International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)
ISSN: 2150-7988 Vol.1 (2009), pp.125-132
http://www.mirlabs.org/ijcisim

Autonomic Wireless Sensor/Actuator Networks for Tracking Environment Control Behaviors

Masayuki Nakamura, Atsushi Sakurai and Jiro Nakamura
NTT Energy and Environment Systems Laboratories
Nippon Telegraph and Telephone Corporation
Morinosato Wakamiya, Atsugi-shi, Kanagawa Pref., 243-0198 Japan
E-mail: {masayuki, a2c, jnaka}@aecl.ntt.co.jp

Abstract

To develop energy-saving environment control systems, we propose autonomic wireless sensor/actuator networks that classify a user's behaviors in relation to environment control such as lighting, and that configure themselves depending on sensor node selection. In our system, a wireless remote control node monitors a user's actions with respect to environment control, and occupancy sensor networks simultaneously detect the user's movement. The system learns the relationship between the responses of the remote control node and the occupancy sensor networks to classify the user's behaviors with only the occupancy sensor networks. The system chooses informative sensor nodes for this behavior classification based on an information gain criterion. These chosen nodes have a high sensing cost. Sensor network routing is controlled based on the sensing cost and communication cost metric. In the resultant sensor networks, the sensing performance is the same as that in the original network, but the resources are successfully allocated to the nodes. In addition, less informative and redundant nodes are identified. We demonstrate tracking environment control behaviors and sensor node selection using the sensor/actuator networks testbed.

1. Introduction

The recognition of daily activities is a key technology when developing home automation and smart home systems. Such systems can detect that an occupant is in a critical condition, or control devices automatically depending on the occupant's behaviors. Activity recognition plays a particularly important role in energy-saving environment control, and one of the main topics is determining how to construct a reliable sensing system.

Mozer developed a neural network house that is embedded with numerous wired sensors for home automation [1]. His system monitors the residents' behaviors, forecasts future movement and controls the environment to satisfy both energy saving and user comfort using reinforcement learning. Wireless sensor networks have been used to recognize the activities of daily living. Tapia *et al.* developed a wireless sensor system that recognizes daily activities in a home [2]. They used simple state-change sensors that can be easily retrofitted to an existing home. To recognize more complicated activities, wireless camera sensors and RFIDs have been used in sensor networks. Lymberopoulos *et al.* developed camera sensor networks for inferring behaviors [3]. Patterson *et al.* developed an activity recognition system using an RFID glove [4]. Wireless sensor/actuator networks are emerging technologies in environment control applications. Nakamura *et al.* developed wireless sensor/actuator networks to realize optimal lighting control [5].

Although sensor networks appear promising in terms of activity recognition, there are certain problems as regards their applications. Sensor networks have strong resource constraints, and it is essential to construct efficient networks. Cayirci *et al.* proposed wireless sensor networks for use in an underwater surveillance system, which classify various objects in the water [6]. They used radiation, mechanical, acoustic, and magnetic sensors for target detection, and a decision tree technique for target classification. In addition, they proposed an underwater sensor network coverage scheme designed to maximize the sensing space.

Yang *et al.* introduced a concept of autonomic sensing for a pervasive sensing system [7]. According to their definition, an autonomic sensing system has certain *self-** properties such as self-configuration, self-organization and self-adaptation. We believe that sensor networks have to configure their own structures to optimize their sensing performance. There are useful nodes and less informative nodes in sensor networks. Sensor networks should use such information to select the nodes and configure the network structures so that the resources are successfully allocated among the nodes.

In this paper we propose sensor/actuator networks adaptive to the user's behaviors for environment control (Figure 1). The sensor/actuator networks choose informative nodes for sensing the user's behaviors and reconfigure their network structures. Visualization of the sensor node deployment and the user's behaviors, sensor node selection, and classification are centralized procedures performed at the server. The sensor network is reconfigured with a decentralized approach using the sensor node selection result. First, we explain the information flow with respect to sensor node selection and the sensor network configuration based on information gain. Next, we illustrate the visualization of the node deployment and the user's behaviors using the Self-Organizing Map (SOM) introduced by Kohonen [8]. Then, we describe our sensor/actuator networks testbed consisting of a remote control node and occupancy sensor nodes. In addition to analyzing the relationship between the responses of nodes, we classify behavior using a decision tree algorithm. We demonstrate sensor node selection and routing control based on the selection.

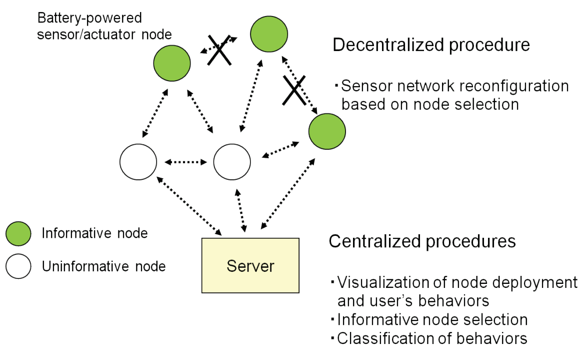

Figure 1 Proposed sensor/actuator networks

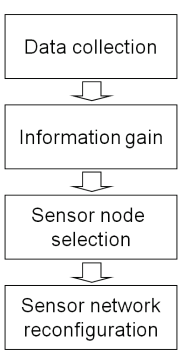

Figure 2 Information flow in proposed sensor/actuator networks

2. Sensor network reconfiguration based on sensor node selection

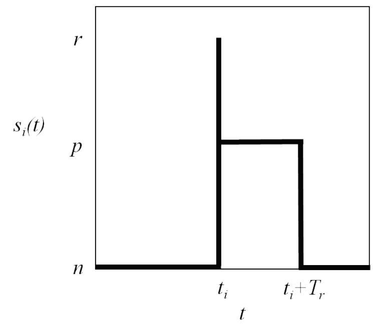

Figure 3 Profile of $s_i(t)$

Figure 2 shows the information flow in the proposed sensor networks. The server collects data from sensor nodes and classifies the data using the decision tree. In the decision tree, the information gain is calculated to enable us to choose the informative sensor nodes. The sensor networks are reconfigured based on the chosen sensor nodes.

First, the feature vector for behavior classification is defined as follows:

$$\{s_1(t), s_2(t), \cdots s_k(t)\},$$

$$s_i(t) = g(t - t_i),$$

$$g(t) = \begin{cases} r, & t = 0 \\ p, & 0 < t < T_r, \\ n, & t \geq T_r \end{cases}$$

where k is the number of nodes, t_i is the firing time of node i, T_r is the retention time, and r, p, and n are nominal values (Figure 3). When a sensor node fires, the server receives the signal and calculates the feature vector of $\{s_1(t), s_2(t), \cdots s_k(t)\}$. Each element in the vector has three values; r when the sensor node fires, p within T_r of firing, and n otherwise.

Now we consider the classification of objects characterized by the feature vector. Let $I(C)$ be expected information defined as

$$I(C) = -\sum_x p_x(C) \log p_x(C),$$

where C is a collection of objects, $p_x(C)$ is the probability of x in C, and $I(C)$ is the expected information needed to classify objects in C [9]. S is defined as a sensor node with values $\{S_1, S_2, \cdots S_v\}$. If sensor S is used for the root of the decision tree, C is partitioned into $\{C_1, C_2, \cdots C_v\}$. The information gain realized by branching on S is calculated as follows:

$$gain(S) = I(C) - \sum_{i=1}^{v} \frac{|C_i|}{|C|} I(C_i).$$

ID3 examines all the sensors and chooses one to maximize the information gain. The same sensor selection is executed in the subtrees.

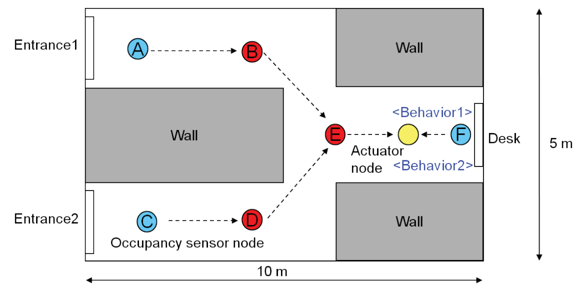

Figure 4 Sensor node deployment in workplace. Dotted arrows indicate RF communications.

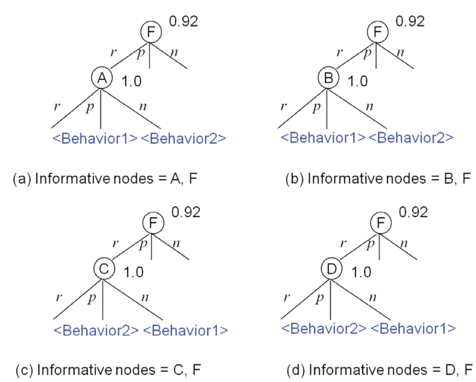

Figure 5 Decision trees for behavior classification. Numerical values indicate information gains.

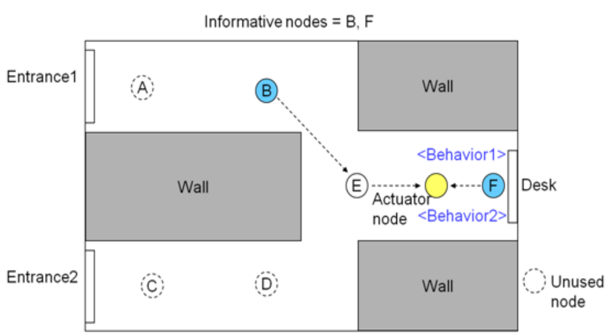

Figure 6 Sensor network reconfiguration with sensornode selection. Dotted arrows indicate RF communications.

The sensing costs of chosen sensor nodes are high because these informative nodes have to be active frequently. In sensor networks, it is essential to allocate resources to nodes. One of the resource allocation techniques involves the use of a sensing cost and communication cost metric [5]. The parent node is chosen so that the total sensing and communication costs are minimized,

$$K = K_{sens} + K_{comm},$$

$$K_{sens} = \begin{cases} \alpha & \text{if chosen sensor} \\ 0 & \text{otherwise} \end{cases},$$

$$p = \arg\min_{j \in N_i} K_j,$$

where K_{sens} is the sensing cost, K_{comm} is the communication cost, K is the total cost, α is a positive number, p is a parent, N_i is the communication range of node i, and K_j is the total cost of neighboring node j. This is a decentralized algorithm performed by each node. In this algorithm, less informative sensor nodes tend to become parent nodes while informative nodes concentrate on sensing tasks.

For instance, consider a workplace where the ceiling is embedded with wireless occupancy sensor nodes as shown in Figure 4. The node's wireless communication range is too short for the detection data to be transmitted to the actuator node (yellow node) directly. Therefore, multihop transmission is necessary in this sensor network. The red nodes in this figure consume more resources than the blue nodes because the red nodes perform sensing and relay tasks as parent nodes whereas the blue nodes perform sensing only.

Now we consider classifying the user's behaviors, for example behavior1: enter the room from entrance1 and sit at the desk, and behavior2: enter the room from entrance2 and sit at the desk. The actuator node controls lights or air-conditioning depending on the behaviors. It is assumed that the user's walking speed is 0.5 m/s and T_r is sufficiently long, for example 100 s, to allow all movements in the room to be tracked. Then the feature vectors $\{s_A(t), s_B(t), s_C(t), s_D(t), s_E(t), s_F(t)\}$ for behavior1 and behavior2 are represented as

Behavior1: $\{p, p, n, n, p, r\}$,

Behavior2: $\{n, n, p, p, p, r\}$.

Another example might simply be passing D after entering the room from entrance2 and this is represented as

$\{n, n, p, r, n, n\}$.

Figure 5 shows the decision trees we employed for behavior classification. We used 18 behavior patterns and the WEKA package for the ID3 decision tree algorithm [10]. First, node F is chosen as an informative node because its information gain is the largest. Next, nodes A, B, C, or D are chosen because their information gain is the largest in the subtrees. These results indicate that only two sensor nodes are necessary for behavior classification. In particular, the sensor selection in Figure

5(b) and (d) realizes sensor networks using the fewest possible sensor nodes. Figure 6 shows the sensor network that corresponds to Figure 5(b). In this sensor network, nodes F and B are informative sensor nodes, and node E is a parent of node B. Thus nodes F and B work as sensing nodes, and node E works only as a parent and does not perform sensing tasks. This result indicates that the resource allocation among the nodes is successful.

3. Visualization of node deployment and user's behaviors using SOM

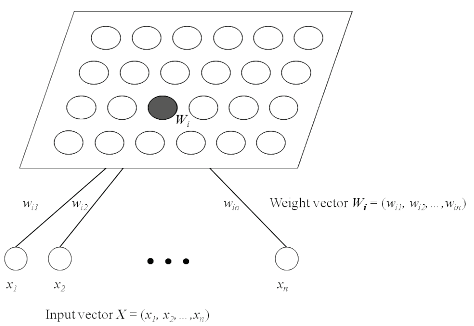

Figure 7 Basic structure of standard SOM

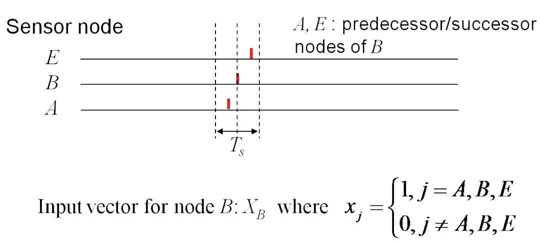

Figure 8 Formation of SOM input vector

A SOM, which is a kind of neural network algorithm, has been successfully used to visualize node deployment without the need for explicit localization devices [11]. In this paper we use a SOM to visualize both the user's behaviors and node deployment.

The SOM involves mapping from input data to a two-dimensional array of units (Figure 7). The SOM can capture the distinctive features of the input data since it is a nonlinear projection of the input data. Therefore the SOM is also called a feature map. The lattice type array that we use can have a rectangular or hexagonal topology structure. The input pattern vector is denoted as $\boldsymbol{X}$, and the weight of unit i as $\boldsymbol{W}_i$. Vector $\boldsymbol{X}$ may be compared with all $\boldsymbol{W}_i$ in any metric. In many practical applications, the smallest Euclidean distance $\|\boldsymbol{X} - \boldsymbol{W}_i\|$ is usually used when defining the best-matching unit c such that

$$\|\boldsymbol{X} - \boldsymbol{W}_c\| = \min_i \|\boldsymbol{X} - \boldsymbol{W}_i\|.$$

The task is to define $\boldsymbol{W}_i$ in such a way that the mapping is ordered and descriptive of the vector distribution $\boldsymbol{X}$. As a result, a set of $\boldsymbol{W}_i$ values is obtained as the convergence limit of the following sequence;

$$\boldsymbol{W}_i(t+1) = \boldsymbol{W}_i(t) + h_{ci}(t)\left[\boldsymbol{X}(t) - \boldsymbol{W}_i(t)\right] \quad \text{for } i \in N_c,$$

where $h_{ci}(t)$ is the neighborhood kernel, which is a function defined over the lattice points, and N_c is the neighborhood of c. In this paper we define the neighborhood kernel as a step function;

$$h_{ci}(t) = \begin{cases} \beta(t) & \text{for } i \in N_c \\ 0 & \text{for } i \notin N_c \end{cases},$$

where the learning rate $\beta(t)$ is a monotonically decreasing function of time ($0 < \beta(t) < 1$).

To apply the SOM to the visualization of the node deployment and the user's behaviors, an input vector should be defined on the basis of sensor detection. The user's behaviors are described by the remote control signals to the actuator node. For instance, in Figure 4, node (or behavior) i's input vector into SOM, X_i, is represented as

$$X_i = \{x_A, x_B, \cdots, x_F, x_{Behavior1}, x_{Behavior2}\},$$

$$x_j = \begin{cases} 1, j \in T_s \\ 0, j \notin T_s \end{cases},$$

where T_s is the analysis interval. When node i fires at time t_i, nodes that fire between $t_i - T_s/2$ and $t_i + T_s/2$ provide a value of 1, and 0 otherwise, in the input vector (Figure 8).

After the input data have been incorporated in the SOM, self-organization results in a topographic map. On the map, similar data points are likely to be projected to neighboring units. The map can be used to show the spatial relationships between data with respect to the nodes; hence, different grid structures can reveal different relationships.

4. Experimental

Figure 9 shows the experimental setup. We used Crossbow Motes, MICA2DOTs for the sensor network testbed [12]. A Mote consists of an Atmega128 microcontroller (4 MHz), a CC1000 radio (315 MHz), and a battery. In this experiment, the Mote's wireless communication range was limited to approximately 5 m. The actuator node consists of a Mote and an IR LED, which controls the lights. The remote control node has buttons that monitor the user's control actions. When the user pushes the buttons, the remote control node fires.

The actuator node receives the signal and controls a light. The occupancy sensor nodes are equipped with PIR occupancy sensors. The occupancy sensor's sampling interval is 0.5 s. They fire and transmit the signals to the actuator node when they detect the user's movement. A base station receives the signals from the actuator node and a PC stores and analyzes the data.

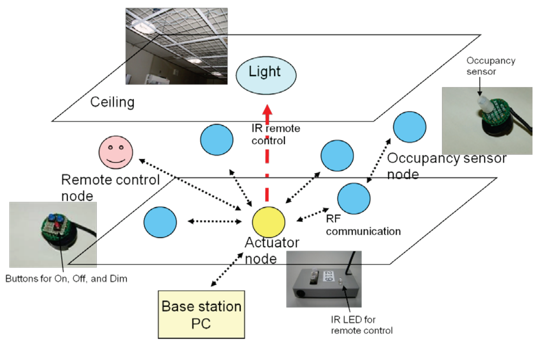

Figure 9 Experimental setup for tracking environment control behaviors

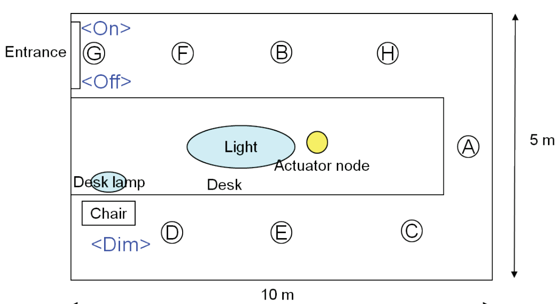

Figure 10 Sensor node deployment in workplace

Figure 10 shows the deployment of the sensor nodes (A-H) in a workplace. The occupancy sensor nodes were arranged on the ceiling (3 m high) to track the user's movement. Their sensing areas did not overlap. The actuator node was on the desk and an IR controllable light was installed on the ceiling immediately above it. The user with a remote control node entered the room, turned on the light, walked to the chair, sat on the chair, dimmed the light, got up from the chair, walked to the door, turned off the light, and left the room.

First, we analyzed the relationship between the responses of the occupancy sensor nodes and the remote control node using the SOM. The input vector into the SOM, X_i, is represented as

$$X_i = \{x_A, x_B, \cdots, x_H, x_{On}, x_{Off}, x_{Dim}\}.$$

We used SOM_PAK software for the SOM analysis [13, 14].

Next, we classified the responses of the occupancy sensors for On, Off, and Dim behaviors. In addition, informative sensor nodes were chosen based on the information gain criterion. After the sensor selection, the occupancy sensor networks were reconfigured based on the sensing and communication cost metric. This metric-based routing was programmed using TinyOS [15].

5. Results

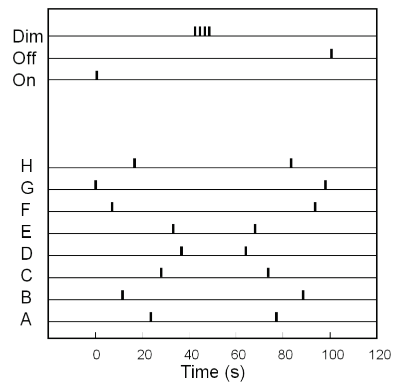

Figure 11 Sensor nodes responses for environment control behaviors

Figure 11 shows the sensor node responses for lighting control behaviors. It is shown that the sensor nodes track the user's behaviors. First, we analyzed these responses using the SOM. The input vectors into the SOM were composed using the responses shown in Figure 11. For instance, node A's vectors were incorporated into X_A:

$$X_A = \{1,0,1,0,0,0,0,1,0,0,0\},$$

with $T_s = 15$ s (Figure 12).

The calculated SOM is shown in Figure 13. The SOM parameters were as follows: number of units: 10 × 10, initial learning rate: 0.2, radius of N_c: 10, topology type: hexagonal, and number of learning steps: 20000. In Figure 13, the gray level corresponds to the distance between the units. As the distance becomes greater, the unit becomes darker. This figure shows that On and Off behaviors occurred close to node G, and Dim behavior occurred close to node D. The cell arrangement displays the behavior sequence. The user entered the room, passed G, turned on the light, passed F, B, H, A, C, E, D, sat on the chair, dimmed the light, got up from the chair, passed D, E, C, A, H, B, F, G, and turned off the light. These behaviors produced the map. This result shows that the

SOM visualized the relationship between the responses of the nodes.

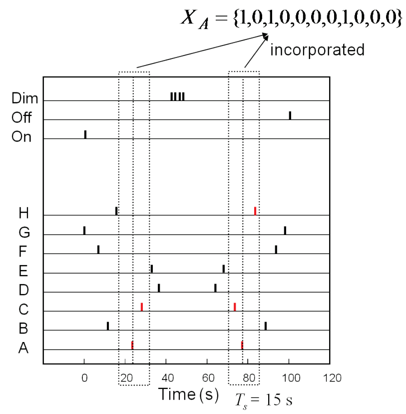

Figure 12 SOM input vector for node A

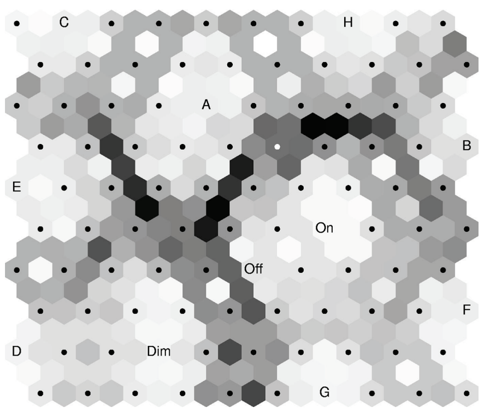

Figure 13 SOM analysis of environment control behaviors

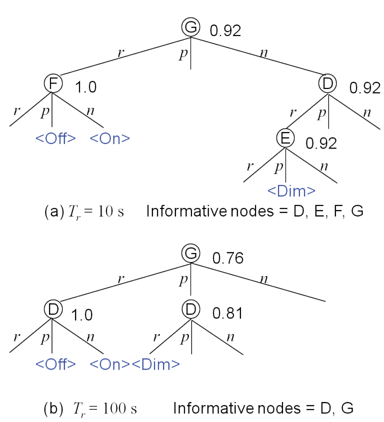

Figure 14 Decision trees for behavior classification

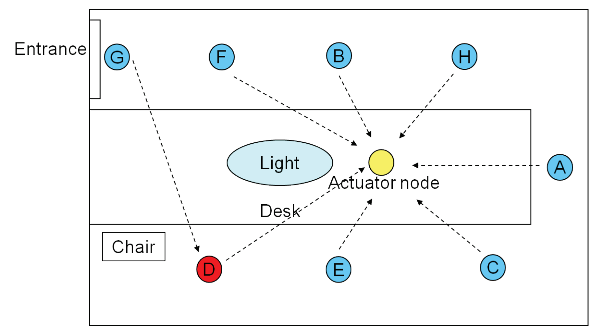

Figure 15 Sensor network reconfiguration without sensor node selection. Dotted arrows indicate RF communications.

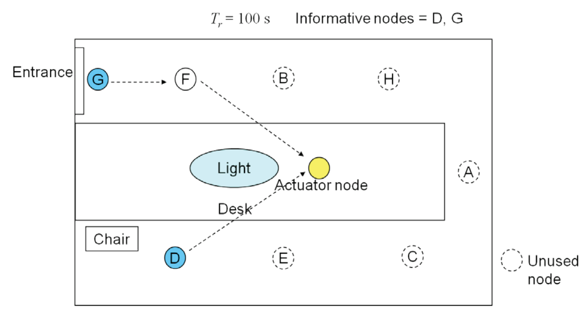

Figure 16 Sensor network reconfiguration with sensor node selection. Dotted arrows indicate RF communications.

Next, we evaluated the sensor selection and network reconfiguration. Each behavior was characterized solely by a feature vector of the occupancy sensor outputs $\{s_A(t), s_B(t), s_C(t), s_D(t), s_E(t), s_F(t), s_G(t), s_H(t)\}$. For instance, the feature vector for passing node C after entering the room was represented as

$$\{p, p, r, n, n, p, p, p\},$$

with T_r = 50 s. On, Off, and Dim behaviors were characterized by the preceding node passing behavior. In this experiment, there was a difference between On behavior and Off behavior when passing node G and a difference between Dim behavior and simply passing when passing node D. These behaviors had to be distinguished.

Figure 14 shows the decision trees of (a) T_r = 10 s and (b) T_r = 100 s. We used a total of 16 behavior patterns including On, Off, and Dim. In Figure 14(a), the retention time was short and four sensor nodes were used. The responses of sensor nodes F and E provided information about the movement direction. On the other hand, only two sensor nodes were used in Figure 14(b). The retention time was long and node D provided information about the movement direction when passing node G. These results indicate that the sensor node selection based on the information gain was successful, and the unselected sensor nodes were not useful for classification. In both cases, there were no misclassifications of the 16 training data.

Figure 15 shows an example of a network configuration without sensor node selection when T_r = 100 s. There were no misclassifications of the 16 training data using the neural net classifier. The parameters in the neural net were as follows: network dimensions: 24 input units, 14 hidden units, and 4 output units, learning rate: 0.3, momentum: 0.2, and training number: 500. All sensor nodes were used for sensing and node D was also chosen as a parent by the communication cost metric. Node D clearly consumed more resources than the other nodes. This result indicates that unnecessary sensor nodes were used, and resources were not allocated uniformly.

Figure 16 shows an example of a network configuration with sensor node selection when T_r = 100 s. As described in Figure 14, there were no misclassifications of the 16 training data. Only nodes G and D were used as sensing nodes. Another sensor node such as node F was used as a parent. No redundant sensor nodes were used. This result was obtained by using the communication and sensing cost metric when choosing a parent.

6. Discussion

The actuator node has an IR LED for controlling a light. It can also communicate with IR controllable devices such as a TV, an air-conditioning unit, and a blind. This testbed will be applicable to various automatic energy-saving controls.

In the experiment, the remote control node annotates the response sequence of the occupancy sensor nodes as well as controlling a light. It is used only in the behavior learning procedure. After learning, occupancy sensor networks can recognize the user's behaviors without the remote control node. The actuator node controls the light based on activity recognition.

An SOM is a useful tool for visualizing sensor node responses or sensor node deployment. In this paper, we used binary input vectors for the SOM. It is possible to obtain a more detailed SOM if we use the temporal information provided by the sensors for the input vector. In sensor networks, it is a time-consuming task to manage and localize sensor nodes. An SOM can visualize the relationships between the user's activities and the node deployment without any distance-based localization techniques.

In this paper we used small feature vectors, namely eight-dimensional vectors whose elements had three values. The feature vectors become larger as the sensor networks become larger. However, sensor selection based on information gain requires only small computational power because of the greedy top-down search procedure. We will be able to deal with larger sensor networks for sensor selection. This remains to be investigated along with routing problems in larger sensor networks.

The way in which we determine the retention time T_r of the feature vector in sensor selection is important. It depends on how long the targeted behaviors last. On, Off, and Dim behaviors are simple and short. Therefore, T_r should be short as in Figure 14 (a). T_r in Figure 14 (b), however, is somewhat difficult. Only two sensor nodes were used because only three behaviors were targeted in this experiment. When more behaviors are targeted, we have to use a shorter T_r.

The feature vector was not based on the temporal data of the sensor nodes. The temporal data might involve abundant information about behaviors. In particular, the classification of complicated behavior composed of simple behaviors requires the use of temporal data. In this paper, we dealt only with simple behaviors. The transformation of the temporal data into feature vectors also remains to be investigated.

Once sensor selection has been performed at the server, the selection result is sent to each node and the network is reconfigured autonomously. Visualization, sensor selection, and classification are centralized procedures that are performed using global information. Network reconfiguration, on the other hand, is a decentralized procedure that is performed locally. In this way the proposed sensor/actuator networks include both centralized and decentralized procedures.

If we focus only on behavior classification, we analyze the data from all the nodes without sensor selection. For instance, a neural net, which is a powerful classifier, uses all of the data without considering which sensor is informative. As a result, redundant sensor nodes are used and, to make matters worse, resource allocation might fail as indicated in Figure 15. We believe that the proposed sensor/actuator networks can optimize both classification performance and network configuration. As shown in Figure 16, sensor selection minimizes node usage and results in successful resource allocation.

7. Conclusion

We proposed autonomic sensor/actuator networks that have self-configuration, self-organization, and self-adaptation properties. They involve centralized procedures for the visualization of node deployment and user's behaviors, informative node selection and behavior classification, and a decentralized procedure for sensor network reconfiguration. We demonstrated tracking environment control behaviors and sensor node selection using a sensor network testbed. First, we used an SOM to visualize the relationship between the responses of nodes. The SOM clarified the sequence of the user's behaviors. Next, we evaluated the sensor node selection and network reconfiguration. Based on the information gain criterion, the informative sensor nodes for behavior classification were successfully chosen without sacrificing the classification performance. The selection information was transmitted to each node and the sensor network was reconfigured using a sensing and communication cost metric. In the resultant networks, it was shown that the redundant sensor nodes were identified for use as parent nodes and the resources were successfully allocated to these nodes. In addition, the system classified the user's behaviors successfully simply by using the occupancy sensor networks after the behavior learning procedure. The proposed sensor/actuator networks will be useful for constructing an automatic energy-saving environment control system adaptive to the user's behaviors.

References

[1] Mozer, M.C., 2005, "Lessons from an Adaptive Home", In: Cook, D.J. and Das, S.K. (ed), Smart environments: Technologies, protocols, and applications, John Wiley & Sons, New Jersey, pp. 273-294.
[2] Tapia, E.M., Intille, S.S., and Larson, K., 2004, "Activity Recognition in the Home Setting Using Simple and Ubiquitous Sensors", Pervasive 2004, pp. 158-175.
[3] Lymberopoulos, D., Ogale, A.S., Savvides, A., and Aloimonos, Y., 2006, "A Sensory Grammar for Inferring Behaviors in Sensor Networks", IPSN2006, pp. 251-259.
[4] Patterson, D.J., Fox, D., Kautz, H., and Philipose, M., 2005, "Fine-Grained Activity Recognition by Aggregating Abstract Object Usage", ISWC '05, pp. 44-51.
[5] Nakamura, M., Sakurai, A., Furubo, S., and Ban, H., 2008, "Collaborative Processing in Mote-based Sensor/actuator Networks for Environment Control Application", Signal Processing, 88(7), pp. 1827-1838.
[6] Cayirci, E., Tezcan, H., Dogan, Y., and Coskun, V., 2006, "Wireless Sensor Networks for Underwater Surveillance System", Ad Hoc Networks, 4, pp. 431-446.
[7] Yang, G.-Z., Lo, B., and Thiemjarus, S., 2006, "Autonomic sensing", In: Yang, G.-Z. (ed.), Body Sensor Networks, Springer-Verlag, London, pp. 333-372.
[8] Kohonen, T., 1990, "The Self-Organizing Map", Proceedings of the IEEE, 78, 9, pp. 1464-1480.
[9] Quinlan, J.R., 1986, "Induction of Decision Trees", Machine Learning, 1, pp. 81-106.
[10] WEKA, http://www.cs.waikato.ac.nz/ml/weka/
[11] Sakurai, A., Nakamura, M. Furubo, S., and Ban, H., 2005, "Sensor Localization Using Self-Organizing Map for Human Tracking", ICST 2005, pp. 311-315.
[12] Crossbow Technology, http://www.xbow.com/
[13] Kohonen, T., Hynninen, J. Kangas, J., and Laaksonen, J., 1996, "SOM_PAK: The Self-Organizing Map Program Package", Technical Report A31, Laboratory of Computer & Information Science, Helsinki University of Technology, January.
[14] SOM_PAK, http://www.cis.hut.fi/research/som-research/nnrc-programs.shtml
[15] TinyOS, http://www.tinyos.net/

Biographies

Masayuki Nakamura received B.E., M.E., and Ph.D. degrees in mathematical engineering and information physics from the University of Tokyo, Tokyo in 1988, 1990, and 1998. He joined NTT in 1990 and has studied information processing in sensor networks and sensing systems. He is currently a Senior Research Engineer, Environmental Information System Project, NTT Energy and Environment Systems Laboratories. Dr. Nakamura is a member of the Society of Instrument and Control Engineers and the Institute of Electrical Engineers of Japan (IEEJ). He received a presentation award from IEEJ in 1998.

Atsushi Sakurai received a bachelor's degree in Policy Management and a master's degree in Media and Governance from Keio University, Kanagawa, Japan, in 2001 and 2003, respectively. His major field of study was Bioinformatics. He joined NTT in 2003 and has been a researcher in the NTT Environmental Information System Project. His current research interest is sensor networks in home and office environments. Mr. Sakurai is a member of the Information Processing Society of Japan. He received the SFC award from Keio University in 2002.

Jiro Nakamura is a Group Leader of the Environmental Information Systems Project, NTT Energy and Environment Systems Laboratories. He received B.E., M.E. and Ph.D. degrees in applied chemistry from Osaka University, Suita, Osaka, in 1987, 1989 and 1995, respectively. In 1989, he joined NTT LSI Laboratories, Atsugi, Japan, where he worked on the development of microfabrication technology. In 2001, he moved to the NTT Information Sharing Laboratory Group. Since moving to his present research department in 2004, he has been engaged in the development of environmental sensing systems and in analyzing the influence of ICT on the global environment. He received the MicroProcess Conference Award in 1997 and the Photopolymer Conference Award in 1998.