Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2011) 1–8

Procedia Computer Science

The 2nd International Conference on Ambient Systems, Networks and Technologies (ANT-2011)

Proposal for Dynamic Organization of Service Networks over a Wireless Sensor and Actuator Network

Takuya Iwai, Naoki Wakamiya, Masayuki Murata

Graduate School of Information Science and Technology, Osaka University, Suita, Osaka 565–0871, Japan

Abstract

In the ambient information society, embedded sensors detect and conjecture environmental and personal conditions. Then, actuators provide users with information services and environmental controls which are suited for time, place, occasion, and people. Although a single service cannot meet diverse requirements of different situations, it is hard to deploy and configure a variety of devices for each of envisioned services. To solve the problem, we propose a mechanism to dynamically self-organize service networks by combining existing devices. In our proposal, we adopt a mathematical model of division of labors in a colony of social insects to accomplish autonomous selection of devices, which offer sensing and control functions to a service network, while taking into account their states and service requirements. Through simulation, we confirmed that our proposal can organize efficient service networks combining nodes with higher residual energy and more service provision.

Keywords: ambient information society, wireless sensor and actuator network, task allocation, response threshold model

1. Introduction

In recent years, many researchers have been actively working in the field wireless sensor and actuator networks (WSANs). A WSAN consists of embedded sensors which detect and conjecture environmental and personal conditions and actuators which control environment and machines. By distributing nodes with appropriate sensors and/or actuators at appropriate locations in an area, e.g. field, building, and room, and organizing a network by wireless communication, a variety of services can be provided in the area. We hereafter call sensors and actuators *devices* and a *node* corresponds to an equipment with CPU, memory, wireless transceiver, and one or more devices.

In the forthcoming ambient information society [1], based on various information about environment and people collected from sensors, users are provided with desirable information services and environmental and machinery controls, which are suited for time, place, occasion, and people. On the contrary to the fact that a variety of requirements arise in such context-aware service provisioning, most of WSANs are constructed and managed in an application-oriented manner to answer specific requirements of an individual application. Nodes are deployed for a specific application and applications do not share them. For example, both of WSANs for intelligent indoor temperature control and intrusion detection employ nodes with a motion sensor to detect location of people in a room, but they are made of dedicated nodes and independent from each other. If they share those nodes and their sensing data, network resources and energy consumption can be saved. Furthermore, application-oriented deployment requires previous knowledge and careful planning about types and locations of nodes to be deployed. However, it is hard to predict all events that may occur in the area and prepare WSANs for all service requirements envisioned. Distributing

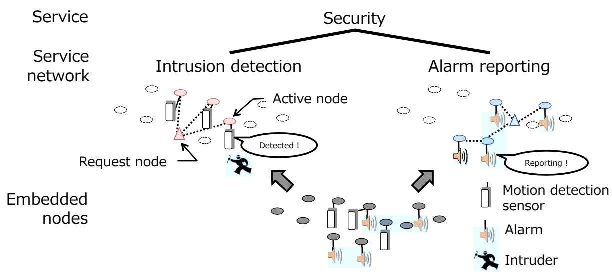


Figure 1: Image of service networks

heterogeneous nodes and organizing them as a single and monolithic WSAN is one of solutions to handle dynamic and diverse requirements of services. However, it is bandwidth- and energetically-expensive to control and manage a large WSAN.

In this paper we propose a mechanism to dynamically and autonomously organize service networks by combining nodes deployed in the area to satisfy requirements of emerging services. We consider a service consists of one or more service networks, each of which provides a group of functions to the service. For example, assume that a security service requires functions of intrusion detection and alarm reporting. For each of functions, a service network is organized as shown in Fig. 1. In this figure, nodes equipped with sensing and/or actuating devices, including an alarm and an intrusion detection sensor, are distributed in the region. By combining these devices, a service network for intrusion detection is statically organized, while a service network for alarm reporting will be organized only when an intruder is detected. A service network consists of a *request node* which initiates organization of the service network, *member nodes* which are equipped with devices to satisfy service requirements, and *relay nodes* which deliver messages among a request node and member nodes. In addition, there are two types of member nodes, i.e., *active nodes* and *idle nodes*. Active nodes offer sensing and/or control functions of devices to one or more services. Active nodes are appointed taking into account several conditions such as service requirements, the degree of engagement in service networks, and the residual energy to efficiently share active nodes among service networks and balance energy consumption among active nodes. To organize service networks in an autonomous manner, we adopt a response threshold model [2], which imitates a mechanism of division of labors in a colony of social insects. From autonomous decision of individuals based on a simple rule, there emerges a group of insects engaged in a certain task and the size of group is dynamically adapted to the demand of the task. In our proposal, a request node advertises the demand for construction of a service network and each node autonomously decides whether to become a worker, i.e. an active node, or not.

The remainder of this paper is organized as follows. First, in section 2 we briefly discuss related work. Next, in section 3 we propose a mechanism for dynamic self-organization of service networks. Then, in section 4, we show results of preliminary simulation experiments to evaluate our proposal. Finally, in section 5, we conclude the paper.

2. Related work

There are several proposals to effectively use heterogeneous nodes and multiple WSANs. TinySOA [3] is based on the concept of SOA (Service Oriented Architecture) to allow application developers to write application programs without concerning differences in node architecture, OS, and programming languages. In our proposal, we assume SOA-based nodes where a common API is provided and they can be managed in an integrated manner.

To share sensors among multiple services, TinyONet is proposed in [4]. A sink manages virtual sensors each of which corresponds to a physical node and groups them into a slice in accordance with service requirements. From a viewpoint of service, a dedicated sensor network, i.e. a slice, is tailored over heterogeneous sensors. Since the authors' focus is on reusability of sensing data, TinyONet assumes that sensing data are periodically collected from all physical sensors to the sink. On request from a service, a virtual sensor returns cached sensing data as if it were obtained from a physical sensor. To keep cached data up-to-date, TinyONet consumes bandwidth and energy and it does not fit to a WSAN. In [5], the authors propose VSNs (Virtual Sensor Networks) which is constructed over independent WSANs. In their proposal, a VSN is constructed of, based on our naming, member nodes and relay

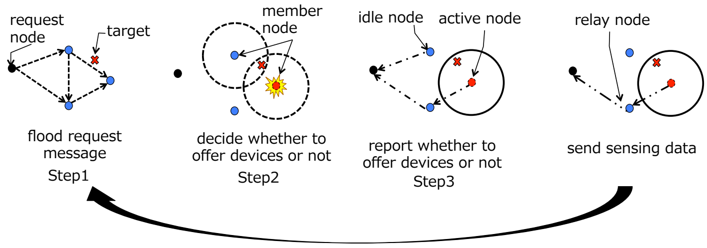


Figure 2: Overview of proposal

nodes belonging to different WSANs, forming the tree topology. A VSN allows a service-oriented and inter-WSAN overlay, but they do not specify the way that member nodes are selected. Furthermore, it is inefficient to make all messages pass through a root node though it benefits from data aggregation.

Regarding on-demand selection of nodes which offer functions to a service, [6] proposes an algorithm for generic role assignment. An application developer provides a role specification using if-then-else statements. It is translated to a role specification message and distributed to nodes by a gateway. Each node decides whether to become ON or OFF based on the received message and property information. Directed diffusion is another example of a node selection mechanism [7]. A sink first distributes an interest message which specifies required functions and conditions to be satisfied. Next, on receiving an interest message, nodes which can answer the request begin to transmit sensing data to the sink. Then, a sink reinforces a path to a better source node, which provides the sink with sensing data first for example. Although those proposals offer the flexibility of node selection to application developers, it is hard to consider all possible conditions and define a complicated rule to select appropriate nodes without errors and contradictions.

3. Dynamic organization of service network

Our proposal adopts a response threshold model of division of labors in a colony of social insects [2] to accomplish dynamic organization of service networks by autonomic and simple decision of each node.

3.1. Overview of our proposal

We assume an area where a large number of heterogeneous nodes equipped with a variety of devices. No WSAN exists at the beginning, while nodes within the range of radio signals can communicate with each other. Organization of a service network is initiated by a request node. We should note here that our proposal can be applied to organization of service networks for both of static and dynamic services. In the case of static services, a request node is a sink of data of periodic monitoring, for example. In the case of dynamic services, a node detecting an event becomes a request node, for example. A request node first disseminates a request message, which specifies required functions, to all nodes (step 1 in Fig. 2). If location information is available, message dissemination can be restricted to the area of interest for the sake of efficiency.

A node which receives a request message and is equipped with devices for the requested functions becomes a member node of the service network. Then, it decides whether to offer devices or not by evaluating equations of the response threshold model (step 2). Decision is reported to the request node by sending a notification message (step 3). Nodes where a notification message traverses become relay nodes and they adjust sleep scheduling if necessary. A node which decides to offer devices begins to obtain sensing data from sensors and/or activate actuators. We call a device which provides a sensing or actuation function to a service network an *active device* and a node with one or more active devices an *active node*. The other member nodes are *idle nodes* and their devices are *idle devices*. Sensing data and/or control information are exchanged among a request node and active nodes via relay nodes until the next timing of periodic dissemination of request messages.

Those steps are repeated while the corresponding service is required, but contents of a request message can change to reflect changes in the context a member node of a service network can initiate organization of a new service network by issuing a new request.

3.2. Internal values of node

A member node maintains a set $\mathbf{S}$ of tuples $(i, k, h, s_i(t))$, which is updated on receiving the t-th request message. i is an identifier of a service network. The identifier, which is unique in the whole network, can be generated as concatenation of a node identifier and a sequence number of service network it initiates. k is a sequence number of request message. h is an identifier of a neighbor node from which it receives the corresponding request message. Finally, $s_i(t)$ is the demand of service network i advertised in the t-th request message by a request node.

A member node also maintains a set $\mathbf{X}$ of $X_{i,j}$ and a set $\mathbf{\Theta}$ of $\theta_{i,j}$ as variables of a response threshold model. $X_{i,j}$ represents the state of device j. j can be either of a number, an attribute, or a name of the device. If device j is offered to service network i, $X_{i,j}$ is set at $true$. Otherwise, it is $false$. $\theta_{i,j}$ is a threshold of device j for service network i.

3.3. Node behavior

A request node disseminates a request message at regular intervals of I_{demand} s. We call an interval between two successive emissions of request messages a *round*. For example, assume a service network for a period data gathering which requires the condition of location at coordinates (x, y) with a motion detection sensor to be reported to a request node every I_{data} s. In this case, a request message emitted at the t-th round is in the form of $(i, motion\ detection\ sensor, k, s_i(t), (x, y), I_{data})$. $s_i(t)$ is the demand of service network i calculated at the beginning of the t-th round. A request message can be easily extended to request multiple functions at the same time.

When a node receives a request message of service network i for the first time, it generates a new tuple $(i, k, h, s_i(t))$ and adds it to the set $\mathbf{S}$. Otherwise, it updates the existing tuple of service network i. Next a node examines whether it has a device which satisfy the request. A node, which is not equipped with the corresponding devices, is a candidate of a relay node. It does not offer any devices but relay messages from and to the request node and member nodes. On the contrary if it has, a node is a member node. It generates and adds new values $X_{i,j}$ and $\theta_{i,j}$ to sets $\mathbf{X}$ and $\mathbf{\Theta}$, respectively, if they do not exist. Initially, $X_{i,j}$ is $false$ and $\theta_{i,j}$ is 0.5. Then, newly generated or existing $X_{i,j}$ and $\theta_{i,j}$ are updated by using equations explained in the next section and decision is made. Decision is reported to the request node by a means of notification message which contains an identifier of the node, an identifier of the service network, and $X_{i,j}$. A notification message is sent to neighbor h, from which a node received the corresponding request message. Following a reverse path, a notification message reaches the request node. Active nodes, whose $X_{i,j}$ is *true*, begins requested functions and sends or receives messages if necessary. In the above example, an active node sends sensing data obtained by a motion detection sensor observing a point (x, y) to the request node at regular intervals of I_{data} s. Every time a member node receives a request message, the above steps are conducted. If a member node does not receive a request message for E_i s, it considers the corresponding service network is not valid any more and it removes corresponding tuples and values from sets $\mathbf{S}$, $\mathbf{X}$, and $\mathbf{\Theta}$.

A request node receives notification messages from member nodes. The information is used to obtain the number of active nodes and the number of member nodes and then to derive the demand. The updated demand is notified to member nodes by a request message at the beginning of the next round. A request node also receives sensing data and other information from active nodes and sends control messages to operate actuators.

3.4. Response threshold model-based decision making

A response threshold model [2] is a mathematical model which imitates a mechanism of adaptive division of labors in a colony of social insects. A colony is divided into two groups of workers and non-workers based on autonomous decision of individuals using a simple rule and the size of groups is well adjusted to satisfy the demand of task. In dynamic organization of a service network, a node decides whether to make a device active or not using this model and the service network is always composed of the appropriate number of active nodes to the demand of service network.

Each of member nodes stochastically decides to whether to provide a service network with a device. A member node has a state value $X_{i,j} \in \{true, false\}$ representing whether device j is active or not for service network i. The probability $P(X_{i,j} = false \rightarrow X_{i,j} = true)$ that a node, which is idle ($X_{i,j} = false$), begins providing the service network with the device is derived by the following equation.

$$P(X_{i,j} = false \rightarrow X_{i,j} = true) = \frac{s_i(t)^2}{s_i(t)^2 + \theta_{i,j}(t)^2 + A_j} \quad (1)$$

$s_i(t)$?(≥ 0) is the demand of service network i at the t-th round. $\theta_{i,j}$ ($1 \geq \theta_{i,j} > 0$) is a threshold which corresponds to hesitation of the node in providing service network i with device j. A_j (> 0) is a variable to take into account the degree of engagement in service networks and the residual energy, which does not exist in the basic response threshold model [2]. The derivation of A_j will be explained in the next section. In our proposal, we define the demand s_i of service network i as follows.

$$s_i(t+1) = s_i(t) + \delta_i - \frac{N_i(t)}{M_i(t)} \tag{2}$$

δ_i ($1 \geq \delta_i \geq 0$) is the increasing rate of demand of service network i. $N_i(t)$ (≥ 0) is the number of active nodes, which is equal to the number of notification messages stating $X_{i,j} = true$ received in response to the t-th request message. $M_i(t)$ (≥ 1) is the number of notification messages received in response to the t-th request message. To prevent idle nodes from becoming active nodes redundantly, the initial demand $s_i(0)$ is set at 0. When the ratio of active nodes to member nodes is less than δ_i, the demand of service network i gradually increases and more idle nodes becomes active.

The propability that an active node stops providing service network i with device j is derived by the following equation.

$$P(X_{i,j} = true \rightarrow X_{i,j} = false) = p_j, \tag{3}$$

where p_j ($1 \geq p_j \geq 0$) is a constant defined per device. This enables rotation of providing the service network with the device among member nodes and prevents redundant active nodes from providing the service network with the device. The average duration that an active node provides the service network with the device is $1/p_j$ rounds. As the result, the ratio of active nodes to member nodes converges to δ_i.

In similar with the basic response threshold model [2], our proposal also have a mechanism of reinforcement which makes specialists. Threshold $\theta_{i,j}$ is adjusted as follows in response to the updated $X_{i,j}$.

$$\theta_{i,j} = \begin{cases} \theta_{i,j} - \xi_j, & \text{if } X_{i,j} \text{ is } true \\ \theta_{i,j} + \varphi_j, & \text{if } X_{i,j} \text{ is } false \end{cases} \tag{4}$$

ξ_j (> 0) and φ_j (> 0) are parameters related to the speed of differentiation. By reinforcement, a member node is more likely to be active once it becomes active. Eventually, the service network is composed of specific active nodes.

3.5. Derivation of variable A_j

In our proposal, variable A_j (> 0) is derived from the degree of engagement in service networks for efficient sharing of devices among multiple service networks and the residual energy for balancing energy consumption among member nodes for longer lifetime. Although there are various ways of considering multiple factors, we adopt simple addition as follows.

$$A_j = c_j + e \tag{5}$$

c_j (≥ 0) and e (> 0) correspond to the degree of engagement in service networks and the residual energy, respectively. In the similar way, other conditions such as reliability of sensing data and radio quality can be taken into account.

Variable c_j is derived as follows.

$$c_j = \begin{cases} W_c \times \left(1 - \frac{F_j}{S_j - 1}\right)^n, & \text{if } S_j > 1 \\ 0, & \text{if } S_j = 1 \end{cases} \tag{6}$$

Parameter W_c (> 0) is a weighting coefficient to balance the degree of engagement in service networks and the residual energy. Exponent n (≥ 1) influences the degree of engagement in service networks. Variable S_j (≥ 1) is derived as $S_j = |\{X_{i,j} \in X \mid i \in I\}|$. I is a set of identifiers of service networks of which a node is a member node. Variable F_j (≥ 0) is derived as $F_j = |\{X_{i,j} \in X \mid i \in I, X_{i,j} = true\}|$ and $F_j \leq S_j - 1$. When there is 1 service network ($S_j = 1$), the node does not need to consider the degree of engagement in service networks and c_j is 0.

Table 1: Parameter setting

Parameter	Description	Value
p_j	probability of quitting task in Eq. (3)	0.01
δ_i	increasing rate of demand in Eq. (2)	0.1
ξ_j	threshold adaptation parameter in Eq. (4)	0.01
φ_j	threshold adaptation parameter in Eq. (4)	0.1
W_c	weight of degree of engagement in Eq. (6)	10
n	exponent coefficient in Eq. (6)	10
W_e	weight of residual energy in Eq. (7)	10
g	gain of sigmoid function in Eq. (7)	50
b	shift of sigmoid function in Eq. (7)	0.6

Variable e is defined as,

$$e = W_e \times \left(1 - \frac{1}{1 + \exp^{-g\times(P_{res}/P_{full}-b)}}\right), \tag{7}$$

where W_e (> 0) is also a weighting coefficient. P_{res} and P_{full} are the amount of residual energy and the total capacity of batteries of a node, respectively. Parameter g (> 0) is a gain of the sigmoid function. Parameter b (> 0) shifts the sigmoid function and determines an inflection point. When the ratio of P_{res} to P_{full} is larger than b, e is close to 0 and does not contribute to probability P in Eq. (1). However, once the ratio decreases below b, e increases to nealy W_e and probability P becomes small.

4. Simulation evaluation

In this section we first briefly explain directed diffusion [7] used for comparison. Then, through simulation, we verify that service networks can be autonomously organized and our proposal outperforms directed diffusion in terms of the number of active nodes and the residual energy.

4.1. Directed diffusion

In directed diffusion, a sink node which corresponds to a request node in our proposal first disseminates an interest message. An interest message specifies a required sensing task by using a naming scheme. Initially, a reporting interval is set longer than one that an application requires. Nodes that can perform the requested task begin to send sensing data in the form of exploratory message to the sink node. On receiving exploratory messages, the sink node sends an interest message with a shorter interval, which is called a reinforcement message, to one of neighbors. For example, a neighbor node from which a sink node receives an exploratory message first is chosen. A reinforcement message is forwarded to a node generating sensing data. In this way, a sink node can obtain sensing data at the desired rate from the best source while keeping other nodes sending sensing data at lower rate for robustness against failures.

4.2. Simulation model

In our simulation, we used OMNet++ [8] as a simulator and the IEEE 802.15.4 for radio communication. 25 nodes form a grid with a separation of 10 m in the area of 40 m × 40 m. Nodes are assumed to be identical and have the same capability. A sensing device can obtain sensing information within the radius of 20 m. The communication range is 15 m. A node runs on two AA batteries of 3.3 V. Based on the datasheet of MICAz [9], a transceiver module consumes 18.8 mA in listening a channel and receiving a message, 17.4 mA in transmitting a message, and 0.021 uA in a sleep mode. An active device additionally consumes 20 mA and an idle device does 0.1 mA. The length of a request message of our proposal, an interest message and a reinforcement message of directed diffusion is set at 36 byte. Regarding a notification message of our proposal, an exploratory message of directed diffusion, and a sensing data message, the length is set at 64 byte. Parameters of our proposal are summarized in Table 1.

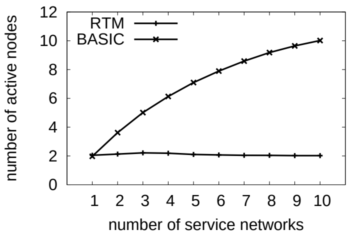
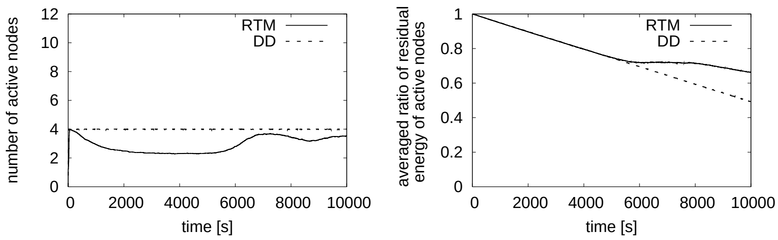


Figure 3: Comparison with basic response threshold model



(a) number of active nodes

(b) ratio of residual energy of active nodes

Figure 4: Comparison with directed diffusion

4.3. Basic evaluation

First, we verify that our proposal can organize service networks which share active nodes. We placed a gateway node at coordinates $(0, 0)$ which plays a role of a request node of multiple services. Each service requires a service network to be organized. The request node packs requests from multiple services into one request message and sends it at regular intervals of 5 s. We change the number of service networks to organize from 1 to 10, but they all requires the same sensing task to monitor the point at coordinates $(20, 20)$ every 1 s. In our node layout, there are 12 nodes which can monitor the point.

Figure 3 shows simulation results averaged over 500 runs. The X-axis corresponds to the number of service networks and the Y-axis shows the average number of active nodes at time 5000 s. For comparison, we consider a scheme which uses the response threshold model without variable A_j, indicated as *BASIC*. At 5000 s, the amount of residual energy is large enough not to affect the value of A_j. As shown in the figure, our proposal, indicated as *RTM*, can maintain the number of active nodes as small as 2 independently of the number of service networks. It implies that service networks effectively share active nodes. In our proposal, a node with a device which has been offered to other service networks is more likely to decide to become active for a new service network than idle nodes. A reason that two nodes become active is that the increasing rate of demand, i.e. δ_i, is set at 0.1. It means that the number of active nodes must be more than 1.2 to compensate the increase in the demand.

On the contrary, whereas Eq. (4) contributes to separation of nodes into active and idle groups to some extent in BASIC, it only controls the number of active nodes in a service network. As a result, the average number of active nodes per service network is about 2, but the average number of service networks that a node offers a device to is about 2 in the case of 10 service networks.

4.4. Comparison with directed diffusion

In this section, we compare our proposal with directed diffusion. Four nodes located at the corners of the area are request nodes in our proposal or source nodes in directed diffusion. They sends a request message or an interest message every 5 s to obtain sensing data about the point at coordinates $(20, 20)$ every 1 s. Regarding directed diffusion, the initial reporting interval specified in an interest message is set at 10000 s. Since the simulation time is 10000 s, nodes which are not reinforced do not send sensing data, which favors directed diffusion in evaluation of energy consumption. A sink node of directed diffusion sends a reinforcement message to a neighbor from which it receives an exploratory message first.

In Fig. 4, simulation results averaged over 500 runs are depicted. Results of directed diffusion are denoted as *DD*. In Figs. 4(a) and 4(b), the x-axis shows simulation time and the y-axis shows the average number of active nodes and the average ratio of residual energy to capacity of active nodes, respectively. As shown in Fig. 4(a), the average number of active nodes in DD is 4. A reason why the number is 4 is that each of sink nodes located at the corners chooses a node which is the closest to itself. Since active nodes do not change during a simulation run, they steadily consumes energy as time passes (see Fig. 4(b)).

In the case of our proposal, the average number of active nodes first increases to 4. Since there are no active nodes at the beginning, the demand increases. By being stimulated by the large demand, many nodes decide to offer a device to the request at the same time. Eventually, as they contributes to reduction of the demand, the average number of active nodes gradually decreases. Finally, the average number of active nodes becomes almost constant at about 2.18, which balances the increase and decrease of the demand. Because of adaptation of a threshold formulated as Eq. (4), a few specific nodes are active during the period until about 5500 s and the other member nodes remain idle. From about 5500 s, the average number of active nodes begins to increase. At this time, the average ratio of residual energy of active nodes stops decreasing in our proposal as shown in Fig. 4(b). It is because active nodes which deplete energy quit offering a device and idle nodes with more energy take over their task. Since it is not possible to hand over all service networks from a former active node to a new active node at once, the average number of active nodes is redundant during the transitional stage. The average number of nodes briefly decreases around 8200 s as a result, but it increases again. A reason for this is that new active nodes have already consumed energy in receiving request messages and sending notification messages while they are idle. Therefore, a new transition stage starts. Obviously, in our node layout, there are twelve nodes which can monitor the point. We can further extend the period by saving energy consumption at idle nodes, but it remains future work.

From the above discussion based on preliminary evaluation, we can conclude that our proposal can keep the number of active nodes smaller than the directed diffusion and let nodes with more energy take over the task from exhausting nodes through their autonomous decision.

5. Conclusions and future work

In this paper, we proposed the mechanism to organize service networks from distributed heterogeneous nodes in an autonomous and self-organizing manner. Results of preliminary experiments support our proposal, but there still remain room for further evaluation and modification. First we need to consider more realistic scenario where communication arises in not only many-to-one but also one-to-one, one-to-many, and many-to-many fashion in a service network. We also plan to improve the mechanism for service networks to share relay nodes. Furthermore, we consider that the response threshold model can solve feature interaction among services competing for the same actuator to some extent. Evaluation from a viewpoint of robustness, scalability, and adaptability is one of future work as well.

Acknowledgement

This research was supported in part by “Global COE (Centers of Excellence) Program” and Grant-in-Aid for Scientific Research (B) 22300023 of the Ministry of Education, Culture, Sports, Science and Technology, Japan and “Early-concept Grants for Exploratory Research on New-generation Network” of the National Institute of Information and Communications Technology, Japan.

References

[1] M. Murata, Towards establishing ambient network environment, IEICE Transactions on Communications 92 (4) (2009) 1070–1076.
[2] E. Bonabeau, A. Sobkowski, G. Theraulaz, J. L. Deneubourg, Adaptive task allocation inspired by a model of division of labor in social insects, in: Proceedings of Biocomputing and Emergent Computation, 1997, pp. 36–45.
[3] E. Avilés-López, J. García-Macías, TinySOA: a service-oriented architecture for wireless sensor networks, Service Oriented Computing and Applications 3 (2) (2009) 99–108.
[4] E. H. Jung, Y. J. Park, TinyONet: A cache-based sensor network bridge enabling sensing data reusability and customized wireless sensor network services, Sensors 8 (12) (2008) 7930–7950.
[5] H. M. N. D. Bandara, A. P. Jayasumana, T. H. Illangasekare, Cluster tree based self organization of virtual sensor networks, in: Proceedings of the International workshops on Wireless Mesh and Sensor Networks, 2008, pp. 1–6.
[6] C. Frank, K. Romer, Algorithms for generic role assignment in wireless sensor networks, in: Proceedings of the International conference on Embedded Networked Sensor Systems, 2005, pp. 230–242.
[7] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: A scalable and robust communication paradigm for sensor networks, in: Proceedings of the International conference on Mobile Computing and Networking, 2000, pp. 56–67.
[8] A. Varga, et al., The OMNeT++ discrete event simulation system, in: Proceedings of the European Simulation Multiconference, 2001, pp. 319–324.
[9] Crossbow Technology, MICAz Datasheet, http://www.xbow.com.