential is shown below.

During a sampling period T_p:

1. If a node receives a potential, it returns its own potential. After returning its potential, it calculates its own potential according to equation (2).
2. A node that receives, and that was intended to receive, the potential returned in step 1. also calculates its own potential.

While waiting for an ID for data transmission:

1. If a node receives an ID, it returns an SREQ containing its own potential. After returning the SREQ, it calculates its own potential.
2. A node that receives, and that was intended to receive, the potential returned in step 1., also calculates its own potential.

Immediately after a sampling period T_p:

1. Potentials of nodes whose potentials have not been updated for a period of T_i are deleted. After this process, the node calculates its own potential.

2.3 Routing in Potential Field

Here, consideration of both routing protocol and MAC protocol (in particularly, duty cycling MAC protocol) is important for energy efficiency. In IRDT, a node that has data to send waits for an ID from an appropriate node [7]. When the node receives an appropriate ID, it forwards the data to the sender of the ID. In our potential-based routing, a potential is transmitted along with an ID. A sender waits for a potential and when it receives a potential, it decides whether to forward data. Most of the proposed potential-based routing schemes use only one receiver that has the minimum potential; however, in duty-cycling MAC protocols, most the energy is consumed while the sender nodes wait for a receiver to awaken. Because energy efficiency is crucial in a sensor network, our potential-based routing uses multiple receivers. This can reduce energy consumption and improve reliability. When node n receives a potential from node r and it is not greater than its own potential, node n always returns an SREQ to node r.

3. CONTROLLED POTENTIAL-BASED ROUTING

In CPBR, multiple sink nodes share information and decide their own potentials for constructing a desired potential field. Sink nodes exchange information at regular time intervals T_m for the purpose of control. We call this information "metric value" (denoted by m). We aim here at balancing the load on sink nodes and sensor nodes.

- **Load balancing of sink nodes**
  For this purpose, sink nodes control their potential to maintain a uniform number of received data packets. The metric value here is the number of data packets received by sink node d (denoted by $N_{dt}(d)$).
- **Load balancing of sensor nodes**
  In the network, the nodes particularly neighboring the sink nodes frequently relay data and thus consume more energy. Thus, the metric value is needed to be based on the sum of the remaining energy of sink node d's neighbors (denoted by $P_{nb}(d)$). To make the energy consumption of the neighbor nodes of sink node d equal, sink nodes control their potential to equalize the decrease rate of $P_{nb}(d)$ (denoted by

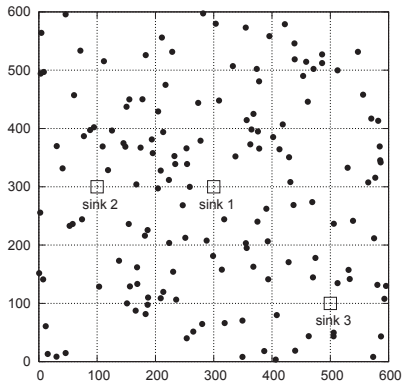


Figure 2: Network model (3 sink nodes)

$P_{rt}(d)$). As the metric value, the energy consumption rate, $\frac{P'_{nb}(d)-P_{nb}(d)}{P_{nb}(d)}$, is used, where $P'_{nb}(d)$ is the last $P_{nb}(d)$.

The potential of sink node d, $\phi(d)$, is given by the potential control function $\Phi(d)$. $\Phi(d)$ is decided with the following algorithm.

1. Sink nodes set their potential to the initial value Φ_{init}:

$$\Phi(d) = \Phi_{init} \ \ (\Phi_{init} < 0). \tag{8}$$

2. Sink nodes calculate $\bar{m}$, which is the average of the metric value. For example, $\bar{m}$ for $N_{dt}(d)$ (denoted by $\overline{m_N}$ for convenience) is defined by equation (9) and $\bar{m}$ for $P_{rt}(d)$ (denoted by $\overline{m_P}$) is defined by equation (10):

$$\overline{m_N} = \frac{\sum_{d\in N_s} N_{dt}(d)}{\sum_{d\in N_s} 1}, \tag{9}$$

$$\overline{m_P} = \frac{\sum_{d\in N_s} \{P'_{nb}(d) - P_{nb}(d)\}}{\sum_{d\in N_s} P_{nb}(d)}. \tag{10}$$

3. The potential of sink node d is given according to expression (11):

$$\Phi(d) = \begin{cases} \Phi(d) * (1 - \theta\frac{m(d)-\bar{m}}{\bar{m}}) & (m(d) \geq \bar{m}) \\ \Phi(d) * (1 + \theta\frac{\bar{m}-m(d)}{\bar{m}}) & (m(d) < \bar{m}), \end{cases} \tag{11}$$

where θ is a constant $(-1 < \theta < 1)$. The change in potential can be larger when it is away from the mean value; conversely, the change can be smaller when it is closer to the mean value. At this time, so as to avoid an aberrant value of potential, the potential is taken to be within a range decided beforehand, $[\Phi_{min}, \Phi_{max}]$.

4. SIMULATION RESULTS

We evaluate the impact of CPBR through computer simulation by using an event-driven simulation program we made written in Visual C++. The network model is a square (length of each side: 600 m) in which 150 sensor nodes are randomly deployed and 3 sink nodes (sink 1, sink 2, and sink 3) are set at points (300, 300), (100, 300), and (500,100), respectively as described in Figure 2. The communication range of each node is 100 m. We assume that data packets are generated by each sensor node according to a Poisson process with intensity λ and are sent to the sink node by multihop relay. The simulation commences after an initializing phase, in which each node sufficiently exchanges its potential with neighbor nodes. The simulation ends at 20,000 seconds. The interval of ID transmission is 1.0 s, and T_p is also set to 1.0 s. Other parameters are set as shown in Table 1. Note that θ decides the rate of increase and decrease of potential. Here, a positive value is used for θ to increase Φ when m is larger than $\bar{m}$ and vice versa. For T_m, the interval of the potential control, two values are used. T_m is set to 500 s when the metric value is the number of received data packets because the frequency of data generation ($= \lambda$) is not too high, and T_m of 50 s is used for the energy consumption rate (in this case, T_m depends on T_i, the standby interval for receiving potentials). When a node receives a potential during T_p, it immediately returns its own potential together with its remaining energy. Since this sampling period occurs at intervals of T_i, the remaining energy at neighbor nodes is updated within $\frac{T_i}{2}$ on average. Therefore, the potentials of the sink nodes can be comparatively controlled over a short cycle.

Table 1: Parameter settings

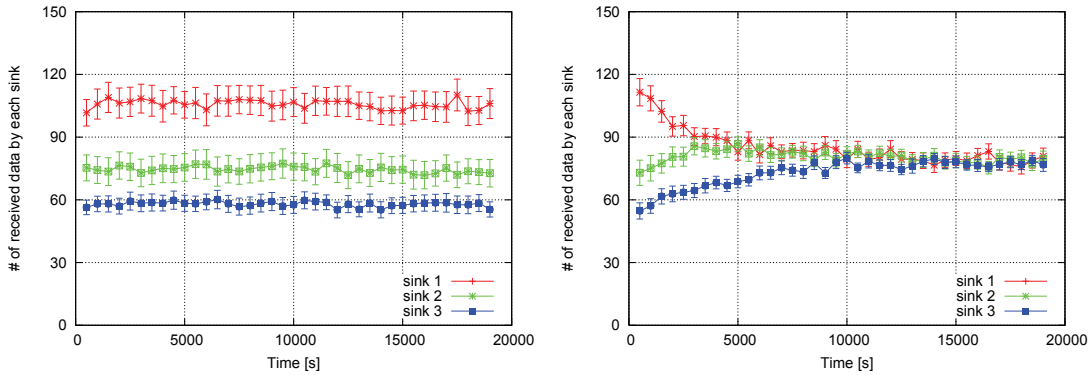
Parameter	Value
λ	0.003 packet/s/node
Current consumption (TX)	20 mA
Current consumption (RX)	25 mA
Current consumption (SLEEP)	0 mA
T_i	100 s
T_m	50 s or 500 s
Φ_{init}	-30
Φ_{min}	-90
Φ_{max}	0
θ	0.5

4.1 Load Balancing of Sink Nodes

Potential control based on the numbers of received data, $N_{dt}(d)$, can balance the load of the sink nodes. Simulation results are shown in Figure 3. In this figure, the y-axis indicates the number of data packets received by each sink node during T_m $(500\ s)$ with 95% confidential interval. When sink nodes do not control potentials (we call it "autonomous" for simplicity), the number of received data packets remains mostly unchanged over time. With potential control of sink nodes (we call it "controlled" for simplicity), the number of received data packets (over 3 sink nodes) converges to a nearly identical value. This identical value equals $\lambda T_m \frac{N_n}{N_s}$ $(= 75)$, where N_n is the number of sensor nodes. Convergence time is about 10000 s, which indicates that 20 times control causes the number of received data packets to converge. When considering the operating time of an actual sensor network system, which can be in units of years, we note that convergence within a realistic time is possible.

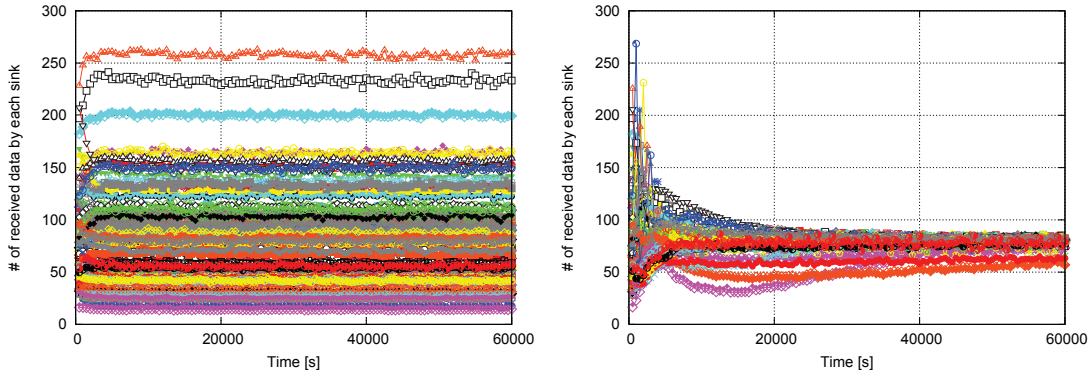
Figure 4 shows the number of received data packets in a network where 5,000 sensor nodes and 100 sink nodes are randomly deployed. The network field shapes a square (length of each side: 3500 m). Also in this case, potential control works properly without a significant change in convergence time; however, there are some issues. The deviation is larger than that of Figure 3(b). This is an inevitable result of a self-organization mechanism when a network grows in scale.

Next, we examine the adaptivity of CPBR to the heterogeneous density of sensor nodes. In Figure 5, we show the impact of potential control when there is a difference in density of sensor nodes: We use a square field, with a 600 meter long side, where the right half of the field has double node density of the left half. CPBR can



(a) Autonomous (b) Controlled

Figure 3: Potential control based on the number of received data packets (3 sinks)



(a) Autonomous (b) Controlled

Figure 4: Potential control based on the number of received data packets (100 sinks)

attain good convergence of the number of received data packets also in this case. This indicates our proposed potential control can adaptively accommodate heterogeneous density of sensor nodes.

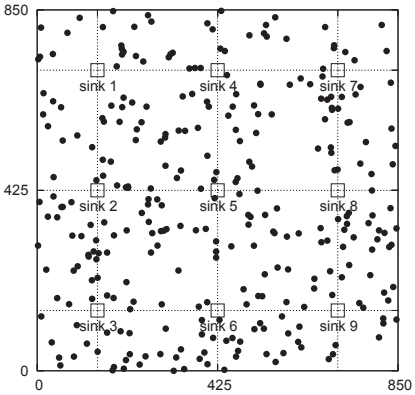


Figure 6: Network model (9 sink nodes)

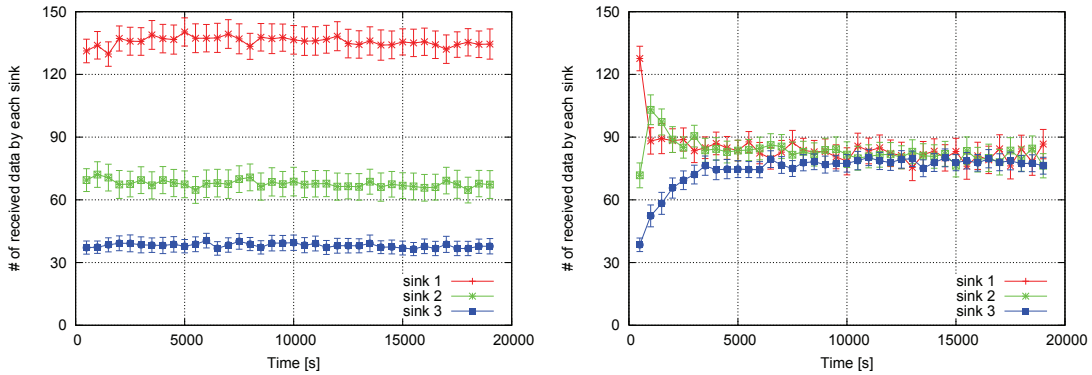
We also demonstrate the robustness of CPBR against sink node failures and additions. Here, we randomly deployed 300 sensor nodes over a square-shaped network (length of each side: 850 m), and placed 9 sink nodes at locations (142, 708), (142, 425), (142, 142), (425, 708), (425, 425), (425, 142), (708, 708), (708, 425), and (708, 142) (denoted by sink 1 to sink 9 respectively) as shown in Figure 6. Here, the number of sink nodes is varied as follows.

- Six sink nodes (2, 4, 6, 7, 8, and 9) are active at 0 second.
- After four hours, two sink nodes (6 and 9) break down.
- Eight hours after that failure, two sink nodes (1 and 3) are added.
- At twelve hours into the simulation, a sink node (5) is added.

Results of the simulation are shown in Figure 7. Comparing Figure 7(a) and Figure 7(b), we find that CPBR can control potentials adequately after failures and additions of sink nodes; thus, CPBR is robust against sink node failures and additions.

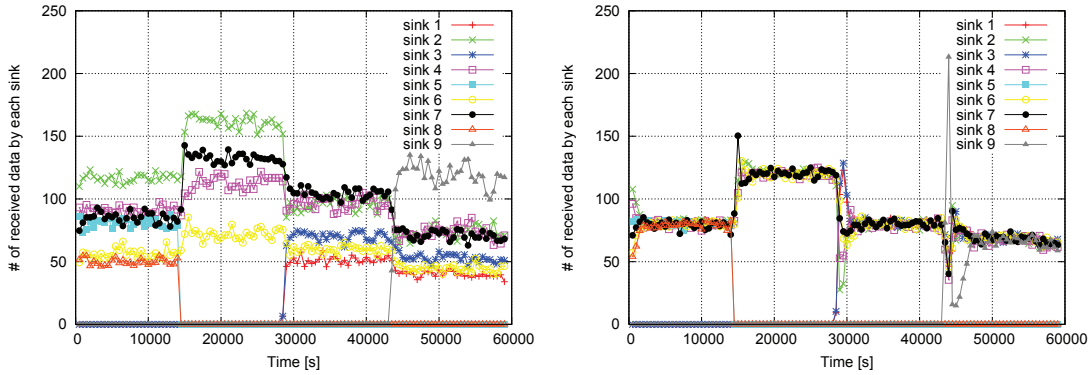
4.2 Load Balancing of Sensor Nodes

Balancing the load of sensor nodes based on the residual energy is expected to be accomplished by potential control based on energy consumption rate. Figure 8 is a simulation result where potential control based on $P_{rt}(d)$ is used. The network model is same as that of Figure 2. The y-axis of the figure indicates the energy consumption of each node with 95% confidential interval in 6-hour simulation, while the x-axis represents each node sorted by



(a) Autonomous (b) Controlled

Figure 5: Potential control based on the number of received data packets (heterogeneous node density)



(a) Autonomous (b) Controlled

Figure 7: Potential control based on the number of received data packets (failures and additions of sink nodes)

descending order. In our potential-based routing, relay load is concentrated on the node that has the minimum potential among the neighbors of the sink node. Hence, once a potential field has been constructed, the relay load remains concentrated on a specific node (as apparent in Figure 8(a)). In this figure, because the number of received data packets at sink 1 is largest, the energy consumption of the heaviest loaded node is larger. Figure 8(c) indicates that potential control can reduce the energy consumption of that node. This is because the number of neighbor nodes at each sink nodes is nearly equal; therefore, the number of data packets received by the sink nodes are also nearly equal. If the number of neighbor nodes differs considerably, the bias of energy consumption may grow even more than the result shown in Figure 8(a). In any situation where there is a major difference among the number of neighbor nodes, the local load-balancing mechanism described in Section 2.2.3 can substantially reduce the energy consumption of the heaviest loaded node as shown in Figures 8(b) and 8(d). The results of CPBR with a local load-balancing mechanism are shown in Figure 8(d). While the total energy consumption rises due to an increase of detours, a 58.0 % reduction in the energy consumption of the heaviest loaded node was attained comparing with the result shown in Figure 8(a). Given that energy is consumed only by communication, we find that CPBR with load balancing acted to extend the time until the first node runs out of energy by 138% ($= \frac{0.58}{1-0.58} \times 100$).

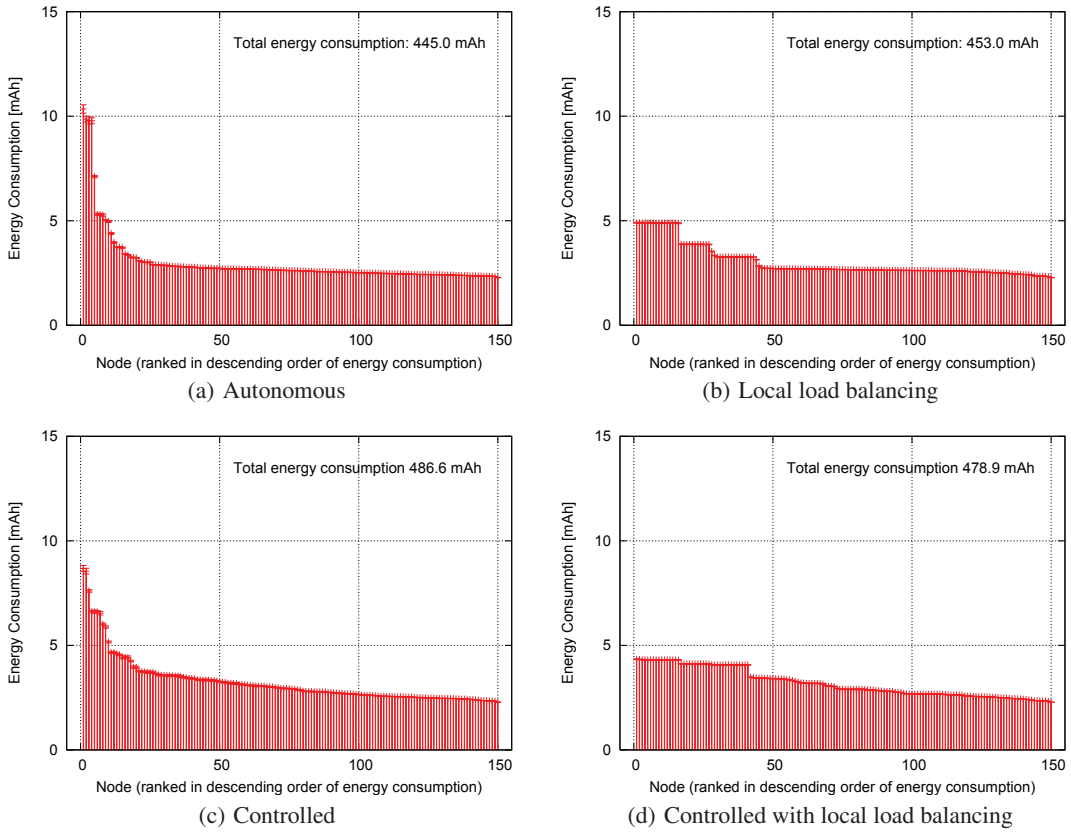
Figure 9 shows the network lifetime based on the number of alive nodes and network lifetime based on reachability to sink nodes. There are various definitions for network lifetime in sensor networks depending on applications [3] and in this paper we use following two simple definitions.

1. The time until the first node depletes its energy (alive node).
2. The time until 20% of nodes lose reachability to sink nodes (80% reachability).

For evaluating network lifetime, we set the battery of sensor nodes to comparatively small value (1.0 mAh) and simulation time is set to longer time than battery lifetime. Comparing "controlled with local load balancing" with "autonomous" in Figure 9(a), the time until the first node depletes its energy is prolonged to more than twice as remarked above. In terms of the time guaranteeing 80% reachability, that of CPBR with load balancing ("controlled with local load balancing") is 11.5% shorter than that of default potential-based routing (autonomous) because CPBR increases total energy consumption. However, the time of CPBR ensuring 100% reachability is 2.14 times longer than that of default.

5. CONCLUSION

In a controlled self-organization scheme intended to ensure desired network behavior, one or more controllers control a portion



(a) Autonomous (b) Local load balancing

(c) Controlled (d) Controlled with local load balancing

Figure 8: Potential control based on neighbors' energy consumption rate (energy consumption distribution)

of nodes operated in self-organization with centralized control, distributed control, or some other control scheme. In this paper, we proposed controlled potential-based routing (CPBR), which is based on a controlled self-organization scheme. In this scheme, sensor nodes calculate their own potential in a self-organized manner while sink nodes control their own potentials according to network manager's demands by distributed control so as to construct a desired potential field. Our CPBR is over IRDT protocol, however, it is not limited to IRDT and it is applicable also in the sensor network where other MAC protocols were adopted. Through computer simulation, we show that load balancing of the sink nodes can be attained in diverse situations with potential control based on the number of data received at each sink nodes. We also show that CPBR, with potential control based on the energy consumption rate, can extend the time until the first node deplete its energy by 138 %. Presently in our laboratory, investigations into the variations in system behavior and in convergence time as network size becomes increasingly large are underway. Additionally, on the subject of our routing scheme, further evaluation on robustness and detailed study on the dependence on node density are our future work.

Acknowledgment

This research was supported in part by "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan, and "Grant-in-Aid for Scientific Research (A) 21240004 and (C) 23500097" of the Japan Society for the Promotion of Science (JSPS) in Japan

6. REFERENCES

[1] A. Basu, A. Lin, and S. Ramanathan. Routing Using Potentials: A Dynamic Traffic-Aware Routing Algorithm. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pages 37–48, Aug. 2003.

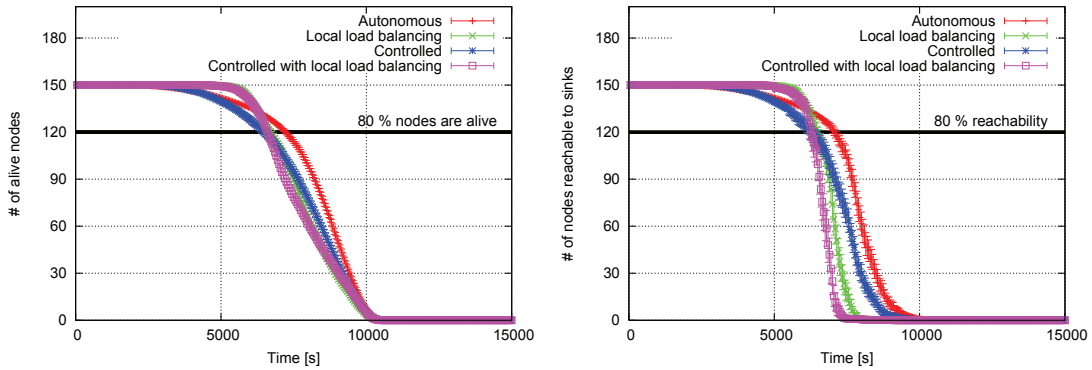
[2] J. Branke, M. Mnif, C. Müller-Schloer, H. Prothmann, U. Richter, F. Rochner, and H. Schmeck. Organic Computing–Addressing Complexity by Controlled Self-Organization. In *Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA)*, pages 185–191, 2006.

[3] I. Dietrich and F. Dressler. On the Lifetime of Wireless Sensor Networks. *ACM Transactions on Sensor Networks*, 5(1):1–39, Feb. 2009.

[4] F. Dressler. *Self-Organization in Sensor and Actor Networks*. Wiley, 2007.

[5] S. Jung, M. Kserawi, D. Lee, and J.-K. K. Rhee. Distributed Potential Field Based Routing and Autonomous Load Balancing for Wireless Mesh Networks. *IEEE Communications Letters*, 13(6):429–431, 2009.

[6] M. Kalantari and M. Shayman. Design Optimization of Multi-Sink Sensor Networks by Analogy to Electrostatic Theory. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pages 431–438, Apr. 2006.



(a) Alive node (b) 80% reachability to sink nodes

Figure 9: Potential control based on neighbors' energy consumption rate (network lifetime)

[7] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi. Energy-Efficient Receiver-Driven Wireless Mesh Sensor Networks. *Sensors*, 11(1):111–137, Jan. 2011.

[8] P. Kumar, J. Kuri, P. Nuggehalli, M. Strasser, M. May, and B. Plattner. Connectivity-Aware Routing in Sensor Networks. In *Proceedings of the International Conference on Sensor Technologies and Applications (SENSORCOMM)*, pages 387–392, Oct. 2007.

[9] J. Li, S. Ji, H. Jin, and Q. Ren. Routing in Multi-Sink Sensor Networks Based on Gravitational Field. In *International Conference of Embedded Software and Systems (ICESS)*, pages 368–375, July 2008.

[10] H. Liu, Z.-L. Zhang, J. Srivastava, and V. Firoiu. PWave: A Multi-Source Multi-Sink Anycast Routing Framework for Wireless Sensor Networks. *NETWORKING 2007. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, pages 179–190, 2010.

[11] Y. Liu and W. K. G. Seah. A Scalable Priority-Based Multi-Path Routing Protocol for Wireless Sensor Networks. *International Journal of Wireless Information Networks*, 12(1):23–33, Jan. 2005.

[12] C. Müller-Schloer, H. Schmeck, and T. Ungerer. *Organic Computing–A Paradigm Shift for Complex Systems*. Birkhäuser, 2011.

[13] H. Ochiai and H. Esaki. Message Routing on Potential-fields in Forwarding-based DTNs. In *Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication (ICUIMC)*, pages 185–193, Jan. 2009.

[14] Y. Sun, O. Gurewitz, and D. B. Johnson. RI-MAC: A Receiver-Initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks. In *Proceedings of the 6th ACM conference on Embedded network sensor systems (Sensys)*, pages 1–14, Nov. 2008.

[15] C. Wu, R. Yuan, and H. Zhou. A Novel Load Balanced and Lifetime Maximization Routing Protocol in Wireless Sensor Networks. In *Proceedings of the IEEE Vehicular Technology Conference (VTC)*, pages 113–117, May 2008.

[16] J. Zhang, Q. Wu, F. Ren, T. He, and C. Lin. Effective Data Aggregation Supported by Dynamic Routing in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1–6, May 2010.