Bio-inspired Self-organizing Control Mechanisms for Cooperative Wireless Ad-hoc and Sensor Networks

January 2013

Hiroshi YAMAMOTO

Bio-inspired Self-organizing Control Mechanisms for Cooperative Wireless Ad-hoc and Sensor Networks

Submitted to
Graduate School of Information Science and Technology
Osaka University

January 2013

Hiroshi YAMAMOTO

List of Publications

Journal Papers

1. Hiroshi Yamamoto, Katsuya Hyodo, Naoki Wakamiya, and Masayuki Murata, "A reaction-diffusion based coding rate control mechanism for camera sensor networks", *Sensors*, Vol. 10, No. 8, pp. 7651–7673, August 2010.

2. Hiroshi Yamamoto, Naoki Wakamiya, and Masayuki Murata, "An inter-networking mechanism with stepwise synchronization for wireless sensor networks," *Sensors*, Vol. 11, No. 9, pp. 8241–8260, August 2011.

Refereed Conference Papers

1. Hiroshi Yamamoto, Katsuya Hyodo, Naoki Wakamiya, and Masayuki Murata, "Implementation and evaluation of a reaction-diffusion based coding rate control mechanism for camera sensor networks," In *Proceedings of 2nd ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC)*, California, USA, September 2008.

2. Hiroshi Yamamoto, Naoki Wakamiya, and Masayuki Murata, "An Inter-networking Mechanism using Stepwise Synchronization for Wireless Sensor Networks," In *Proceedings of the 1st Workshop on Bio-inspired Models and Technologies for Ambient Information Society (BioAmbIS)*, pp. 17–21, Boston, USA, December 2010.

Non-Refereed Conference Paper

1. Hiroshi Yamamoto, Naoki Wakamiya, and Masayuki Murata, “Study on interaction between layered self-organization based control,” In *Proceedings of 4th international symposium on applied sciences in biomedical and communication technologies (ISABEL)*, Barcelona, Catalonia, Spain, October 2011.

Non-Refereed Technical Papers

1. Hiroshi Yamamoto, Katsuya Hyodo, Naoki Wakamiya, and Masayuki Murata, “Implementation and evaluation of a reaction-diffusion based coding rate control mechanism for camera sensor networks,” *Technical Report of IEICE (IN2008-6)*, Vol. 108, No. 60, pp. 31–36, May 2008. (in Japanese)

2. Hiroshi Yamamoto, Naoki Wakamiya, and Masayuki Murata, “Proposal and evaluation of a stepwise synchronization-based inter-networking mechanism for wireless sensor networks,” *Technical Report of IEICE (USN2010-4)*, Vol. 110, No. 50, pp. 17–20, May 2010. (in Japanese)

3. Hiroshi Yamamoto, Naoki Wakamiya, and Masayuki Murata, “Proposal and evaluation of an inter-networking mechanism using stepwise synchronization for wireless sensor networks,” *Technical Report of IEICE (USN2010-40)*, Vol. 110, No. 378, pp. 13–18, January 2011. (in Japanese)

Preface

Recently, billions of devices, such as PCs, mobile phones and sensor nodes are connected to the Internet, to make our life safe, secure, and confortable. The scale of the Internet continuously increases in the number, heterogeneity, and mobility of devices and many new applications are emerging with help of proliferation of networked devices. To deal with such large and complex networks, self-organizing control mechanisms, where each node autonomously decides its behavior based on local information it observes or obtains from neighbors and the global control emerges through mutual interaction among neighboring nodes, have gathered much attention. Although self-organizing phenomena are recognized in several research fields, those researches inspired by self-organizing behavior of biological systems are most active and promising. Because biological systems are inherently autonomous and self-organizing, where there is no centralized control unit dominating the whole, and they often exhibit scalable, adaptive, and robust properties, it is expected that network control mechanisms can achieve the scalability, adaptability, and robustness by being inspired by such biological systems. Furthermore, since application's requirements on networks, e.g. type of devices to accommodate, required bandwidth, acceptable delay and packet loss, and operational policy, differ with each application, networks must provide various kinds of functionalities to a variety of applications. Because it is apparent that a monolithic network adopting a single self-organizing control mechanism cannot satisfy all requirements, we need to consider coordinating multiple networks adopting multiple self-organizing control mechanisms in a cooperative manner. Although there are many successful self-organizing control mechanisms, whose superiority to conventional mechanisms was verified through simulation and practical experiments, cooperation among multiple self-organizing control mechanisms is not well investigated.

In this thesis, we study self-organizing control and cooperation mechanisms in wireless ad-hoc and sensor networks. We begin this thesis by proposing, implementing, and evaluating a self-organizing video coding rate control mechanism for wireless camera networks, where nodes cooperate with each other to adjust video coding rate not to exceed the network capacity while obtaining high quality images of targets. In camera sensor networks, which is composed of sensor nodes equipped with a camera, each node captures a video image around itself, and sends the video image toward a base station. Compared to typical scalar and even analog sensor data such as humidity, temperature, and acceleration, the volume of video data generated by cameras is considerably large. Since the capacity of wireless sensor networks is limited to accommodate such huge video traffic, wireless camera networks cannot accommodate huge traffic such as video images generated by cameras even if they adopt well-known techniques on QoS control for video transmission in wireless networks. In surveillance and monitoring applications, users are interested in video images that capture targets or certain phenomena. Therefore, camera sensor nodes should adjust the video coding rate in accordance with the location and velocity of targets. In our proposal, by adopting the reaction-diffusion model, we generate spot patterns of concentrations of virtual morphogens, so that the morphogen concentrations have a peak at a camera detecting a target in its observation area, and then map the morphogen concentration to the video coding ratio. In addition, to prepare cameras in the direction of a moving target for future detection, their video coding rates, in other words, their morphogen concentrations are made slightly higher than others. Furthermore, we also propose a mechanism to keep the total of coding rate low for closely located targets considering the limited capacity of a wireless network. Through simulation and practical experiments, we verify that our proposal can avoid network congestion and keep the quality of video image from a camera having the target high.

Next, we propose a mechanism for multiple sensor networks to cooperate and be integrated for smooth communication among networks to offer a unified service to users. In general, wireless sensor networks adopt a sleep scheduling mechanism to save energy. Operational frequencies, that is frequencies that they wake up and resume operation, are different among networks depending on application's requirement and characteristics of devices. When wireless sensor networks, whose

operational frequencies are largely different, want to exchange information among them for coordinated actions, a node belonging to the faster network has to stay active in order to wait for a node belonging to the slower network to wake up in transmitting a message. For smooth and moderate inter-networking, we propose stepwise synchronization between wireless sensor networks, where sensor nodes located near the border of two networks adjust their operational frequencies to bridge the gap in their intrinsic operational frequencies. In our proposal, we adopt the pulse-coupled oscillator model and strengthen the degree of entrainment at border nodes to intensively shift the operational frequency toward that of the other network while the degree of entrainment is weakened as the distance to the border increases. As a result, the operational frequencies of nodes near the border are adjusted to somewhere between the original operational frequencies of wireless sensor networks. Through simulation experiments, it is shown that the delay in communication between border nodes was reduced, and the energy consumption was balanced among nodes, which leads to prolongation of the lifetime of wireless sensor networks.

Finally, we further investigate the influence of interaction or interdependency between layered self-organizing mechanisms on performance and stability of the whole network system. Although there are many proposals on self-organization based protocols that are useful, effective, and practical, there has never been any in-depth investigation into interaction, interference, and synergetic effects among multiple self-organization based control. Then, we study mutual interaction among layered self-organization based control. We consider an ad-hoc network and an overlay network that is constructed over the ad-hoc network, both of which adopt adaptive routing protocols based on the attractor selection model. We modify the degree of coupling by changing the way how layered self-organization control shared an objective parameter. Through simulation experiments, we show that the upper layer-aware coupling can provide the best performance at the sacrifice of the stability such as variation of paths.

Acknowledgments

First of all, I would like to express my sincere appreciation to my supervisor, Professor Masayuki Murata of Graduate School of Information Science and Technology, Osaka University, for his patient encouragement, meaningful advices, and valuable discussions. His creative suggestions, insightful comments, and precise guidance have been essential for my research activity.

I heartily grateful to the members of my thesis committee, Professor Koso Murakami, and Professor Teruo Higashino, Graduate School of Information Science and Technology, Osaka University, and Professor Hirotaka Nakano of Cybermedia Center, Osaka University, for their valuable comments, and reviewing of this thesis.

I especially would like to express my appreciation to Professor Naoki Wakamiya of Graduate School of Information Science and Technology, Osaka University for his appropriate and elaborated guidance. My study would not have been possible without his continuous support and encouragement.

I would like to thank to Associate Professor Go Hasegawa, Associate Professor Shin'ichi Arakawa, Assistant Professor Yuichi Ohshita, Assistant Professor Yoshiaki Taniguchi and Assistant Professor Yuki Koizumi of Osaka University, Dr. Kenji Leibnitz of National Institute of Information and Communications Technology, Assistant Professor Shinsuke Kajioka of Nagoya Institute of Technology, and Mr. Katsuya Hyodo, who gave me helpful comments and feedback. Their critical comments and useful suggestions have been always informative and helpful.

I am thankful to all the members of Advanced Network Architecture Laboratory in Graduate School of Information Science and Technology, Osaka University, for their inciting discussions and fellowship.

Finally, I deeply thank my mother and my grand parents for their invaluable support and encouragement in my life. This work would not have been possible without them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Recently, billions of devices, such as PCs, mobile phones and sensor nodes are connected to the Internet, to make our life safe, secure, and confortable. The scale of the Internet continuously increases in the number, heterogeneity, and mobility of devices and many new applications are emerging with help of proliferation of networked devices. For example, in the ambient information society [1–3], by distributing and embedding sensor devices in the environment and letting them cooperate with each other, the embedded network is expected to control the environment and provide personalized information services to a user taking into account time, place, occasion, and person. M2M (Machine to Machine) or IoT (Internet of Things) is another research area which attracts researchers and industry, as it enriches the quality and resolution of information useful for a variety of technologies by generating so-called "big data" which put too much burden on the Internet [4].

Information networks must be robust, adaptive, and reliable to accommodate the considerable number of heterogeneous devices and the huge amount of traffic. However, especially in wireless networks such as wireless ad-hoc and sensor networks, where the capacity of wireless links is limited, the centralized optimal control is not realistic. It is because collection of up-to-date information from and distributing control information to all nodes consume the vast amount of network resources. Therefore, in wireless ad-hoc and sensor networks, fully distributed and adaptive

control mechanisms are needed. Although a lot of distributed and adaptive control mechanisms are proposed, most of current mechanisms and protocols adopt complicated rules with fine-tuned parameters to achieve the optimal performance and to handle environmental changes such as node failures and movements [5–7]. It means that assumptions on the operational environment are made in defining rules and setting parameters. Such control mechanisms are prone to unexpected events and conditions and easily collapse, while they achieve the optimal performance in the assumed environment.

To address the problems, self-organizing control mechanisms, where each node autonomously decides its behavior based on local information it observes or obtains from neighbors and the global control emerges through mutual interaction among neighboring nodes, have gathered much attention. Although self-organizing phenomena are recognized in several research fields such as mathematics, physics, chemistry, social science, and biology, those researches inspired by self-organizing behavior of biological systems are most active and promising [8–10]. Biological systems are inherently autonomous and self-organizing, where there is no centralized control unit dominating the whole, and they often exhibit scalable, adaptive, and robust properties. By being inspired by such biological systems, it is expected that network control mechanisms can achieve the scalability, adaptability, and robustness.

Furthermore, since application's requirements on networks, e.g. type of devices to accommodate, required bandwidth, acceptable delay and packet loss, and operational policy, differ with each application, networks must provide various kinds of functionalities to a variety of applications. Since it is apparent that a monolithic network adopting a single self-organizing control mechanism cannot satisfy all requirements, we need to consider coordinating multiple networks adopting multiple self-organizing control mechanisms in a cooperative manner. Although there are many successful self-organizing control mechanisms, whose superiority to conventional mechanisms was verified through simulation and practical experiments, cooperation among multiple self-organizing control mechanisms is not well investigated.

In this thesis, we study self-organizing control and cooperation mechanisms in wireless ad-hoc and sensor networks. First we consider a self-organizing control mechanism for a wireless camera network where nodes cooperate with each other to adjust video coding rate not to exceed the

network capacity while obtaining high quality images of targets. Then, we propose a mechanism for multiple sensor networks to cooperate and be integrated for smooth communication among networks to offer a unified service to users. Finally, we further investigate the influence of interaction or interdependency between layered self-organizing mechanisms on performance and stability of the whole network system.

Wireless Sensor Networks

By distributing a large number of sensor nodes with wireless communication capability and organizing a wireless sensor network, we can obtain detailed information about surroundings, remote region, and objects [11, 12]. Each wireless sensor node can be equipped with a variety of sensors, e.g. thermometer, hygrometer, and illuminometer, as far as the space, cost, and energy allow. In wireless sensor networks, sensor nodes obtain observatory information about the environment and objects by using their sensors and report obtained sensor data to a base station or a sink node that has wireless or wired connection to the outer network, behind which an administrative user or monitoring center exists. In some cases, sensor data are directly sent to other sensor nodes to share information or to actuators to control other devices [13].

Control mechanisms for wireless sensor networks must be scalable, adaptive, and robust, because of a large number of sensor nodes, random or unplanned deployment of sensor nodes, and dynamic topology changes due to addition, movement, and removal of sensor nodes and unstable wireless communication. In addition, due to difficulty in managing a large number of nodes in a centralized manner, mechanisms must be fully distributed and self-organizing. Specifically, communication overhead to collect and maintain up-to-date information about the condition of a wireless sensor network is very costly in regarding to the wireless bandwidth and energy. Furthermore, sensor nodes are limited in power and computation capacities because a sensor node has a poor processor and small memories, and runs on battery power to save cost and size. Therefore, mechanisms for wireless sensor networks must be simple and energy efficient. Since each node decides its behavior based on a set of simple rules and the local information through only local communication, self-organizing control mechanisms are suited for wireless sensor networks.

Wireless Ad-hoc Networks

Ad-hoc networks are also infra-less wireless networks, which are formed dynamically without a central control unit as with wireless sensor networks. A wireless ad-hoc network is composed of a number of nodes such as laptop PCs, PDAs and mobile phones with computation and wireless communication capabilities, although the number of nodes is smaller and the capacity of nodes is higher than those in a wireless sensor network. One of major differences between ad-hoc networks and sensor networks is their communication paradigms, where ad-hoc nodes adopt a point-to-point communication, whereas sensor nodes mainly use a broadcast paradigm [11]. In ad-hoc networks, an ad-hoc node communicates with another ad-hoc node directly between the nodes, if they are neighboring, or through intermediate nodes acting as routers [14, 15]. Therefore, in ad-hoc networks, dynamic and adaptive routing mechanisms play an important role. As in the case of wireless sensor networks, since ad-hoc networks also suffer from unpredictable and dynamically changing nature of the wireless networks, control mechanisms such as routing protocols must be dynamic and adaptive, which is realized by self-organization.

1.2 Outline of Thesis

In this thesis, we study self-organizing control mechanisms in wireless ad-hoc and sensor networks. First, we propose a self-organizing video coding rate control mechanism for wireless camera networks, where nodes cooperate with each other to adjust video coding rate not to exceed the network capacity while obtaining high quality images of targets. By adopting the reaction-diffusion model, the coordinated adjustment of video coding ratio emerges without global knowledge, through iterative exchange of control information only among neighboring nodes and updating the state of node itself. Then, we propose a mechanism for multiple sensor networks to cooperate and be integrated for smooth communication among networks to offer a unified service to users. To bridge the gap between the intervals of sleep scheduling, only a part of nodes adjust their intervals in a stepwise fashion by adopting the pulse-coupled oscillator model. By strengthening the degree of entrainment at border nodes, the stepwise synchronization emerges in self-organizing manner. Finally, we further investigate the influence of interaction or interdependency between layered self-organizing

mechanisms on performance and stability of the whole network system. We considered a scenario where an ad-hoc network and an overlay network constructed on the ad-hoc network, both of which adopt an attractor selection model-based routing mechanism while they have different goals, coexist and influence each other. By modifying the way of sharing an objective parameter, i.e. activity, between the networks, we study the influence. Above studies on self-organizing control and cooperation mechanisms will provide a deeper understanding on self-organizing cooperation in wireless networks.

A Reaction-Diffusion based Coding Rate Control Mechanism for Camera Sensor Networks [16–18]

In Chapter 2, we begin this work by proposing, implementing, and evaluating a self-organizing video coding rate control mechanism. In camera sensor networks, which is composed of sensor nodes equipped with a camera, each node captures a video image around itself, and sends the video image toward a base station. Compared to typical scalar and even analog sensor data such as humidity, temperature, and acceleration, the volume of video data generated by cameras is considerably large, ranging from a few hundred Kbps to several hundred Mbps. The capacity of wireless sensor networks is limited to accommodate such huge video traffic. Although many studies, such as bandwidth allocation, retransmission control [19], FEC (Forward Error Correction) [20], and other error concealment techniques [21] have been made for QoS control for video transmission in wireless networks, they do not help much when the volume of video traffic is, for example, twice as much as the capacity of the network.

Therefore in large scale camera sensor networks, in addition to these QoS control mechanisms, application-level control to reduce the amount of video traffic without impairing the application is required. When we consider surveillance and monitoring applications, not all video data from cameras are equally important. Users are interested in video images that capture targets or certain phenomena. Therefore, the quality of video data from nodes detecting targets should be as high as possible without overwhelming the wireless network capacity, whereas those nodes far from the targets can suppress the video coding rate to avoid wasting the wireless network capacity in

transmission of irrelevant and redundant video data. For this purpose, we need a mechanism for camera sensor nodes to adjust the video coding rate in accordance with the location and velocity of targets.

To accomplish the above mentioned goal, we propose an autonomous mechanism based on a reaction-diffusion model [22] for coding rate control in camera sensor networks for remote surveillance and tracking applications. Specifically, we generate spot patterns of concentrations of virtual morphogens so that the morphogen concentrations have a peak at a camera detecting a target in its observation area. In addition, to prepare cameras in the direction of a moving target for future detection, their video coding rates, in other words, their morphogen concentrations are made slightly higher than others. Furthermore, we also propose a mechanism to keep the total of coding rate low for closely located targets considering the limited capacity of a wireless network. First, through simulation experiments, we verified the effectiveness of our proposal. Then we implemented our proposal on a wireless sensor network and conducted practical experiments. Through simulation and practical experiments, we verified that our proposal can avoid network congestion and keep the quality of video image from a camera having the target high.

An Inter-networking Mechanism with Stepwise Synchronization for Wireless Sensor Networks [23–26]

Next in Chapter 3, we consider self-organizing cooperation among multiple wireless sensor networks.

In general, wireless sensor networks adopt a sleep scheduling mechanism to save energy. Operational frequencies, that is, frequencies that they wake up and resume operation, are different among networks depending on application's requirement and characteristics of devices. When wireless sensor networks, whose operational frequencies are largely different, want to exchange information among them for coordinated actions, a node belonging to the faster network has to stay active in order to wait for a node belonging to the slower network to wake up in transmitting a message. Even when an energy-efficient MAC protocol such as S-MAC [27] and X-MAC [28] is used, such communication consumes the substantial energy at the former node and it would bring danger of

energy depletion.

To address the problem, we propose stepwise synchronization between wireless sensor networks for smooth and moderate inter-networking, where sensor nodes located near the border of two networks adjust their operational frequencies to bridge the gap in their intrinsic operational frequencies. Since only nodes near the border change their operational frequency, the remaining nodes can keep their frequency and thus energy consumption in inter-networking can be reduced. The stepwise synchronization is self-organized based on a nonlinear mathematical model of synchronization of oscillators, called the pulse-coupled oscillator (PCO) model [29]. The PCO model describes emergence of synchronization in a group oscillators with different frequencies by mutual interactions through stimuli. By adopting the PCO model to scheduling, operational frequencies of nodes can be appropriately adjusted without any centralized control in wireless sensor networks. In our mechanism, we strengthen the degree of entrainment at border nodes to intensively shift the operational frequency toward that of the other network while the degree of entrainment is weakened as the distance to the border increase. As a result, the operational frequencies of nodes near the border are adjusted to somewhere between the original operational frequencies of wireless sensor networks. Through simulation experiments, it was shown that the delay in communication between border nodes was reduced, and the energy consumption was balanced among nodes, which leads to prolongation of the lifetime of wireless sensor networks.

Study on Interaction between Layered Self-Organization based Control [30]

In Chapter 5, we consider a scenario where two self-organizing control mechanisms, both of which have different goals, coexist and influence each other.

As mentioned above, self-organization is considered one of key design principles to establish highly scalable, adaptive, and robust network systems to accommodate dynamic, diverse, and massive nodes and traffic. Although there are many proposals on self-organization based protocols that are useful, effective, and practical, there has never been any in-depth investigation into interaction, interference, and synergetic effects among multiple self-organization based control. In Chapter 5, we study mutual interaction among layered self-organization based control. We consider an ad-hoc

network and an overlay network that is constructed over the ad-hoc network, both of which adopt adaptive routing protocols based on the attractor selection model [31], i.e. a mathematical model of adaptive behavior of biological systems. We modified the degree of coupling by changing the way how layered self-organization control shared an objective parameter. Through simulation experiments, we confirmed that the upper layer-aware coupling can provide the best performance at the sacrifice of the stability such as variation of paths.

Chapter 2

A Reaction-Diffusion based Coding Rate Control Mechanism for Camera Sensor Networks

A wireless camera sensor network is useful for surveillance and monitoring for its visibility and easy deployment. However, it suffers from the limited capacity of wireless communication and a network is easily overflowed with a considerable amount of video traffic. In this chapter, we propose an autonomous video coding rate control mechanism where each camera sensor node can autonomously determine its coding rate in accordance with the location and velocity of target objects. For this purpose, we adopted a biological model, i.e. reaction-diffusion model, inspired by the similarity of biological spatial patterns and the spatial distribution of video coding rate. Through simulation and practical experiments, we verify the effectiveness of our proposal.

2.1 Introduction

By distributing a large number of sensor nodes with wireless communication capability and organizing a wireless sensor network (WSN), we can obtain detailed information about surroundings, remote region, and objects [11]. Sensor nodes are equipped with a variety of sensors depending on

applications. In particular, a camera sensor network, which is composed of sensor nodes equipped with a camera, are useful in applications such as remote surveillance and monitoring [32, 33] and thus it is one of key technologies for the safe and secure living environment. For example, by placing camera sensor nodes along streets, we can monitor the traffic condition, children on the way to and from a school, or track suspicious people.

Generally, control mechanisms for sensor networks must be scalable, adaptive, and robust, because of a large number of sensor nodes, random or unplanned deployment, and dynamic topology changes due to addition, movement, and removal of sensor nodes. In addition, due to difficulty in managing a large number of nodes in a centralized manner for the limited bandwidth insufficient to periodically collect the up-to-date information from nodes, mechanisms must be fully distributed and self-organizing. Camera sensor networks place additional challenging issues as summarized in [34]. In particular, compared to typical scalar and even analog sensor data such as humidity, temperature, and acceleration, the volume of video data generated by cameras are considerably large, ranging from a few hundred Kbps to several hundred Mbps. The capacity of wireless sensor networks is limited to accommodate such huge video traffic. For example, the communication rate of ZigBee with IEEE 802.15.4 [35], a standard protocol for WSN, is only 250 Kbps. Even when we adopt IEEE 802.11b with the capacity of 11 Mbps, theoretical analysis indicates that the throughput available to each node is in the order of $O(\frac{1}{\sqrt{n}})$, where n is the number of nodes in the multi-hop wireless network [36]. Therefore, only a few nodes can join the network, even when each node transmits video data coded at only a few Mbps using IEEE 802.11b. Many studies, such as bandwidth allocation, retransmission control [19], FEC (Forward Error Correction) [20], and other error concealment techniques [21] have been made for QoS control for video transmission in wireless networks. They are useful and effective to maintain the quality of video data when a wireless network is lightly loaded and moderately congested, but they do not help much when the volume of video traffic is, for example, twice as much as the capacity of the network.

Therefore in large scale camera sensor networks, in addition to these QoS control mechanisms, application-level control to reduce the amount of video traffic without impairing the application is required. When we consider surveillance and monitoring applications, not all video data from cameras are equally important. Users are interested in video images that capture targets or certain

phenomena. Therefore, the quality of video data from nodes detecting targets should be as high as possible without overwhelming the wireless network capacity, whereas those nodes far from the targets can suppress the video coding rate to avoid wasting the wireless network capacity in transmission of irrelevant and redundant video data. For this purpose, we need a mechanism for camera sensor nodes to adjust the video coding rate in accordance with the location and velocity of targets. To the best of our knowledge, such application-level control has not been addressed well and we can only find a few [37] [38]. In [37], a content-aware control system assigning higher rate to cameras capturing important images is proposed. However, they mainly consider bandwidth allocation among cameras on a node, not among camera nodes. In [38], the authors propose dynamic allocation of wireless communication bandwidth to camera nodes based on the rate-distortion model and the activity of captured video. However, the bandwidth allocation is performed in a centralized manner, where a central location has the complete information about the status of wireless network and characteristics of video data at all cameras.

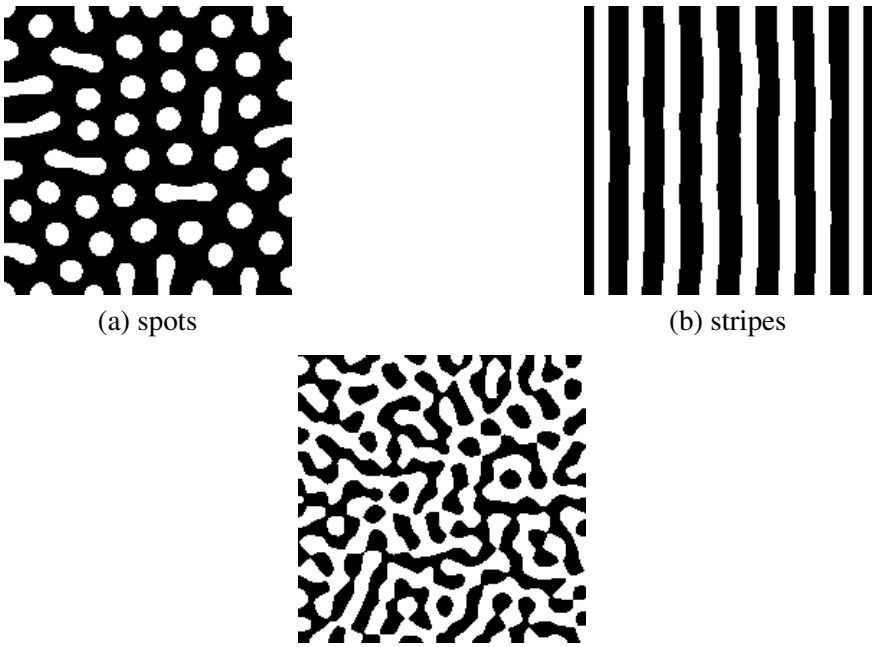
Only if we know the complete and up-to-date information about the location and status of camera sensor nodes and the location and velocity of targets, we can optimally assign the wireless network capacity to those nodes and set the video coding rate to avoid congestion of the network while keeping the application-level QoS. However, maintenance of such global and up-to-date information involves much communication overhead and the battery power of nodes. Therefore, an autonomous, distributed, and self-organizing mechanism is required for each node to appropriately determine the video coding rate based on local information obtained by exchanging messages, e.g. general HELLO messages, with neighboring nodes, while avoiding local network congestion.

To accomplish the above mentioned goal, we adopt a reaction-diffusion model in this chapter. A reaction-diffusion model was first proposed by Dr. Alan Turing as a mathematical model for pattern generation on the surface of body of fishes and mammals [22] . In a reaction-diffusion model, through local interactions among molecules of neighboring cells, a variety of spatial patterns, i.e. heterogeneous distribution of morphogen concentrations, emerge in a self-organizing manner. Autonomously generated patterns can be used for routing, clustering, scheduling, and topology control on sensor networks [39–42]. In smart sensor networks for a forest fire application, a stripe pattern is organized from a robot load point to a fire control point through local and mutual interactions

among distributed sensor nodes and mobile robots walk along the stripe to fight a fire [39]. RD-MAC [40] is a reaction-diffusion based MAC protocol, where they noticed the similarity among a scheduling pattern of spatial TDMA and a spot pattern of leopards. A node inhibits packet emission of neighboring nodes in its range of radio signals while encouraging nodes out of the range to send packets for better spatial use of a wireless channel. For camera sensor networks, a cooperative control model for a surveillance system, which consists of plural Pan-Tilt-Zoom cameras and has no central control unit, is proposed [42]. Each camera adjusts their observation area to decrease blind spots in the whole surveillance area by control algorithms based on a reaction-diffusion model. Our early work on a reaction-diffusion based control mechanisms for a sensor network [41] verified the practicality of reaction-diffusion based pattern formation on a wireless sensor network under the influence of collisions through experiments and two acceleration schemes for faster pattern generation were proposed.

In this chapter, we propose an autonomous mechanism based on a reaction-diffusion model for coding rate control in camera sensor networks for remote surveillance and tracking applications. Specifically, we generate spot patterns of concentrations of virtual morphogens so that the morphogen concentrations have a peak at a camera detecting a target in its observation area. In addition, to prepare cameras in the direction of a moving target for future detection, their video coding rates, in other words, their morphogen concentrations are made slightly higher than others. As a result, we see a pattern of concentric circles for a stationary target and an elliptic pattern for a moving target (see Fig.2.3). Furthermore, we also propose a mechanism to keep the total of coding rate low for closely located targets considering the limited capacity of a wireless network. First, through simulation experiments, we verified the effectiveness of our proposed mechanism. Then we implemented our proposed mechanism on a wireless sensor network composed of PCs with cameras and conducted practical experiments. Through practical experiments, we verified that the video coding rate was appropriately adjusted in accordance with the location and velocity of a moving target. We also evaluated the effectiveness of the mechanism in avoidance of congestion and the perceived quality of received video data.

The rest of this chapter is organized as follows. In Section 2.2, we introduce a pattern generation mechanism of a reaction-diffusion model. Next in Section 2.3, we describe our reaction-diffusion



(a) spots (b) stripes

(c) maze

Figure 2.1: Example of patterns generated by reaction-diffusion model

based control mechanism for camera sensor networks. In Section 2.4, we show and discuss results of simulation experiments. In Section 2.5, we describe our implementation of the mechanism and show and discuss results of practical experiments. Finally, we conclude this chapter in Section 2.6.

2.2 Pattern Formation by Reaction-Diffusion Model

In a reaction-diffusion model, by mutual interaction among neighbor cells through chemical reaction and diffusion of two morphogens, spatially heterogeneous distribution of concentration of morphogens appears. Depending on the form of reaction-diffusion equations and their parameters, a variety of patterns can be generated as illustrated in Fig. 2.1.

Generally, a reaction-diffusion model is formulated by a pair of partial differential equations as

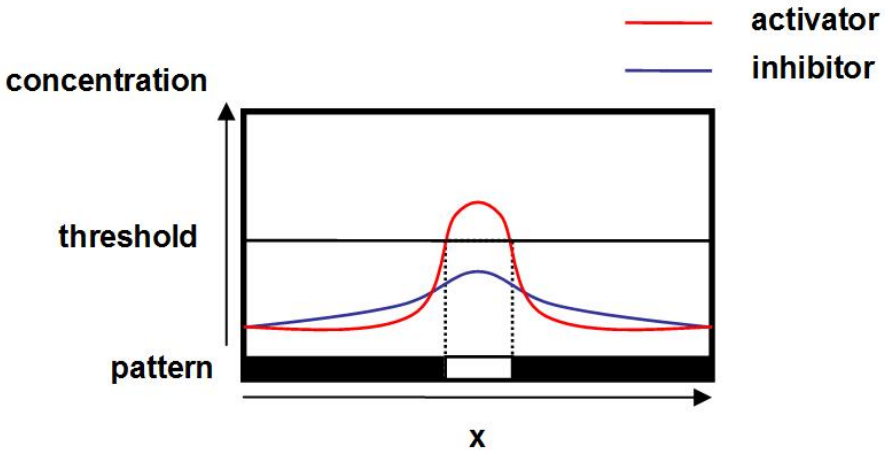


Figure 2.2: Concentration distribution of activator and inhibitor

follow.

$$\begin{cases} \dfrac{\partial u}{\partial t} = F(u,v) + D_u \nabla^2 u \\ \dfrac{\partial v}{\partial t} = G(u,v) + D_v \nabla^2 v, \end{cases} \tag{2.1}$$

where u and v are the concentrations of activator and inhibitor, respectively. D_u and D_v are the diffusion rate of activator and inhibitor respectively. F and G are functions for reactions.∇^2 is the Laplacian operator. The first term of the right-hand side is called a reaction term and the second term is called a diffusion term. In the reaction-diffusion model, the following two conditions must be satisfied to generate patterns. First, the activator activates itself and the inhibitor, whereas the inhibitor restrains itself and the activator. Second, the inhibitor diffuses faster than the activator ($D_v > D_u$).

A mechanism of pattern generation can be explained as follows. In Fig. 2.2, those hypothetical chemicals are arranged in a line on the horizontal axis. The vertical axis corresponds to the concentrations of activator and inhibitor. Now consider that the concentration of activator has the peak at the point x illustrated in Fig. 2.2, by a slight perturbation. The concentrations of activator and inhibitor are increased around the point x by being activated by the activator. The generated inhibitor diffuses faster than the activator and restrains generation of activator at further regions.

On the other hand, at the point x, because of the different rate of diffusion, the activator stays and the concentration of activator is kept higher than that of inhibitor. Consequently, the diversity in the concentration of activator emerges and a pattern appears. For example, when we color points where the concentration of activator exceeds a certain threshold with white and others with black, we can see a black-while-black pattern shown at the bottom of Fig. 2.2.

In this chapter, since we want to generate spot patterns where spots correspond to targets, we introduce the *stimulus* which is an increment of activator and apply the following reaction-diffusion equation.

$$\begin{cases} \dfrac{\partial u}{\partial t} = F(u,v) + D_u \nabla^2 u - du + E(t) \\ \dfrac{\partial v}{\partial t} = G(u,v) + D_v \nabla^2 v - gv, \end{cases} \tag{2.2}$$

and

$$\begin{cases} F(u,v) = \max\{0, \min\{au - bv - c, M\}\} \\ G(u,v) = \max\{0, \min\{eu - hv - f, N\}\}, \end{cases} \tag{2.3}$$

where d and g are parameters for decomposition or decrease of morphogens per unit time. $E(t)$ is the amount of stimulus. Intentionally placed activator, i.e. $E(t)$, increases the activator concentration at the point and a spot centered at the point would emerge. a and e correspond to the rate of activation and b and h are for inhibition. c and f are parameters for decrease of morphogens per unit time. M and N are constants of limit.

Theoretically speaking, these parameters must satisfy the following conditions to generate a pattern centered at a point where stimulus $E(t) > 0$ exists [43].

$$\frac{a-d}{b}\frac{f}{e} - \frac{c}{b} \leq 0 \ \text{ and } \ \frac{a-d}{b}\frac{M}{d} - \frac{c}{b} < \min\{\frac{e}{h+g}(\frac{M}{d} - \frac{f}{e}), \frac{N}{g}\} \tag{2.4}$$

or

$$a - d > h + g \ \text{ and } \ \frac{a-d}{b}\frac{M}{d} - \frac{c}{b} < \min\{\frac{e}{h+g}(\frac{M}{d} - \frac{f}{e}), \frac{N}{g}\} \tag{2.5}$$

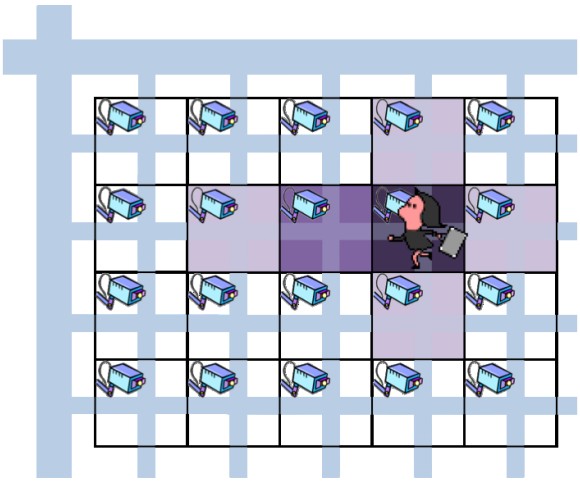


Figure 2.3: Camera sensor network

As can be seen, the parameter setting does not depend on system conditions such as the size of region.

2.3 Reaction-Diffusion based Coding Rate Control Mechanism

Figure 2.3 illustrates a surveillance or monitoring system that we consider in this chapter. Each square corresponds to the observation area of a camera sensor node. The darker the square is, the higher the video coding rate is. We assume that nodes are arranged in a grid topology, considering town or room monitoring as an application of the mechanism. For example in a town, we can consider such a scenario where camera sensor nodes are placed at intersections. A node can communicate with four neighbors in up, right, down, and left directions. Nodes at a corner of the monitoring region have two neighbors and nodes at an edge have three neighbors. The assumption on node layout can be relaxed by further discretization $\nabla^2 X(X = u, v)$ as follow.

$$\nabla^2 X = \sum_{j \neq i} \kappa_{i,j}(X_j - X_i) \tag{2.6}$$

$$\kappa_{i,j} = \begin{cases} d_{i,j}^{-2}, & d_{i,j} \leq R, \\ 0, & \text{otherwise} \end{cases} \tag{2.7}$$

where, X_i and X_j are the concentration of morphogens in node i and j, and $d_{i,j}$ is the distance between the two nodes. R is the maximum transmission range.

Each node has a camera and a wireless communication device. A camera or a node has the capability of object recognition and motion detection with which the existence, speed, and direction of a target in its observation area are recognized.

2.3.1 Basic Behavior

Basically, at regular control intervals of T seconds, each node calculates the reaction-diffusion equation by using the information it received in the preceding control interval, adjusts its video coding rate in accordance with morphogen concentrations, and then broadcasts a message containing information about its morphogen concentrations, stimulus E, attenuation coefficient A, and NIP (Notification of Inhibitor Peak) notification to its neighbors. Message emission is done once per control interval and the message can be combined with a general HELLO message. We call the duration between the t-th control timing and the $t+1$-th control timing as the t-th control interval. We should note here that nodes behave in an asynchronous manner, although the control interval is identical among nodes. It means that timing of reaction-diffusion calculation and message emission is different among nodes.

The reaction-diffusion equation is identical among nodes. Since nodes are arranged in a grid layout and messages are exchanged at regular intervals, we spatially and temporally discretize Eqs. (2.2) and (2.3) as follows.

$$\begin{cases} u_t = u_{t-1} + \Delta t \Big\{ F(u_{t-1}, v_{t-1}) - d u_{t-1} + E(t-1) \\ \qquad + D_u \dfrac{(u^u_{t-1} + u^d_{t-1} + u^l_{t-1} + u^r_{t-1} - 4u_{t-1})}{\Delta h^2} \Big\} \\ v_t = v_{t-1} + \Delta t \Big\{ G(u_{t-1}, v_{t-1}) - g v_{t-1} \\ \qquad + D_v \dfrac{(v^u_{t-1} + v^d_{t-1} + v^l_{t-1} + v^r_{t-1} - 4v_{t-1})}{\Delta h^2} \Big\}, \end{cases} \tag{2.8}$$

and

$$\begin{cases} F(u_{t-1},v_{t-1})=\max\{0,\min\{au_{t-1}-bv_{t-1}-c,M\}\} \\ G(u_{t-1},v_{t-1})=\max\{0,\min\{eu_{t-1}-hv_{t-1}-f,N\}\}. \end{cases} \tag{2.9}$$

At the t-th control timing, a node calculates the above reaction-diffusion equation to derive its morphogen concentrations u_t and v_t. A set of u^u_{t-1}, u^r_{t-1}, u^d_{t-1}, and u^l_{t-1} and a set of v^u_{t-1}, v^r_{t-1}, v^d_{t-1}, and v^l_{t-1} correspond to concentrations of activator and inhibitor of neighboring nodes in up, right, down, and left directions. These values are obtained from messages that a node received in the $t-1$-th control interval. If a node did not receive a message from a neighboring node in the $t-1$-th control interval, the latest value obtained in the preceding intervals is used instead. Δh and Δt correspond to the distance between nodes and the discrete step interval of time, respectively. There is the theoretical range of Δt for the equation reaches convergence and a stable pattern is formed.

$$0<\Delta t<\min\{\frac{2}{d+4D_u(\Delta x^{-2}+\Delta y^{-2})},\frac{2}{g+4D_v(\Delta x^{-2}+\Delta y^{-2})}\}. \tag{2.10}$$

If the degree of temporal discretization is not within this range, a pattern does not converge. $E(t-1)$ in Eq. (2.8) is the amount of stimulus which is determined at the t-th control timing, based on messages it received in the $t-1$-th control interval and the condition of a target if exists. The stimulus controls the distribution of morphogen concentrations, that is, a pattern. Usually, the amount of stimulus is zero. A node which detects a target in the $t-1$-th control interval appropriately sets E and A in accordance with the speed and direction of the target, so that a spot pattern centered at the node is generated. The stimulus diffuses to nodes in the direction of target movement so that they prepare for the future appearance of the target in their observation area as shown in Fig. 2.3. The attenuation coefficient A is used for this purpose. Details of stimulus determination and diffusion will be explained in subsection 2.3.2. NIP notification is used to regulate the amount of stimulus when two or more targets are closely located. Details of NIP will be given in subsection 2.3.3.

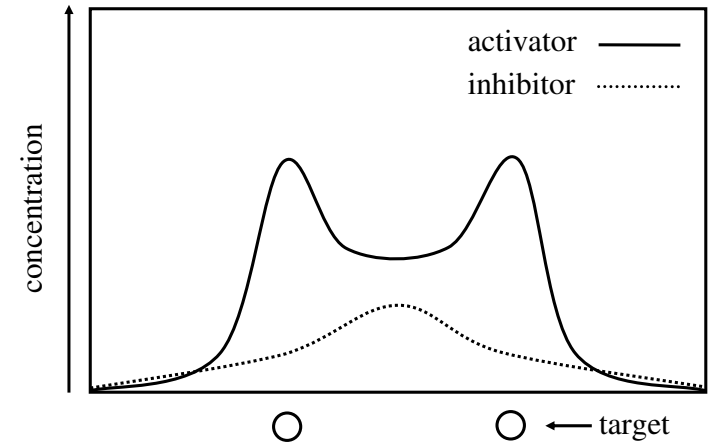
Once morphogen concentrations are derived, a node translates the concentrations to the video coding rate that it uses during the t-th control interval. As explained in Section 2.2, a spatial pattern

generated by a reaction-diffusion model comes from the spatial heterogeneity in concentration of activator. However, an approach to directly map the concentration of activator to the video coding rate fails when two or more targets are close together. In Fig. 2.4(a), we illustrate the distribution of concentrations of morphogens. As can be seen, the region between two peaks of the concentration of activator has the slightly high concentration of activator, because the region is activated by diffused activator while it is also inhibited by diffused inhibitor. If we set the video coding rate in proportional to the concentration of activator for example, regions inbetween closely located targets generate unnecessarily high-quality video data. Therefore, in our mechanism, by focusing on a phenomenon that the concentration of inhibitor is also high in the region as shown in Fig. 2.4(a), a node determines the video coding rate based on $u/\sqrt{v}$, whose distribution is illustrated in Fig. 2.4(b). If $v < 1$, then we set $u/\sqrt{v} = 0$, in order to avoid the divergence of $u/\sqrt{v}$. In this chapter, we determine the video coding rate based on the value $u/\sqrt{v}$.

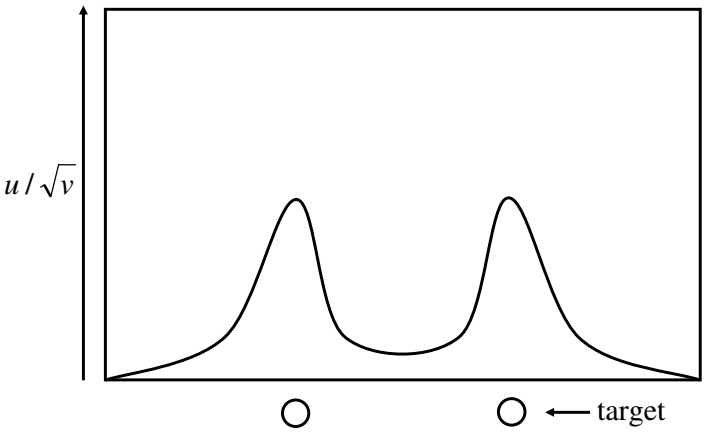
2.3.2 Stimuli Arrangement

In the reaction-diffusion equation, stimuli decide the position, shape, and size of spot patterns. When we set the stimulus high at a certain node, the concentration of activator becomes high at the node and a spot centered at the node emerges. To keep monitoring a moving target, cameras in the region to which the target is expected to move should use the sufficiently high coding rate for the future appearance (Fig. 2.3). Therefore, in our mechanism, a node detecting a moving target diffuses the stimulus to nodes in the moving direction while the stimulus decreases as the distance to the target increases.

If a node has a target in its observation area at the control timing, it sets the amount of stimulus E, the attenuation coefficient A, and the direction of diffusion. The node calculates the morphogen concentrations, set the video coding rate, and broadcasts a message containing this information. A node receiving a message first sees whether it is in the direction of the target movement. If not, it ignores the information. If the node should be prepared for the target, it first calculates the amount of stimulus E' from the informed E and A as $E' = A \times E$. This stimulus is used as $E(t-1)$ at the t-th control timing in calculating the morphogen concentrations. After the calculation, the information



(a) Distribution of morphogen concentrations



(b) Distribution of $u/\sqrt{v}$

Figure 2.4: Distribution of u, v, and $u/\sqrt{v}$ with closely located targets

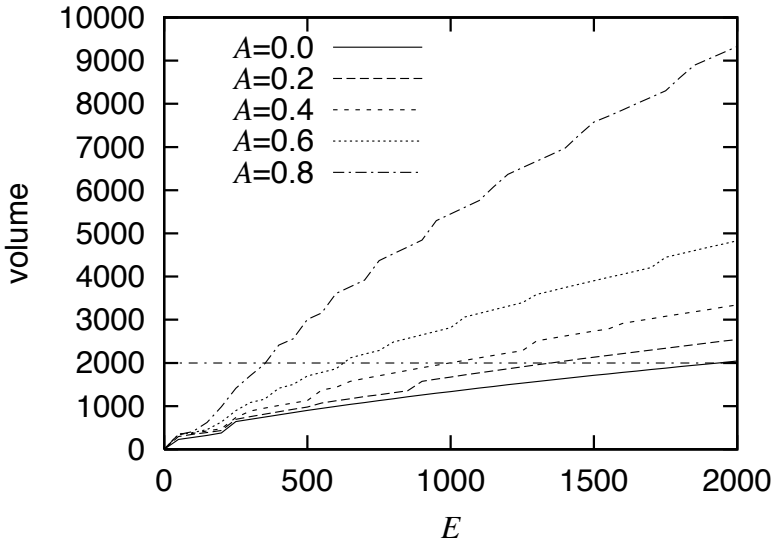


Figure 2.5: Relationship among A, E, volume of pattern

about the stimulus, including E', A, and the direction is further diffused to neighbor nodes by being embedded in a broadcast message. When a node receives multiple messages containing stimuli from neighbor nodes, it uses the sum of E' as $E(t-1)$.

The amount of stimulus E and the attenuation coefficient A are determined in accordance with the movement of target and the capacity of wireless channel. In Fig. 2.5, the relationship among E, A, and the volume of a generated pattern is shown for the case with a single target. The volume of a pattern is defined as,

$$v(E, A) = \sum_{(i,j)} u(i,j)/\sqrt{v(i,j)}. \tag{2.11}$$

$u(i,j)$ and $v(i,j)$ are the concentrations of activator and inhibitor at node (i,j) on a converged stable pattern, respectively. Since a node chooses the video coding rate depending on $u(t)/\sqrt{v(t)}$, the volume corresponds to the total amount of traffic generated by nodes in a spot. As shown in the figure, the volume $v(E,A)$ is almost in proportional to the stimulus E. As the attenuation coefficient A increases, the volume $v(E,A)$ increases for the same stimulus E. Since the attenuation coefficient A determines the range of stimulus diffusion, a node detecting a target first sets the attenuation coefficient A in accordance with the speed V of target. Then, the node determines

the stimulus E to keep the total traffic at a certain volume from $v(E, A)$. For example, when the capacity of local wireless network is 2000 in volume and A is 0.4, E is set at 1010. The mapping from the speed to the attenuation coefficient depends on the system conditions and application requirements. When the distance between adjacent nodes is large, the attenuation coefficient should be large not to spread a spot pattern too broadly and thus not to exceed the capacity of local wireless network. Regarding mapping from the network capacity to the volume, we suggest to use 2000 for any capacity as far as actual video coding rate is adjusted accordingly. In our experiments we use MPEG-2 as a coding algorithm and the coding rate ranges from 0.75 Mbps to 2 Mbps for an IEEE 802.11g network. In case of IEEE 802.11b for example, the maximum coding rate should be kept as low as 400 kbps while using the same parameters as our IEEE 802.11g network, e.g. local network capacity of 2000 in volume.

2.3.3 Stimuli Adjustment

With the mechanisms we explained so far in this chapter, a desired spot pattern appears and each node generates video data with the appropriate quality in accordance with the location, speed, and direction of a moving target without knowing the complete information about the whole system. However, when two or more targets are located close together, the total amount of video traffic would exceed the network capacity in that area. As a consequence, the perceived video quality considerably deteriorates for loss and delay of video data. To tackle the problem, we additionally propose the stimuli adjustment mechanism.

A basic idea is as follows. When two or more targets are close together, spots centered at them overlap with each other. If a node can detect the overlap and inform to the nodes setting the stimuli, the amount of stimuli can be adjusted so that spots become small and apart from each other. As shown in Fig. 2.4(a), when two targets are closely located, both of concentrations of activator and inhibitor become high at the inbetween region. Especially, the concentration of inhibitor has a peak at the center. By using this phenomena, a node at the overlapping point detects the overlap.

At the control timing, a node compares the concentration of inhibitor of itself with those of neighboring nodes. If the inhibitor concentration is the highest at the node and it does not have a

Table 2.1: Parameter settings for reaction-diffusion equation

parameter	value	parameter	value
a	0.08	h	0.05
b	0.2	D_u	0.004
c	0.2	D_v	0.1
d	0.03	M	0.2
e	0.1	N	0.5
f	0.14	Δt	0.1
g	0.06	Δh	1.0

target in its observation area, it sets the NIP notification in a message it broadcasts. Now, a node receives a message. If $u/\sqrt{v}$ of the node is higher than that of a node from which it received NIP, the node has the stimulus, or any neighboring node of the node has a target, it sets NIP in a broadcast message. As a consequence, NIP follows the gradient of $u/\sqrt{v}$ and diffused stimuli toward nodes having a target.

When a node having a target receives a message with NIP, it reduces the stimulus E as $E \times \alpha$ ($0< \alpha <1$). When the targets move apart from each other and the overlap disappears, the stimulus has to be increased. Therefore, a node having a target increases the stimulus as $E = E + \Delta e$, if it does not receive any NIP in the preceding control interval. The stimulus must be large enough to generate a pattern and smaller than the maximum to keep the volume. The range is determined from $v(E, A)$. For example, with $A = 0.4$, the range is from 440 to 1010.

2.4 Simulation Experiment

In this section, we show results of simulation experiments to verify the effectiveness of our mechanism. One hundred nodes are arranged in a 10×10 grid with separation of 100 meters. Parameter setting for the reaction-diffusion equation is summarized in Table 2.1. These parameters are chosen to satisfy the Turing condition of Eq. (2.4). For the stimuli adjustment, $\alpha = 0.999$ and $\Delta e = 1$ are used. We determine the relationship $v(E, A)$ from Fig. 2.5, which is obtained by preliminary experiments with one stationary target. Assuming that the capacity of wireless network is 2000 in

Table 2.2: Mapping from V to A and range of E

V (km/h)	A	Upper limit of E	Lower limit of E
$V = 0$	0.0	1960	830
$0 < V \leq 2$	0.2	1370	700
$2 < V \leq 4$	0.4	1010	440
$4 < V \leq 6$	0.6	620	390
$6 < V$	0.8	360	260

volume, the mapping from the speed V to the attenuation coefficient A and the range of the stimulus E are summarized in Table 2.2. These values are determined from $v(E, A)$ shown in Fig. 2.5, so that the area of high coding rate spreads toward the direction of a moving target while keeping the total volume of a generated pattern does not exceed the capacity of a wireless network as will be verified in Section 2.4.2. Initially, the morphogen concentrations are set at zero.

2.4.1 Stationary Target

First, we consider a scenario where there is one stationary target ($V = 0$) at the location of (5,5) in the monitoring region ($0 \leq x \leq 9, 0 \leq y \leq 9$). Figure 2.6 illustrates the distribution of $u/\sqrt{v}$ when a pattern converges. In the figure, each square corresponds to a node or its monitoring area. The darker the square is, the higher the video coding rate and $u/\sqrt{v}$ are. A triangle indicates the location and direction of a node. As shown in the figure, a spot pattern centered at the target is self-organized. Figure 2.7 shows the distribution of morphogen concentrations and $u/\sqrt{v}$ on a horizontal line $y = 5$. As shown in the figure, the concentration of activator and $u/\sqrt{v}$, that is, the video coding rate is the highest at the node having the target.

In a reaction-diffusion model, a pattern does not appear at once. Since calculation and communication require energy and time and adjustment of video coding rate must be performed in time to monitor a moving target, we have to consider the time required for pattern generation. Figure 2.8(a) shows the transition of the total volume against the number of calculations to show how fast a stable pattern emerges. For a pattern to converge, nodes have to calculate the reaction-diffusion equation and exchange messages about 2800 times as a line labeled as “normal” shows. To accelerate pattern

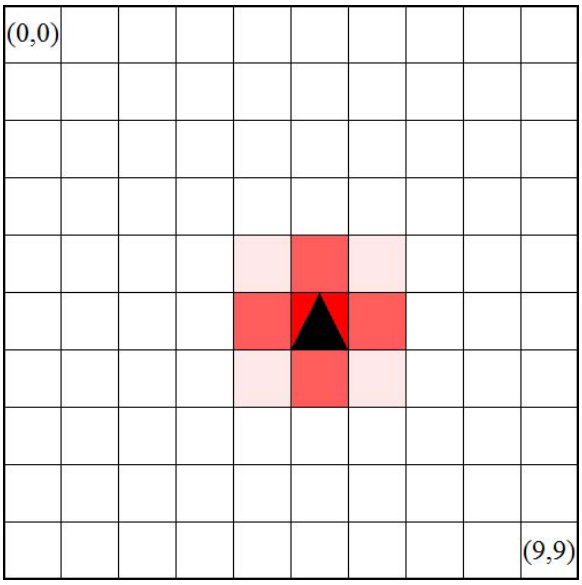
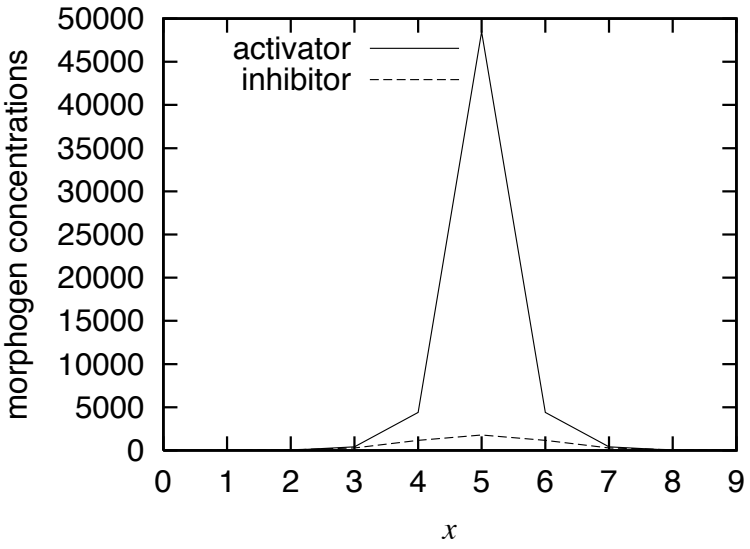
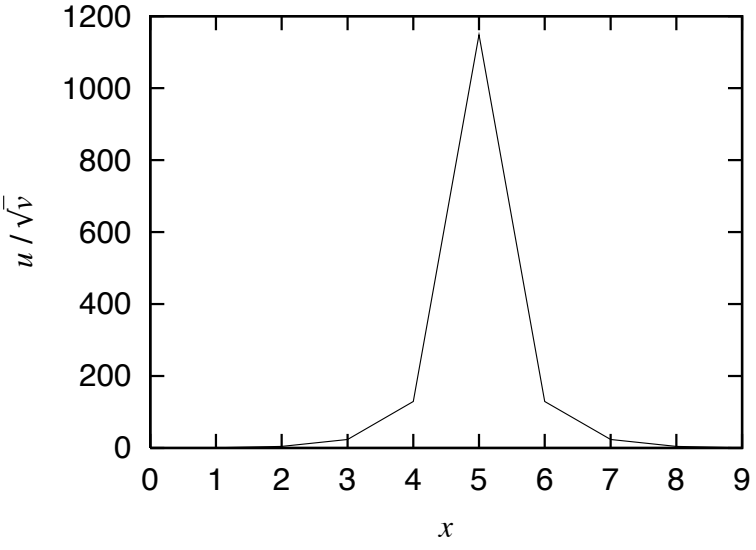


Figure 2.6: Distribution of morphogen concentrations with one stationary target

generation, we introduce an acceleration method using a larger discrete step interval Δt [41]. A line labeled as "acceleration" in Fig. 2.8(a) shows the result of acceleration by setting Δt at 2.0, which is within the limit of $\Delta t = 2.32$ derived from Eq. (2.10). By the acceleration, the number of calculations is greatly reduced to 140. Now, assume that a target is moving at the speed of 4 km/h. A target passes across the observation area of 100 m $\times$ 100 m in 90 seconds. Since the control interval should be in an order of several seconds at least, the 140-times calculation is still too large to generate a pattern in time. However, it is not necessarily required for the whole pattern to converge from a practical point of view as far as a camera having a target generates the high-quality video data. Figure 2.8(b) shows the transition of $u/\sqrt{v}$ at a node detecting a target. For $u/\sqrt{v}$ to converge, the node needs to calculate the reaction-diffusion equation 85 times, but the concentration drastically increases to the sufficiently high value in about 30-times calculations. As nodes exchange messages and calculate the reaction-diffusion equation, further nodes eventually adjust the video coding rate as shown in Fig. 2.8(b). A pattern is gradually generated from the center.

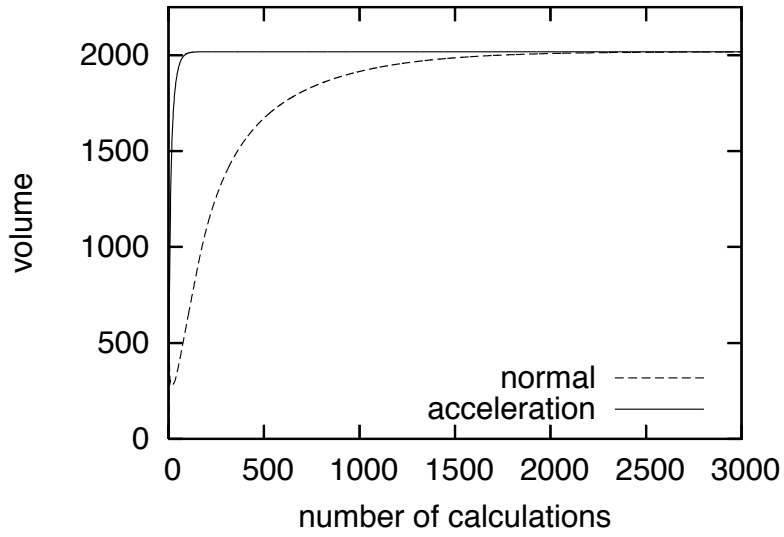


(a) Distribution of morphogen concentrations

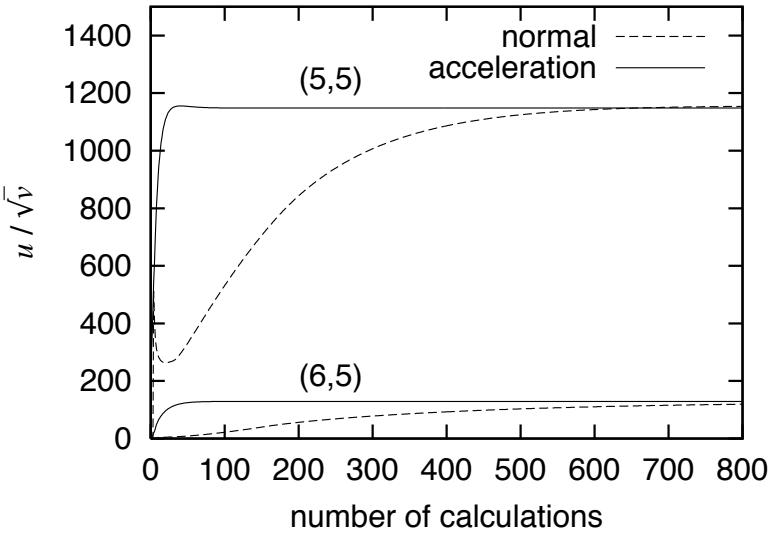


(b) Distribution of $u/\sqrt{v}$

Figure 2.7: Simulation results of one stationary target

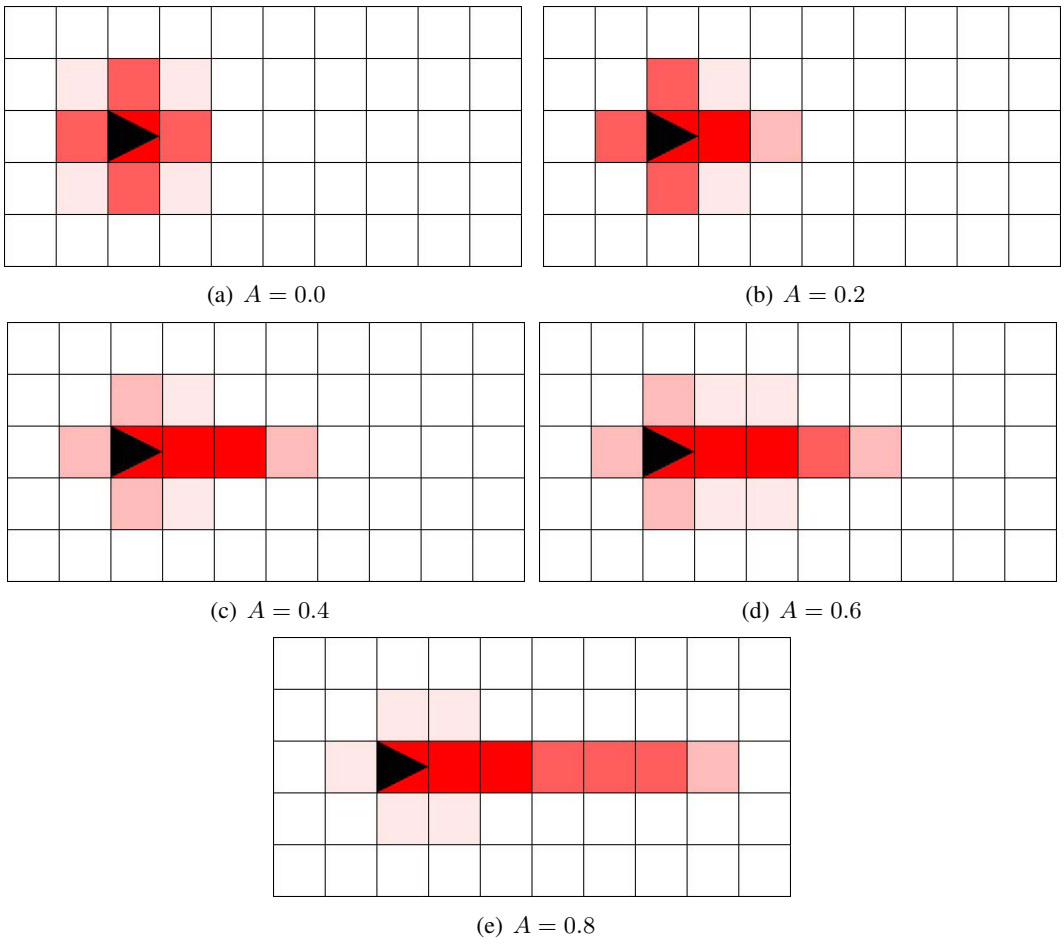


(a) Transition of volume



(b) Transition of $u/\sqrt{v}$

Figure 2.8: Transition of volume and video coding rate against number of calculations



(a) $A = 0.0$ (b) $A = 0.2$

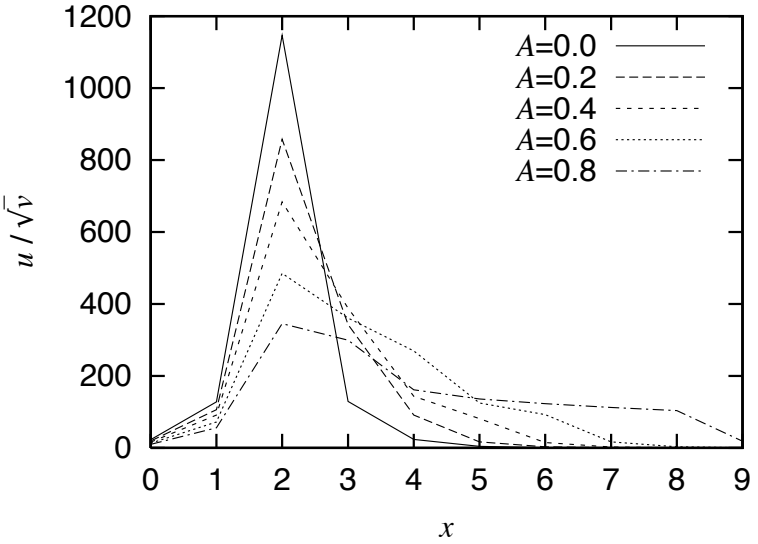
(c) $A = 0.4$ (d) $A = 0.6$

(e) $A = 0.8$

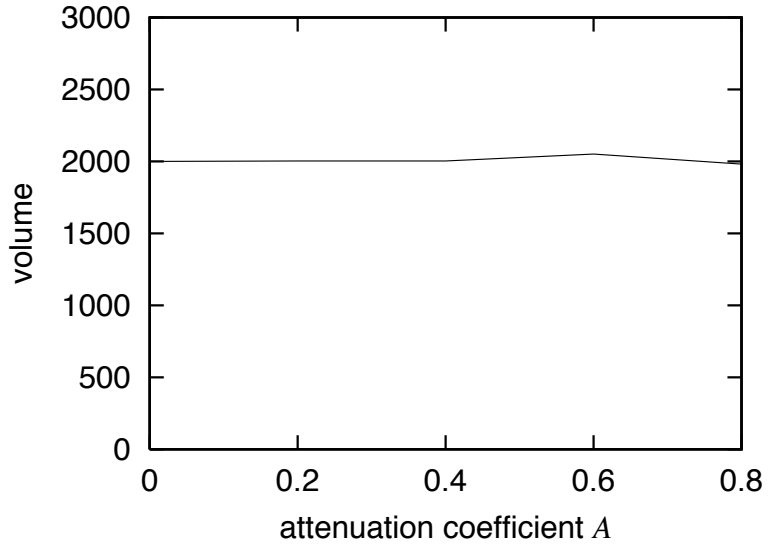
Figure 2.9: Generated patterns for a moving target

2.4.2 Moving Target

Next we make a node located at (2,5) move at the speed of 1, 3, 5, and 7 km/h, which corresponds to the attenuation coefficient A of 0.2, 0.4, 0.6, and 0.8, respectively. Figure 2.9 illustrates generated patterns. Figure 2.10(a) shows the distribution of $u/\sqrt{v}$ for the different attenuation coefficient on $y = 5$. As can be seen, as A, i.e. the speed of moving target, increases, the resultant pattern spreads wider and the height of peak becomes lower. As shown in Fig. 2.10(b), the total volume is kept constant at the given capacity even when the speed changes with our stimulus setting.

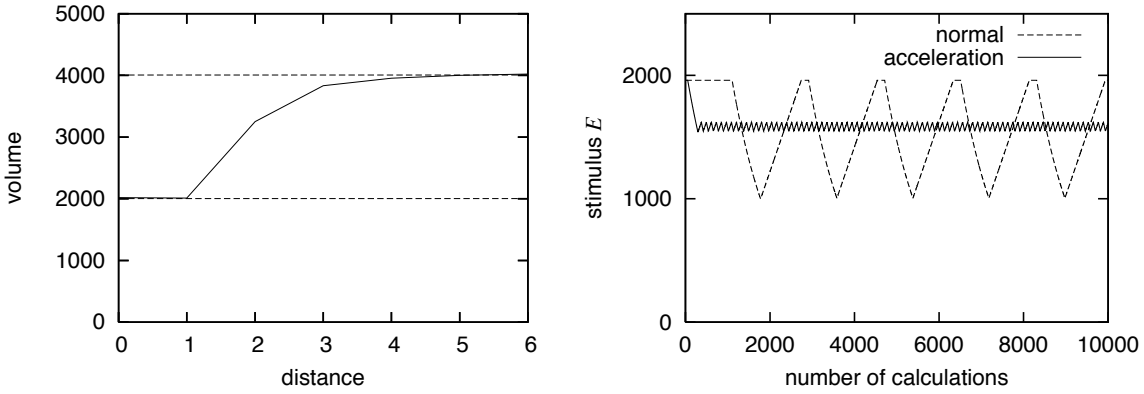


(a) Distribution of $u/\sqrt{v}$



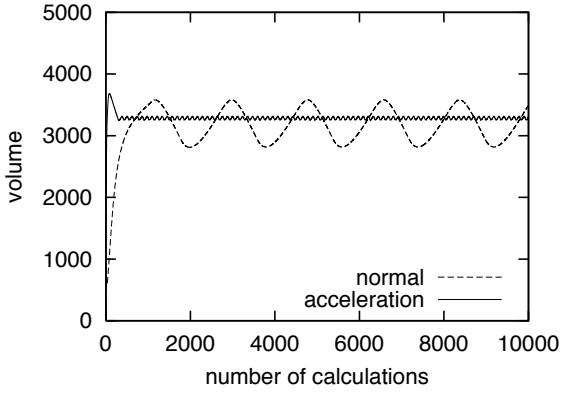
(b) Volume

Figure 2.10: Simulation results of one moving target



(a) Volume against target distance

(b) Transition of stimulus



(c) Transition of volume

Figure 2.11: Simulation results of two stationary targets

2.4.3 Two Stationary Targets

To verify the effectiveness of our stimuli adjustment for closely located targets, we conduct simulation experiments with two stationary targets. Figure 2.11(a) shows the relationship among the distance between targets and the total volume of converged pattern. Two dashed lines correspond to the volume for the case of one stationary target (lower line) and its doubled amount (upper line), respectively. The distance of zero corresponds to the case of one stationary target. As shown in the figure, the total volume is suppressed when the distance between two targets is small. In cases of the distance of one and two, intermediate nodes detect the overlap, NIP is sent to the nodes detecting targets, and the stimuli are decreased. Once the stimuli become small enough, the two nodes begin

to increase the stimuli again for not receiving NIP. As a result, the stimuli fluctuate as shown in Fig. 2.11(b), where the transition of stimulus at one of nodes having a target is depicted for the case of distance of two. It should be noted that the acceleration method also contributes to suppression of the oscillation. We see the same effect in the total volume as shown in Fig. 2.11(c). In the case of distance of three or more, spots centered at targets do not overlap with each other and NIP is not used.

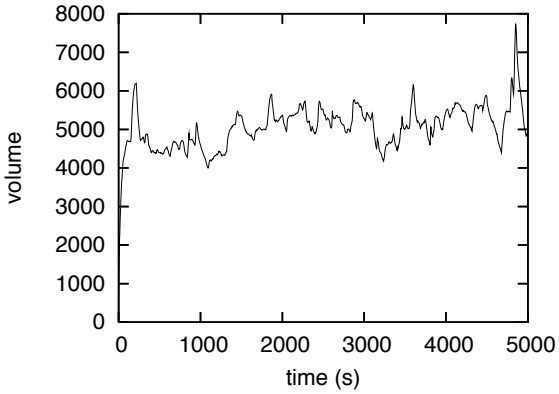
2.4.4 Multiple Moving Targets

We consider a scenario with multiple moving targets. Initially, targets which move to the random direction at the random speed from 0 to 8 km/h are located at randomly chosen nodes. At every control timing, a target changes the speed and direction with the probability of 0.005. The control interval is set at 2 seconds. At every $180/V$ control intervals, a moving target migrates to a neighboring node in the moving direction. Here, $180/V$ is the time required for a target moving at the speed of V km/h travels 100 meters. The discrete step Δt is set at 2.0.

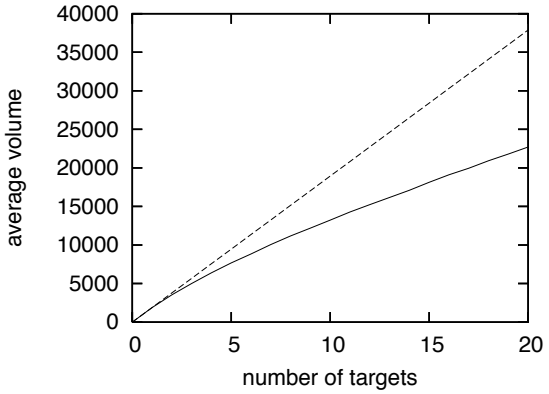
Figure 2.12(a) depicts the transition of the total volume of the whole region with three moving targets. Depending on the distance among targets and their speed and direction, the total volume dynamically changes. One reason that the total volume exceeds 6000, i.e. the triple of the volume of a single target, is that the high concentration of activator sometimes remains behind a fast moving target. The other reason is that suppression of stimulus E is too slow for closely located targets moving at the speed of more than 6 km/h. We need to accelerate pattern adaptation, but it is one of future work.

Figures 2.12(b) and 2.12(c) show the relationship among the number of targets and the average and maximum volume of generated patterns, respectively. The dashed lines stand for the product of the number of nodes and the value for one target. As shown in the figures, our mechanism suppresses the traffic volume much lower by reaction-diffusion based control.

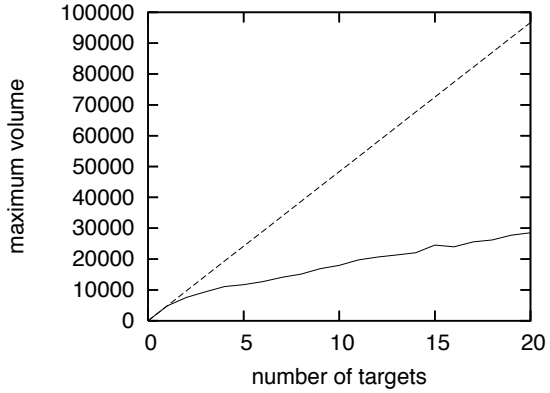
Although results are not shown, we also conducted simulation experiments on a large-size network, where 2500 nodes are arranged in a 50×50 grid with separation of 100 meters. We verified that every node appropriately adjusts the video coding rate to monitor moving targets.



(a) Transition of volume with three moving targets



(b) Average volume against number of moving targets



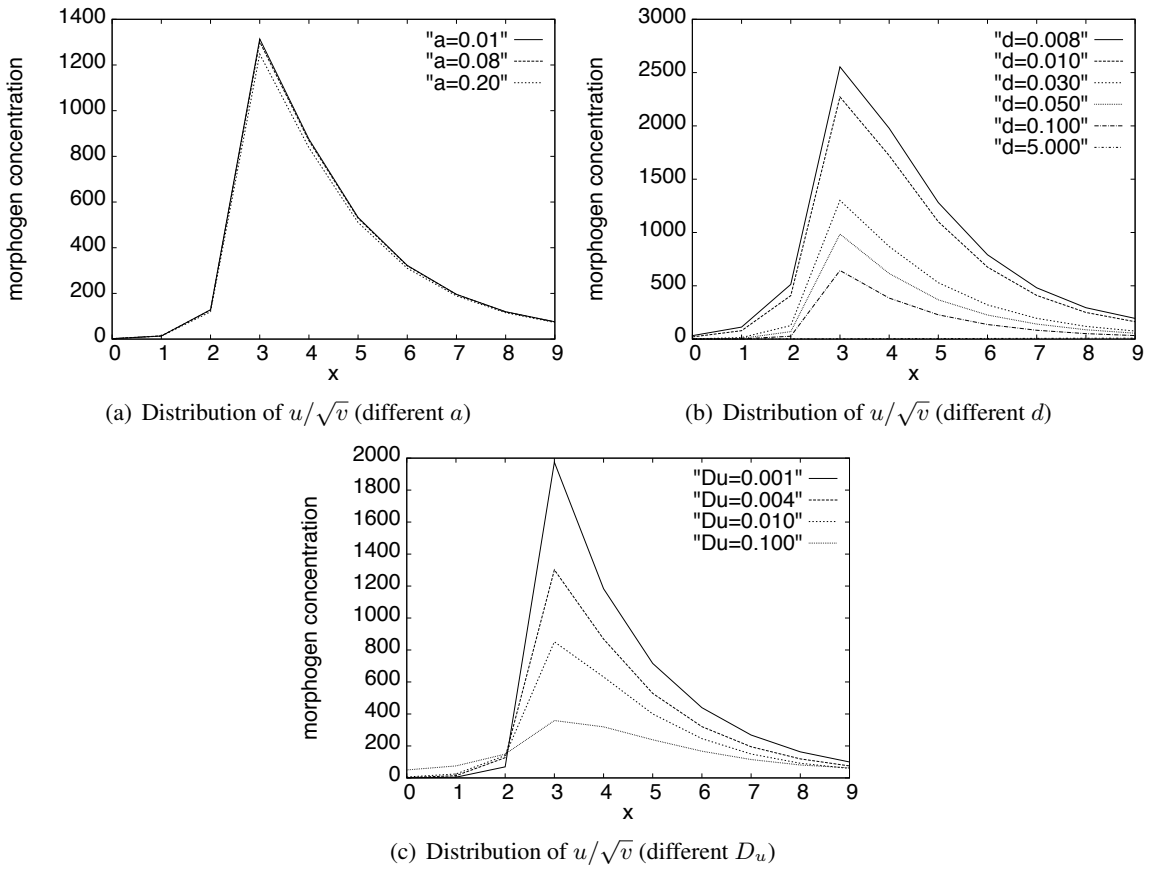
(c) Maximum volume against number of moving targets

Figure 2.12: Simulation results of multiple moving targets

2.4.5 Influence of Parameter Setting

To see the influence of parameter setting on generated patterns, we conducted experiments on a network of 10×10 grid layout where a moving target is located at $(3, 5)$ with the speed of 5 km/h heading to the right direction.

When we change a in Eq. (2.3), the distribution of $u/\sqrt{v}$ on $y = 5$ does not change much as shown in Fig. 2.13(a). Although not shown in figures, pattern generation is insensitive to setting of b, c, e, h, and f. On the contrary, d, g, D_u, and D_v in Eq. (2.2) affect generated pattern very much. Figure 2.13(b) shows that a larger d decreases the height of $u/\sqrt{v}$, by decomposing the activator at the higher rate. Parameter g has the same effect. The larger diffusion rates D_u also



(a) Distribution of $u/\sqrt{v}$ (different a)

(b) Distribution of $u/\sqrt{v}$ (different d)

(c) Distribution of $u/\sqrt{v}$ (different D_u)

Figure 2.13: Simulation results of one moving target for different parameter values

decreases the activator concentration by diffusing the activator mode to neighbor nodes as shown in Fig. 2.13(c). Since $u/\sqrt{v}$ corresponds to the coding rate and further to the local channel capacity, these parameters should be carefully chosen in accordance with the targeted coding rate through preliminary numerical experiments.

These results also imply that we can control the shape and volume of pattern by dynamically changing parameters, i.e. d, g, D_u, and D_v, in accordance with the available wireless capacity and the velocity of a target. However, for this purpose all nodes have to use the same parameters and thus adaptation of parameters introduces considerable communication overhead in disseminating a new set of parameters. On the contrary, as discussed in Section 2.3, the amount E of stimulus and the attenuation coefficient A affects the shape and volume of pattern as well. Since adaptation of

these parameters, i.e. E and A to dynamically changing operational conditions can be achieved at a node independently from the others, we plan to regulate parameters E and A, while using the same and fixed set of reaction-diffusion parameters listed in Table 2.1 on all nodes. It further relaxes the sensitivity of pattern generation to parameter setting, although it requires preliminary experiments to obtain Table 2.2. We need an algorithm to dynamically adjust Table 2.2 to fit to the actual operational environment, but it remains one of future work.

2.5 Practical Experiment

To verify our proposal in an actual environment, we implemented the mechanism and conducted practical experiments. In this section, we give an overview of our implemented system and show results of experiments.

2.5.1 Video Coding Rate Control

For the purpose of local logging of experimental data at each node, we used laptop PCs equipped with an IEEE 1394 camera and IEEE 802.11g wireless interface for a camera sensor node. Figure 2.14 shows how messages and video data are processed in a node. A node maintains information including the morphogen concentrations and stimuli on neighbor nodes and itself. At regular intervals, a node first checks whether it has a target in the observation area or not. If a target exists, a node determines the corresponding stimulus information, i.e. E and A, in accordance with the velocity of the target as in Table 2.2. Initially, the stimulus E is set at its maximum limit. Next, a node calculates the reaction-diffusion equation (2.8) based on the morphogen concentrations of itself and neighbors and the stimuli. The computational complexity is not high where Eq. (2.8) is simple floating point arithmetic and can even be transformed to integer arithmetic. Then, it determines the coding rate based on the value $u/\sqrt{v}$ as in Table 2.3. Finally, it broadcasts a message containing its morphogen concentrations, the stimuli information, and NIP, by using UDP/IP broadcasting. The payload size is 1649 bytes containing logging information. Since the communication and computational overhead at a node are not too high, our proposal can be implemented on low-cost visual sensor nodes with the capability of wireless communication, arithmetic computation, and video rate

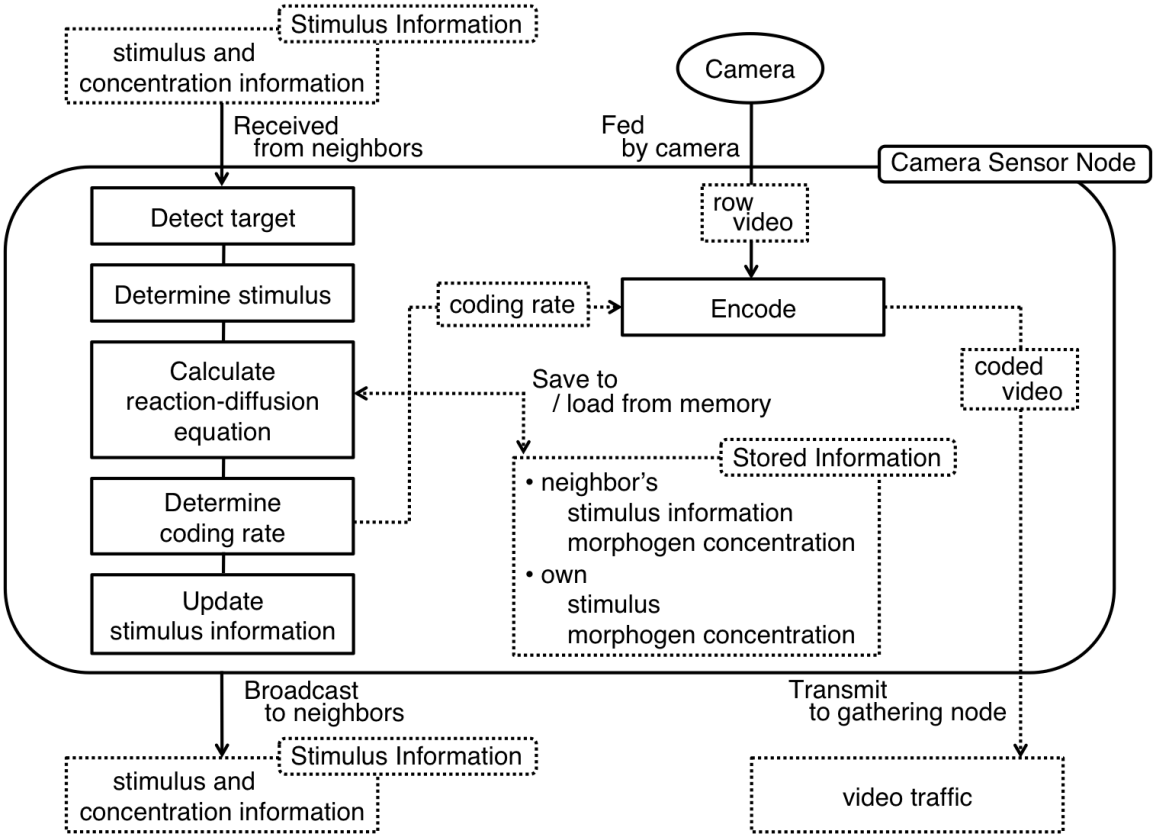


Figure 2.14: Flow of information and data in a node

control.

By using the latest coding rate, a node generates MPEG-2 video data from raw video frames fed by a camera in YUV422 format. As an encoder, we used a software encoder called `mpeg2vidcodec_v12` developed by MSSG (MPEG Software Simulation Group) [44] and modified it so that the coding rate could be dynamically changed during coding a MPEG-2 stream. Video data are sent to a base station by using UDP/IP unicast communication. The payload of one video data packet amounts to 1092 bytes. Although we adopt MPEG-2 in our experiments for the purpose of proof of concept, other video coding algorithms such as MPEG-4 and other rate control mechanisms [45] such as frame dropping or temporal resolution adaptation can be applied to our proposal and we do not limit to our experimental setting. By choosing more sophisticated algorithms and mechanisms together with error concealment techniques in the deployment phase, a video surveillance system with

Table 2.3: Mapping from concentrations of morphogens to coding rate

$u/\sqrt{v}$	coding rate
$0 < u/\sqrt{v} \geq 5000$	0.75 Mbps
$5000 < u/\sqrt{v} \geq 10000$	1 Mbps
$10000 < u/\sqrt{v}$	2 Mbps

the higher video quality and the higher tolerance to bit errors can be realized.

2.5.2 Object Detection

Although some commercially available cameras are capable of object and motion detection, in our implementation, we used a simple mechanism for the purpose of preliminary experiments and easier control. The accuracy and effectiveness of object detection are out of scope of our research.

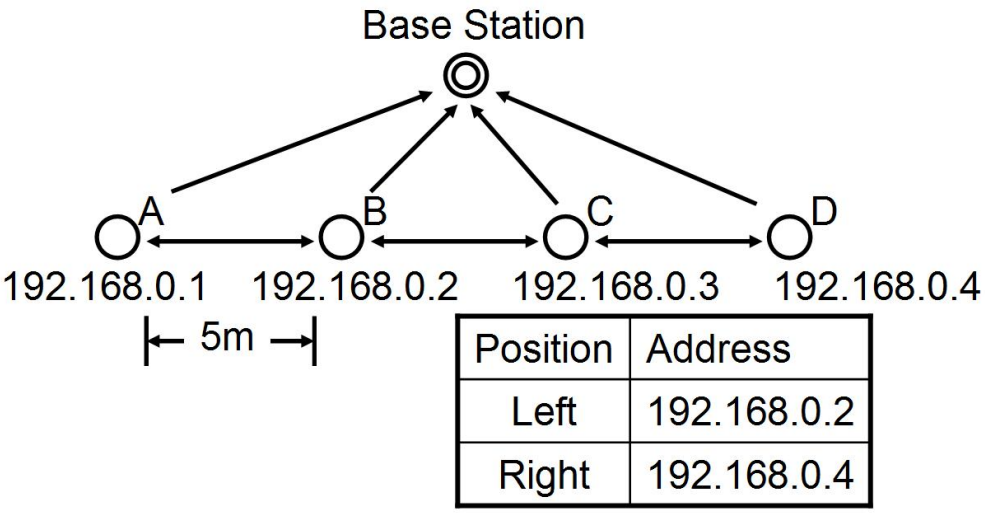
A node detects the existence and movement of an object by comparing the difference in luminance between two successive video frames. The luminance difference is derived for each pixel (i, j) by subtracting the luminance $f(i, j)$ in the current frame from the luminance $F(i, j)$ in the preceding frame. If the number of pixels whose absolute value of luminance difference is above the threshold x exceeds the threshold y, a node considers that there is a moving object in the observation area. The moving direction of the target is estimated by comparing the average coordinates of changed pixels in the successive several frames. Since estimation of the moving speed requires the complicated image processing such as estimation of the object size and its distance, we did not implement the speed detection mechanism and used the fixed value.

2.5.3 Experimental Setting

In simulation experiments, we verified that each node could adjust their coding rate properly and the total amount of video traffic was suppressed by applying our proposal even for cases of multiple targets. However, due to time and facility limitation, the experiments in this chapter were with only one target and four camera sensor nodes.

We arranged four camera sensor nodes 5 m apart in a line as illustrated in Fig. 2.15. A base station was located between nodes B and C and apart from the line. They were connected by IEEE

Figure 2.15: Node layout in experimental system



802.11 IBSS (Independent Basic Service Set) mode. All nodes belonged to the same IP subnet by having the same network address. Although all nodes were within the range of wireless communication, we manually configured them so that a camera sensor node could exchange messages only with a base station and its adjacent neighbors by ignoring messages from distant nodes. The control interval was set at 0.25 second. Thus, a node consumes 53 Kbps for control messages. All nodes operated asynchronously, where they broadcast messages, calculated the reaction-diffusion equation, and adjusted the coding rate at different timing. Other parameters used for the experiments are summarized in Table 2.1.

2.5.4 Evaluation Measure

We use packet loss rate and PSNR (Peak Signal to Noise Ratio) as measures of evaluation. The packet loss rate is defined as the ratio of the number of video data packets which are not received at a base station to the number of video data packets sent from camera sensor nodes. The PSNR is

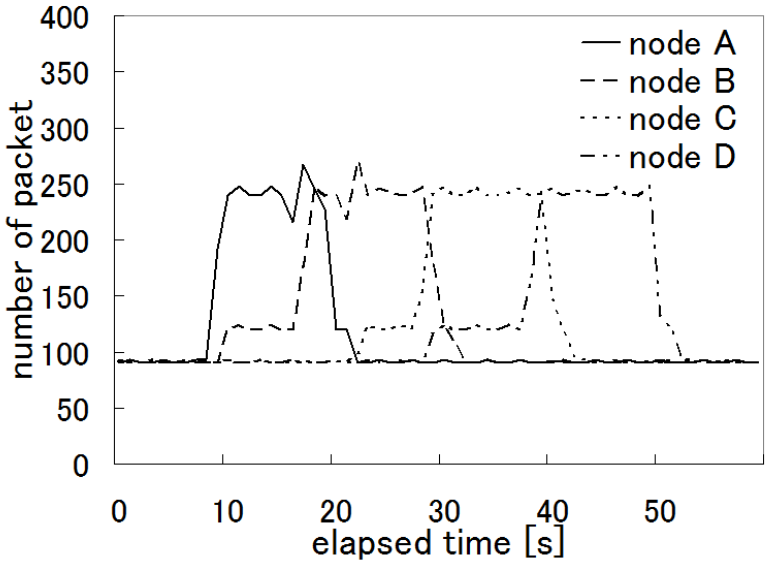


Figure 2.16: The number of emitted packets with our proposal

derived by the following equation.

$$PSNR = 20 \log 10(\frac{255}{\sqrt{MSE}}) \tag{2.12}$$

MSE represents the mean square error. MSE of an image of $m \times n$ pixels is calculated the by following equation.

$$MSE = \frac{\sum (f(i,j) - F(i,j))^2}{mn} \tag{2.13}$$

where $F(i,j)$ and $f(i,j)$ are the luminance at the pixel (i,j) of an original image and that of a comparison image, respectively.

2.5.5 Results and Discussion

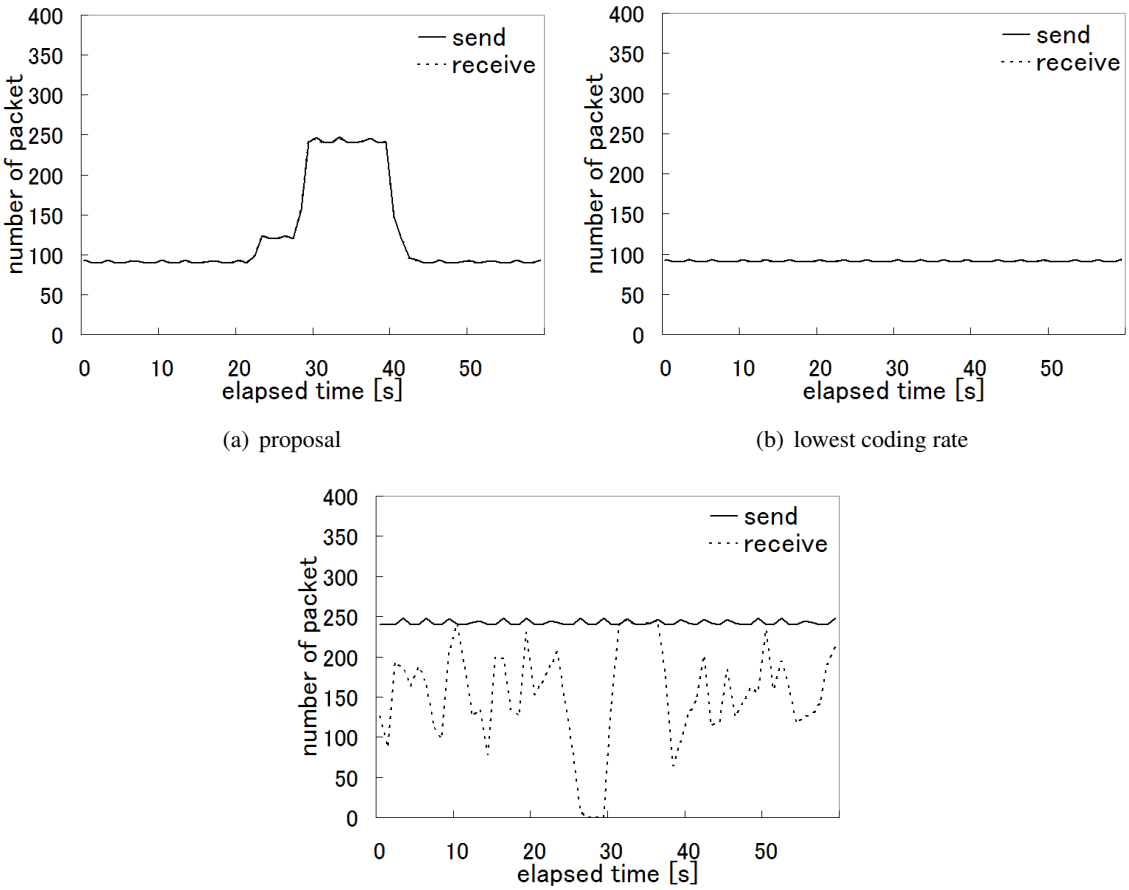
We first evaluated the performance of our implemented system and found that, to accomplish the detection, calculation, broadcasting, and coding in a real-time fashion, the spatial resolution and the frame rate should be kept as low as 320×240 pixels and 8 frames/sec, respectively. This results in the video rate of 1 Mbps at maximum and cannot cause congestion in the wireless network of only

four nodes. Therefore, in the following, we use dummy traffic equivalent to video data of 720×480 pixels and 30 frames/sec in size. On the base station, loss of video data packets was monitored and recorded, from which received video data were generated by using the video coding rate determined in experiments.

Figure 2.16 illustrates the time variation in the number of packets emitted by the camera sensor nodes per second when our proposal was applied. As the target walked at the speed of about 1.8 km/h from the left (node A) to the right (node D), it moved from the observation area of one node to another in about 10 seconds. This corresponds to the time variation in the video traffic shown in Fig. 2.16, where the duration that a node sent video data packets at the rate of about 250 packets/sec, i.e. 2 Mbps is about 10 seconds. A node in the direction of the target movement first raised its coding rate from 0.75 Mbps to 1 Mbps, and then set it at 2 Mbps on having the target in its observation area, and finally returned to its normal rate of 0.75 Mbps when the target disappeared from the observation area.

Next, in Fig. 2.17, we compare three alternatives focusing on the node C, which was located near the base station and thus affected by other traffic. As we do not see any visible gap between the number of packets sent by the node C and the number of packets received at the base station for the node C in Fig. 2.17(a), the packet loss rate was only 0.01 $\%$ for the node C during the whole experiment. Figure 2.17(b) shows the case of adopting the lowest coding rate, i.e. 0.75 Mbps, hesitating to generate the high quality video and introduce much video traffic into the network. In this case, the packet loss rate was the lowest and 0 $\%$. Finally, the packet loss rate of the case where all nodes aggressively used the highest coding rate is shown in Fig. 2.17(c). As shown in the figure, the considerable number of packets were lost and the loss rate was about 36.7 $\%$ for overloading the wireless network.

Figure 2.18 shows how the whole network was loaded. The figure illustrates the time variation in the total number of packets sent by all sensor nodes and the total number of packets the base station received. The number of received packets in Fig. 2.18(c) indicates the capacity of the wireless network. It is approximately 4.65 Mbps on average. It is almost equivalent to the total video traffic when one node uses the coding rate 2 Mbps, another node does that of 1 Mbps, and the other two nodes keep the lowest at 0.75 Mbps. In our proposal, the camera sensor nodes autonomously



(a) proposal

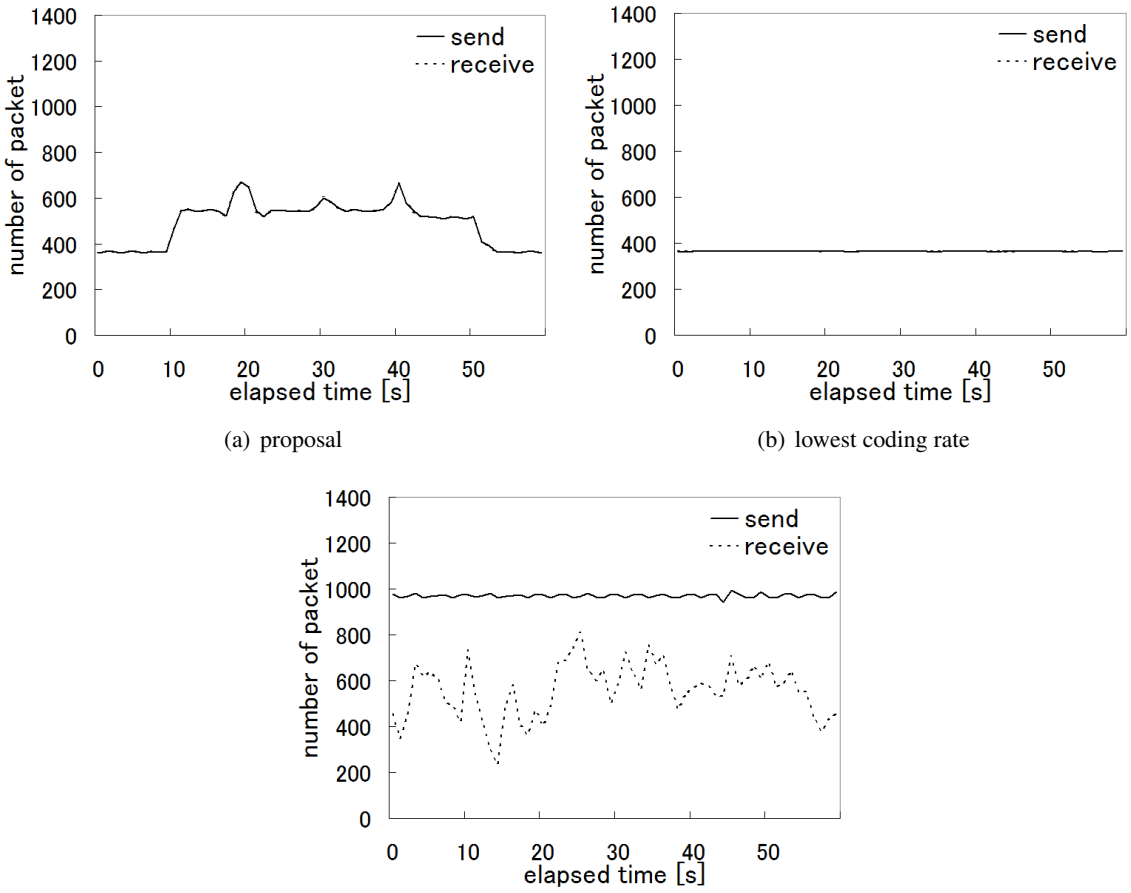
(b) lowest coding rate

(c) highest coding rate

Figure 2.17: The number of sent and received packets (Node C)

choose the optimal coding rate allocation in accordance with the location of the target by adopting the reaction-diffusion model. The average transmission rate of video data per node is about 1.03 Mbps with our proposal. The resultant packet loss rates are 0.24 %, 0.02 %, and 42.5 % in Figs. 2.18(a), 2.18(b), and 2.18(c), respectively. The packet loss rates for nodes having the target in observation area are 0.03 %, 0 %, and 38.8 %, respectively.

The high packet loss rate severely affects the perceived video quality. Figure 2.19 shows the time variation in the PSNR value for video data sent by the node C and that received at the base station for the node C. The PSNR is derived against original video frames captured by a camera. Since the packet loss rate was too high to decode the received video data when the highest coding



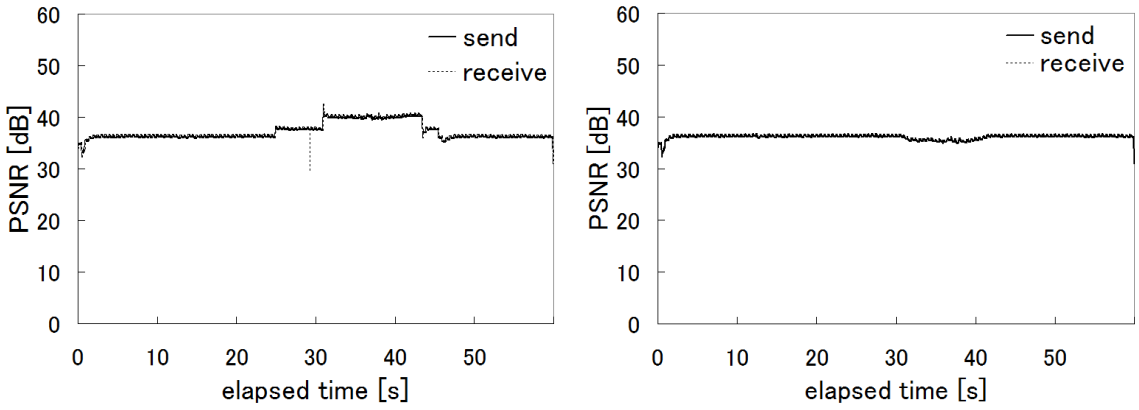
(a) proposal

(b) lowest coding rate

(c) highest coding rate

Figure 2.18: The total number of sent and received packets

rate was always used, the PSNR values are only shown for the other two cases. In the case that our proposal was adopted, the node C adjusted the video coding rate in accordance with the target movement, as we see the increase in the PSNR to 40.0 dB at about 31 second. The reason for the instantaneous decrease of PSNR at about 29 second was for loss of packets containing the header information. This can be recovered by using FEC or we can avoid loss of header information by prioritizing important packets to the others. On the contrary, although such spike-like variation does not appear in Fig. 2.19(b), the PSNR was around only 36.7 dB. Especially when the node C had the target in its observation area, the PSNR slightly decreased to 35.5 dB for the complexity and activity of captured images. Figure 2.20 compares the perceived video images from the node C



(a) proposal

(b) lowest coding rate

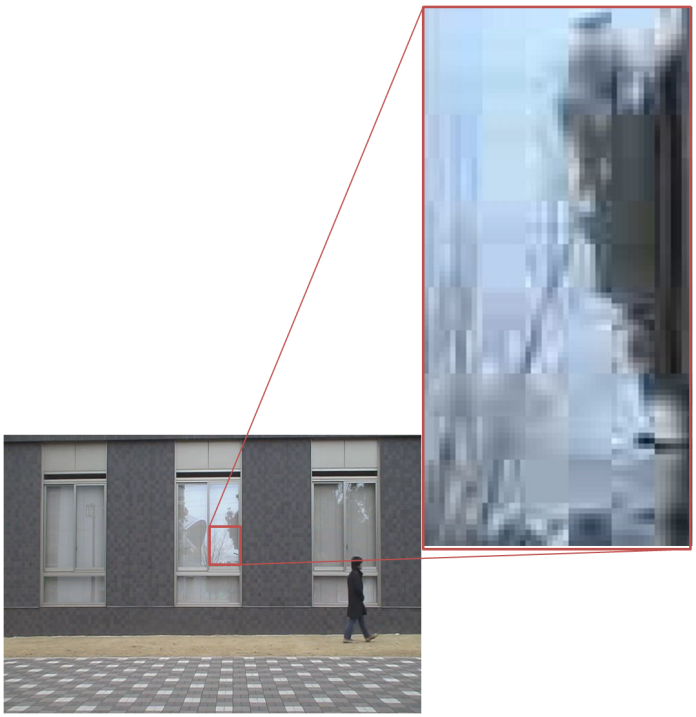
(c) highest coding rate

Figure 2.19: PSNR of sent and received video data (Node C)

at 40 second for the cases of our proposal and lowest coding rate. Although the difference is not obvious, we see a finer image with our proposal.

In conclusion, with our mechanism, camera sensor nodes autonomously adjusted video coding rate to avoid congestion of the wireless network in order to satisfy application requirements on the perceived video, where a user could monitor a target at the high quality description on a display.

(a) proposal



(b) lowest coding rate

Figure 2.20: Perceived video image sent from node C at 40 [s]

2.6 Conclusion

In this chapter, we proposed a reaction-diffusion based autonomous control mechanism for camera sensor networks. In our mechanism, nodes periodically exchange information about the morphogen concentrations, calculate the reaction-diffusion equation, and adjust the video coding rate. By setting the stimulus at a node detecting a target and diffusing the stimulus, the video coding rate becomes high at a node with a target and nodes in the moving direction. Through simulation and practical experiments, we verified the effectiveness of our proposal. We showed that network congestion was avoided and the quality of video data from a camera having target was high.

Chapter 3

An Inter-networking Mechanism with Stepwise Synchronization for Wireless Sensor Networks

To realize the ambient information society, multiple wireless networks deployed in the region and devices carried by users are required to cooperate with each other. Since duty cycles and operational frequencies are different among networks, we need a mechanism to allow networks to efficiently exchange messages. For this purpose, we propose a novel inter-networking mechanism where two networks are synchronized with each other in a moderate manner, which we call stepwise synchronization. With our proposal, to bridge the gap between intrinsic operational frequencies, nodes near the border of networks adjust their operational frequencies in a stepwise fashion based on the pulse-coupled oscillator model as a fundamental theory of synchronization. Through simulation experiments, we show that the communication delay and the energy consumption of border nodes are reduced, which enables wireless sensor networks to communicate longer with each other.

3.1 Introduction

The ambient information society is the concept and framework where intelligent environment detects, reasons, and satisfies overt and potential demands of people without their interaction [1–3]. In the ambient information society, people do not need to be aware of existence of networked information devices embedded in the environment. They do not need to intentionally access a network to control the environment to make it comfortable and satisfy their demands. Instead, the embedded network controls the environment and provides personalized information services to a user taking into account time, place, occasion, and person.

To realize the ambient information society, networks deployed and operating in the same environment must cooperate with each other in exchanging information, sharing information, and controlling each other. For example, a person has a wireless body area network [7] which consists of vital sensors, accelerometers, PDA, and other devices. On the other hand, a room has embedded wired and wireless networks which consist of sensors and actuators for environmental control. Intelligent home appliances also constitute embedded networks. When the person enters the room, those networks should cooperate with each other for smart environmental control. However, in general, those devices organize different and independent networks operating on different control policies. Therefore, so that the room provides the person with a comfortable environment, we need a mechanism for different networks to smoothly and dynamically connect and share their information. However, it is not a trivial task.

In general, wireless sensor networks adopt a sleep scheduling or duty cycling mechanism to save energy. Operational frequencies, that is, frequencies that they wake up and resume operation, are different among networks depending on application's requirement and characteristics of devices. For example, an air conditioner would obtain and use the temperature information every minute to adjust its thermostat. On the other hand, devices to detect locations of people have to report their detection result very frequently at an order of seconds. When they want to exchange information among them for intelligent control of room temperature to intensively regulate the temperature around a person in the room, a node belonging to the location detection system has to stay active in order to wait for a node belonging to the thermal management system to wake up in transmitting a

message. Even when an energy-efficient MAC protocol such as S-MAC [27] and X-MAC [28] is used, such communication consumes the substantial energy at the former node and it would bring danger of energy depletion.

There are several proposals on dynamic composition of multiple networks [46, 47]. In [46], they consider a mechanism for overlay networks to dynamically compose a hierarchical structure by two types of composition schemes, i.e. absorption and gatewaying. In [47], cooperation between wireless networks is accomplished by organizing an overlay network by connecting gateway nodes belonging to different wireless networks. Although they can be applied to ambient information networking to some extent, they have a major problem that they do not take into account the difference in operational policies, more specifically, operational frequencies of different wireless sensor networks.

ZigBee [35], a standard protocol for wireless sensor networks, also provides interconnecting schemes such as PAN bridge, PAN marge and Peer-to-Peer, which enable a ZigBee Personal Area Network (PAN) to communicate with other PANs [48]. However, they also do not take into account the difference of the operational frequency. Although a bridge node can mediate communication among PANs with different operational frequencies, the bridge node consumes much energy and it would shorten the lifetime of PANs.

To address the problem, we propose stepwise synchronization between wireless sensor networks for smooth and moderate inter-networking, where sensor nodes located near the border of two networks adjust their operational frequencies to bridge the gap in their intrinsic operational frequencies. Since only nodes near the border change their operational frequency, the remaining nodes can keep their frequency and thus energy consumption in inter-networking can be reduced. The stepwise synchronization is self-organized based on a nonlinear mathematical model of synchronization of oscillators, called the pulse-coupled oscillator (PCO) model [29]. The PCO model describes emergence of synchronization in a group oscillators with different frequencies by mutual interactions through stimuli. By adopting the PCO model to scheduling, operational frequencies of nodes can be appropriately adjusted without any centralized control in wireless sensor networks [49–52]. In our mechanism, we strengthen the degree of entrainment at border nodes to intensively shift the operational frequency toward that of the other network while the degree of entrainment is weakened

as the distance to the border increase. As a result, the operational frequencies of nodes near the border are adjusted to somewhere between the original operational frequencies of wireless sensor networks. Through simulation experiments, we evaluate the effectiveness, robustness and adaptivity of our proposal and the influence of parameter setting.

The rest of this chapter is organized as follow. First in section 3.2, we explain the pulse-coupled oscillator model. Next in section 3.3, we describe the details of our proposal. In section 3.4, we show and discuss results of our simulation experiments. Finally, we conclude the chapter in section 3.5.

3.2 Pulse-Coupled Oscillator Model and Synchronization

A pulse-coupled oscillator model is a mathematical model which explains synchronized flashing of a group of fireflies [29]. It is considered that a firefly maintains a biological timer, based on which it intermittently flashes. The flashing frequency depends on its intrinsic timer frequency, which could be different among individuals. However, when fireflies form a group, they begin to flash in synchrony. A mechanism of biological synchronization is explained as follow. When a firefly observes a flash of another firefly, it is stimulated and its timer advances by a small amount. Because of nonlinearity in timer or stimulus, by repeatedly stimulating each other, their timers begin to expire synchronously, then flash at the same time. Among PCO models [29, 53, 54], in this chapter we use the model proposed in [29].

In the PCO model [29], oscillator i maintains phase ϕ_i $(0 \leq \phi_i \leq 1)$ of a timer and state x_i $(0 \leq x_i \leq 1)$ given by a function of phase. The dynamics of phase ϕ_i is determined by the following differential equation.

$$\frac{d\phi_i}{dt} = F_i \tag{3.1}$$

where F_i $(F_i > 0)$ stands for the intrinsic timer frequency of oscillator i. State x_i is determined

from phase ϕ_i by the following monotonically increasing nonlinear function,

$$x_i = \frac{1}{b}\ln[1 + (e^b - 1)\phi_i] \tag{3.2}$$

where b ($b > 0$) is a dissipation parameter that dominates the rate of synchronization.

When phase ϕ_i and state x_i reach 1, oscillator i fires and both phase ϕ_i and state x_i go back to 0. When an oscillator fires, the oscillator stimulates oscillators that are coupled with the firing oscillator. If oscillator j is stimulated by oscillator i at time t, oscillator j increases its state x_j by a small amount ϵ and phase ϕ_j changes accordingly as

$$x_j(t^+) = B(x_j(t) + \epsilon), \tag{3.3}$$

where

$$B(x) = \begin{cases} x \ \ (0 \leq x \leq 1) \\ 0 \ \ (x < 0) \\ 1 \ \ (x > 1) \end{cases}$$

and

$$\phi_j(t^+) = \frac{e^{bx_j(t^+)} - 1}{e^b - 1} \tag{3.4}$$

When state $x_j(t^+)$ and phase $\phi_j(t^+)$ reach 1 by being stimulated, oscillator j also fires. Once oscillator j fires by being stimulated by oscillator i, oscillator j continually fires by being stimulated by oscillator i, if F_i is greater than or equal to F_j. If F_i is less than F_j, oscillator i continually fires by being stimulated by oscillator j. At this time, oscillators i and j are considered synchronized. To avoid overshoot and instability, an oscillator is not stimulated by two or more oscillators at the same time, and an oscillator is not stimulated at the time when it fires.

Figure 3.1 shows how dissipation parameter b affects the relationship between state x and phase ϕ, where b is changed from 1.0 to 7.0. As can be seen in Fig. 3.1, when b is set at a small value, e.g. 1.0, the amount of phase shift on receiving a stimulus is almost the same despite the timing that the

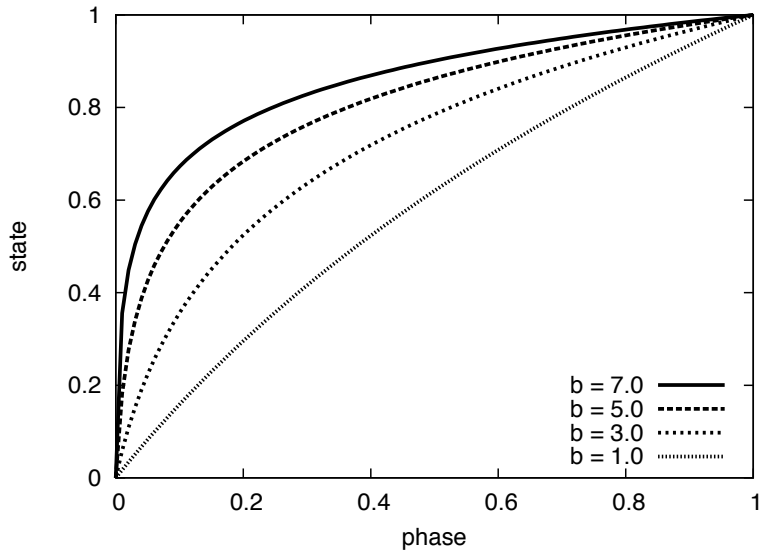
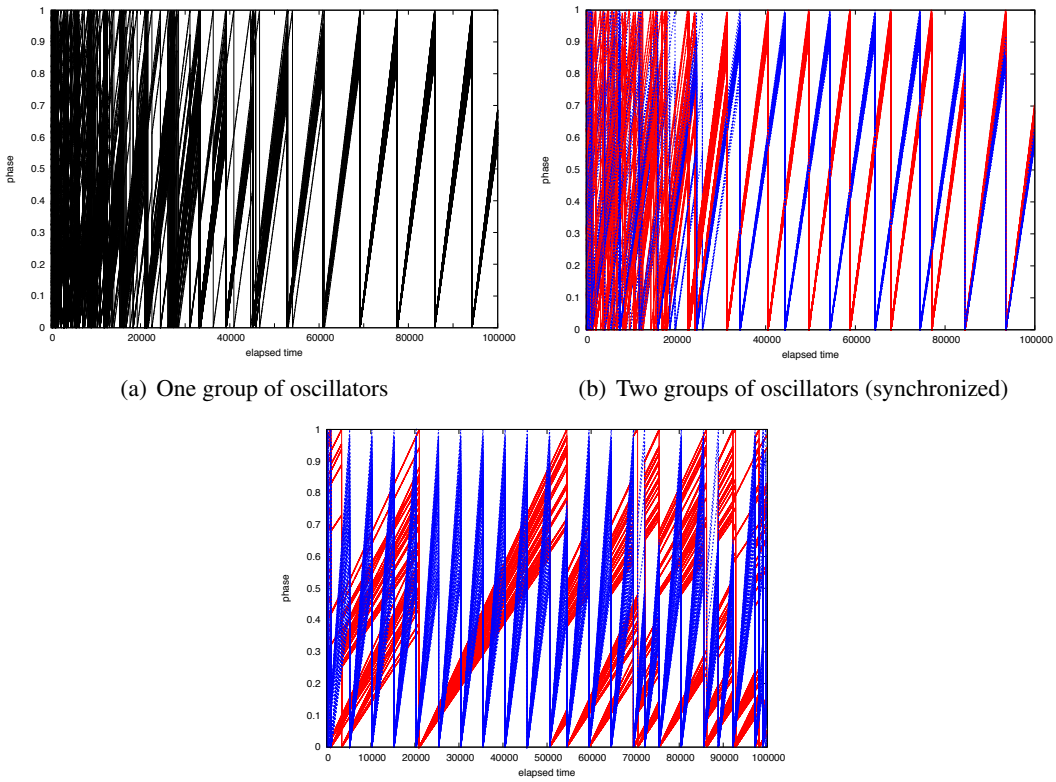


Figure 3.1: Effect of changing b on relationship between state and phase

stimulus is received. On the other hand, in the case that b is large, e.g. 7.0, a stimulated oscillator changes its phase by a large amount when its state value is large.

In Fig. 3.2, we show results of numerical analysis for three different scenarios. First, Fig. 3.2(a) shows phase transition of 100 oscillators arranged in a 10×10 grid. Intrinsic frequency of each oscillator is randomly chosen within the range of [0.9, 1.1]. Initial phase ϕ is also chosen at random. An oscillator is coupled with four neighbors located in up, right, down, and left of the oscillator and stimuli are never lost and always received by the four neighbors. There is no delay in stimuli propagation and phase and state change. b and ϵ are set at 3.0 and 0.1, respectively. At first, phases are different among oscillators. The increasing rates of phase are also different among oscillators due to the difference of intrinsic frequency. As time passes, timings of firing gradually get closer by stimulating each other. Finally, all oscillators become synchronized and fire at the same time.

Then, we show how two groups of oscillators whose intrinsic frequencies are different get synchronized to the same operational frequency. As in the above example, 100 oscillator are arranged in a 10×10 grid. The initial phase is chosen at random, and b and ϵ are set at 3.0 and 0.1, respectively.



(a) One group of oscillators

(b) Two groups of oscillators (synchronized)

(c) Two groups of oscillators (unsynchronized)

Figure 3.2: Phase transition in numerical analysis

Half of oscillators in the left forming a 5×10 grid set their intrinsic frequencies within the range of $[0.9, 1.0]$ at random, and they belong to group 1. The other oscillators set their intrinsic frequency within the range of $[1.0, 1.1]$ at random, and they belong to group 2. During the first four-fifths of simulation time, oscillators are not stimulated by oscillators belonging to the other group. That is, there is no coupling among oscillators of different groups. During the last one fifth of simulation time, oscillators at the border are coupled with each other so that they can stimulate each other. As can be seen in Fig. 3.2(b), during the first four-fifths, oscillators belonging to the same group identified by the same color get synchronized and oscillators belonging to different groups fire at different timings. As inter-group communication is allowed at time 80,000, two groups become integrated and all oscillators fire at the same time and same frequency. The operational frequency

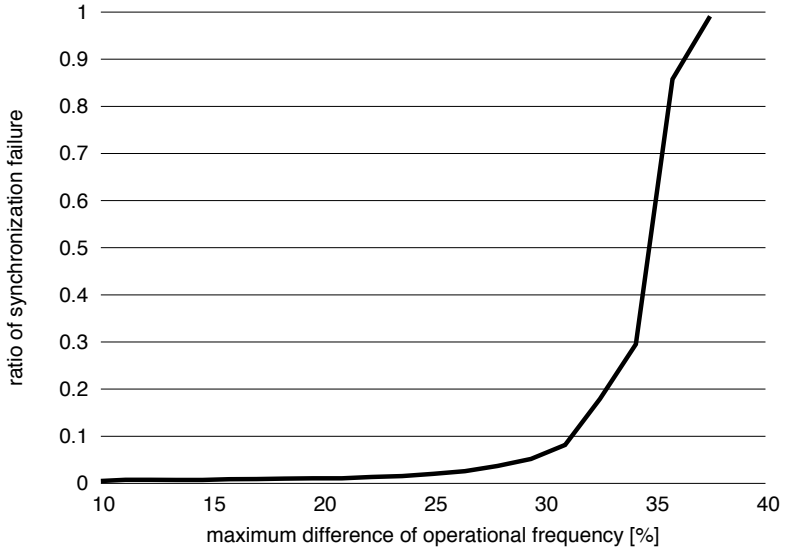


Figure 3.3: Ratio of synchronization failure against maximum difference in operational frequency

of the unified group is dominated by the highest frequency of intrinsic frequencies of all oscillators, i.e. about 1.1.

Finally, we show a case that two groups of oscillators cannot accomplish synchronization for a big difference in operational frequencies. From the above setting, we change the ranges of operational frequency to [0.2, 0.3] for group 1 and [1.0, 1.2] for group 2. An inter-group communication is allowed at time 50,000. An obtained result is shown in Fig. 3.2(c). As can be seen, the global synchronization cannot be accomplished and synchronization in each group is even lost by being disturbed by stimuli from the other group.

To see how difference in operational frequency affects synchronization, Fig. 3.3 illustrates the relationship between the difference and the ratio of synchronization failure. We distribute 100 oscillators randomly in a 100 $\times$ 100 region. Each oscillator is coupled with oscillators within a 25 radius. Stimuli are never lost and received by coupled oscillators with no delay. The difference in operational frequency on the x-axis is changed by adjusting the range of operational frequency, between which an intrinsic operational frequency of each oscillator is randomly chosen. More specifically, the upper limit is set at 1.1 and the lower limit is changed from 1.0 to 0.8. When the

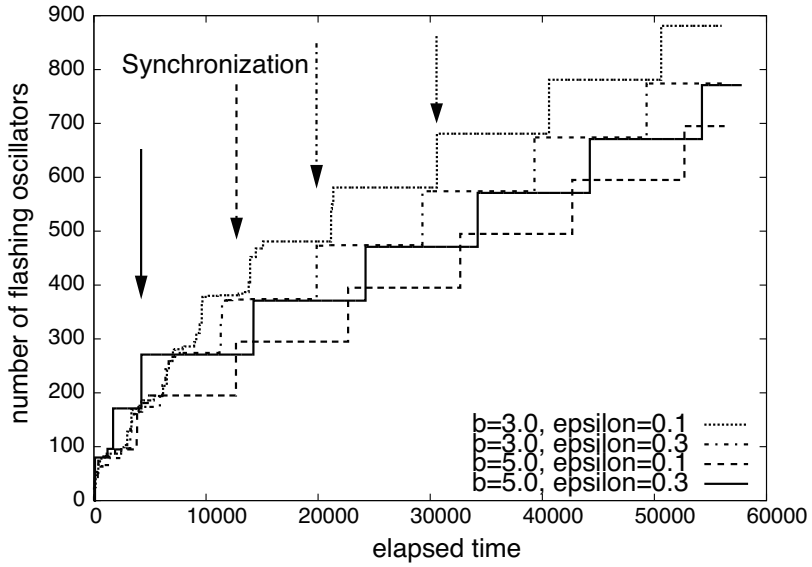


Figure 3.4: Cumulative number of flashing oscillators

range is $[1.0, 1.1]$, the difference is given as $(1.1 - 1.0)/1.0 = 0.1$. Initial phase is set at random, and b and ϵ are set at 3.0 and 0.1, respectively. We conducted 10,000 simulation runs for each of operational frequency settings. The y-axis shows the ratio that the global synchronization cannot be accomplished until the end of simulation run, i.e. 100,000 simulation time units, among 10,000 simulation runs. In Fig. 3.3, when the maximum difference in operational frequency is small, the global synchronization can be accomplished with high probability. However, when the maximum difference is greater than 30 %, the ratio of the synchronization failure suddenly increases. The result supports a need for a stepwise synchronization mechanism for wireless sensor networks adopting PCO-based synchronization mechanism [49] to be interconnected without too much burden on border nodes.

Not only operational frequencies but parameters of PCO also influence synchronization. In Fig. 3.4, we show how the cumulative number of flashing oscillators changes with different parameters b and ϵ against time. 100 oscillators are arranged in a 10×10 grid. Each oscillator is coupled with four neighbors. The operational frequency is identical among oscillators and their initial phase is

set at random. The height of stepwise increase in the number corresponds to the number of oscillators simultaneously flashing. As indicated by arrows, the time when a group of oscillators reach synchronization and begins to flash in synchrony is about 30,000 with "b=3.0, ϵ=0.1", about 20,000 with "b=3.0, ϵ=0.3", about 13,000 with "b=5.0, ϵ=0.1", and about 4,000 with "b=5.0, ϵ=0.3", respectively. When we adopt larger b and ϵ, the speed of synchronization apparently increases. This fact is a main source of our idea to achieve the stepwise synchronization. Much larger parameters further accelerate synchronization, but the stability deteriorates [49]. Although delay is not taken into account in these figures, it is known that the synchronization is accomplished in the environment with loss and delay of stimuli [55, 56].

3.3 Inter-networking Mechanism with Stepwise Synchronization

In this section, we propose an inter-networking mechanism with stepwise synchronization based on pulse-coupled oscillator model. First, we describe a targeted scenario and how we apply the PCO model to synchronization in wireless sensor networks, and then, we explain our stepwise synchronization-based inter-networking mechanism.

3.3.1 Target Scenario

In this chapter, we assume that two or more wireless sensor networks co-exist in the region, i.e. field or room. Each wireless sensor network is composed of nodes designated for a specific application, such as temperature control, health monitoring, home security, and so on. There can exist a server or central control unit, e.g. home server, which gather sensor information and control nodes. Each node is capable of wireless communication with other nodes within the radio range independently of wireless sensor networks to which they belong. Each node adopts duty cycling with which it switches between two states, i.e. active and sleep. An operational interval of node is the sum of active and sleep periods in one duty cycle. Operational intervals are identical among nodes belonging to a wireless sensor network and they are synchronized. For example, in a wireless sensor network to monitor temperature of a room at intervals of one minute, all nodes in the network simultaneously wake up every minute, obtain and report temperature, and go back to sleep. An operational

frequency of node is defined as a reciprocal of an operational interval. Synchronization of nodes is accomplished within a wireless sensor network by a PCO-based synchronization mechanism such as [49]. We consider a scenario where multiple wireless sensor networks with different operational frequencies exchange sensor information and control messages with each other for integrated and intelligent services as stated in Section 1.

3.3.2 PCO based Synchronization in Wireless Sensor Networks

In applying the PCO model to synchronization, a wireless sensor node corresponds to an oscillator. It stimulates neighbor nodes in the range of radio signals by broadcasting a message. A message is used for both of synchronization and data communication with such a mechanism where control messages for synchronization are embedded in messages for sensor data [57].

Node i maintains state x_i and phase ϕ_i as variables of a timer of frequency F_i and calculates its new state and phase at regular intervals, e.g. at the granularity of timer. When state x_i and phase ϕ_i reach 1, node i sets both state x_i and phase ϕ_i at 0 and tries to broadcast a message which informs neighbor nodes that the node fires. Since a wireless channel is the shared medium, there is possibility that broadcasting is delayed to wait for the channel to become available. However, from our previous experiments, the influence of delay on synchronization is negligible [55]. When a node receives a broadcast message, it is stimulated. The stimulated node, say node j, increments its state x_j by a small amount ϵ by Eq. (3.3) and calculates new phase ϕ_j^+ based on the new state x_j^+ by using Eq. (3.4). If the new state x_j^+ and new phase ϕ_j^+ reach 1, node j also fires and broadcasts a message. Since duty cycling is adopted on a node, only neighboring nodes that are awake can receive stimuli. Details of integration of duty cycling and the PCO model will be given later.

3.3.3 Overview

Now we propose a stepwise synchronization-based inter-networking mechanism. As an example, in Fig. 3.5, two wireless sensor networks with different operational frequencies are adjacent, and they attempt to cooperate. When we couple border nodes to let them stimulate each other, two wireless sensor networks will be unified to a single network with the operational frequency identical to

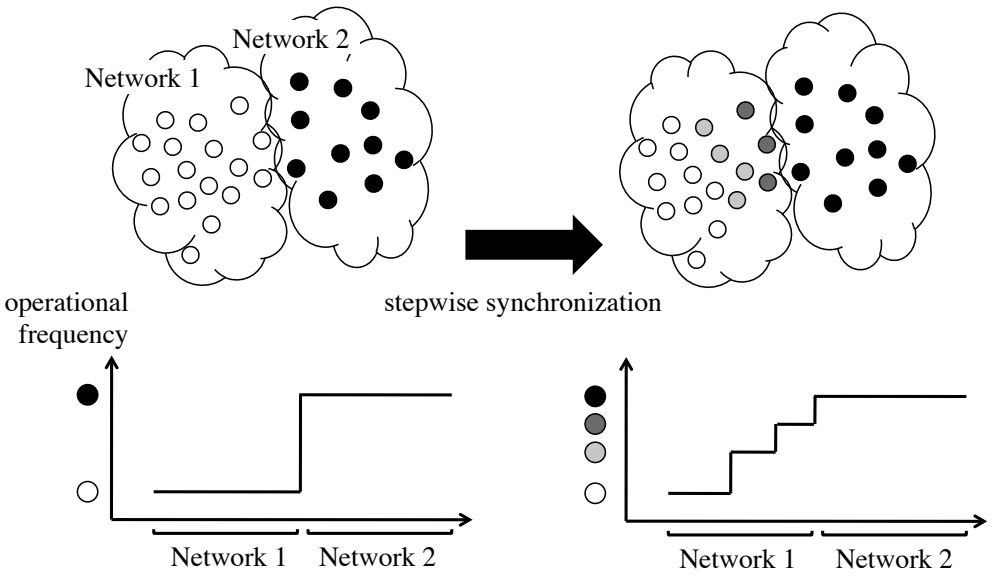


Figure 3.5: Stepwise synchronization

the faster one. However, it sacrifices too much energy of nodes belonging to a slower network by forcing them to wake up more frequently. Even if such unification and synchronization are allowed, there is limitation on the difference in operational frequencies in achieving the global synchronization. As shown in Figs. 3.2(c) and 3.3, it is hard for two networks whose operational frequencies are different by more than 35% to be fully synchronized. Therefore, we need a new mechanism to accomplish stepwise synchronization where only a part of network is involved in the synchronization and those networks with largely different operational frequencies can be synchronized. For this purpose, we adjust the degree of entrainment in accordance with the distance to the border. We focus on the fact that the dissipation b and the stimulus ϵ influence the degree of entrainment and the speed of synchronization (see Fig. 3.4).

In our proposal, we set larger values of b and ϵ at nodes located at the border of wireless sensor networks to strengthen entrainment and shift the operational frequency much. By receiving stimuli from the other network, nodes located at the border of wireless sensor networks actively changes their operational frequencies for larger parameters. Then smaller values are applied to nodes as the distance to the border becomes larger. Nodes distant from the border of wireless sensor networks

are also entrained by receiving stimuli from nodes located at the border, but the impact is smaller for smaller parameters and thus their operational frequencies stay rather closer to the original frequency. Consequently, we observe a stepwise change in operational frequencies around the border of two networks as illustrated in Fig. 3.5. Such stepwise synchronization can bridge the large gap in operational frequencies which cannot be overcome by the PCO model alone.

Figure 3.6 illustrates phase transition when two wireless sensor networks adopt our stepwise synchronization. Note that the intrinsic operational frequency of Network 1 is faster than that of Network 2. In Fig. 3.6, nodes located at the border of networks apply larger values to parameters b and ϵ, e.g. "$b = 4.0$, $\epsilon = 0.3$". Due to the larger values of b and ϵ, Node 1 increases its state largely when it receives a stimulus and the interval between two firing timings, i.e. the operational interval, become shorter. The operational interval of Node 2 also become shorter by receiving stimuli from Node 1 more frequently than the original operational frequency. However the degree of change in the operational interval is smaller than Node 1, because the entrainment is weaker for the smaller parameters b and ϵ. As the distance to the border increases, the strength of entrainment becomes smaller and thus the degree of change in the operational frequency becomes smaller.

3.3.4 Behavior of Node

Figure 3.7 shows the duty cycling in our proposal. Node i maintains state x_i and phase ϕ_i of a timer of frequency F_i. The phase automatically advances and the state accordingly changes independently of whether it is awake or sleeping. When state x_i and phase ϕ_i reach 1, node i sets both state x_i and phase ϕ_i at 0 and tries to broadcast a stimulus message. After the duration required to broadcast a stimulus message, node i goes to sleep for $T^n_{sleep} = T^{n-1}_i \times (1 - DutyRatio)$ independently whether it could successfully broadcast a stimulus message or not. T^{n-1}_i, called operational interval, is defined as the duration from the $n-1$-th firing timing to the n-th firing timing. $DutyRatio$ is the duty ratio parameter which is determined in advance. Although a sleeping node does not receive any stimulus message, there is a case that the state and phase occasionally reach 1. In such a case, a node wakes up to broadcast a stimulus message and after broadcasting it immediately goes to sleep. When T^n_{sleep} has passed, the node wakes up and becomes capable of sending and receiving

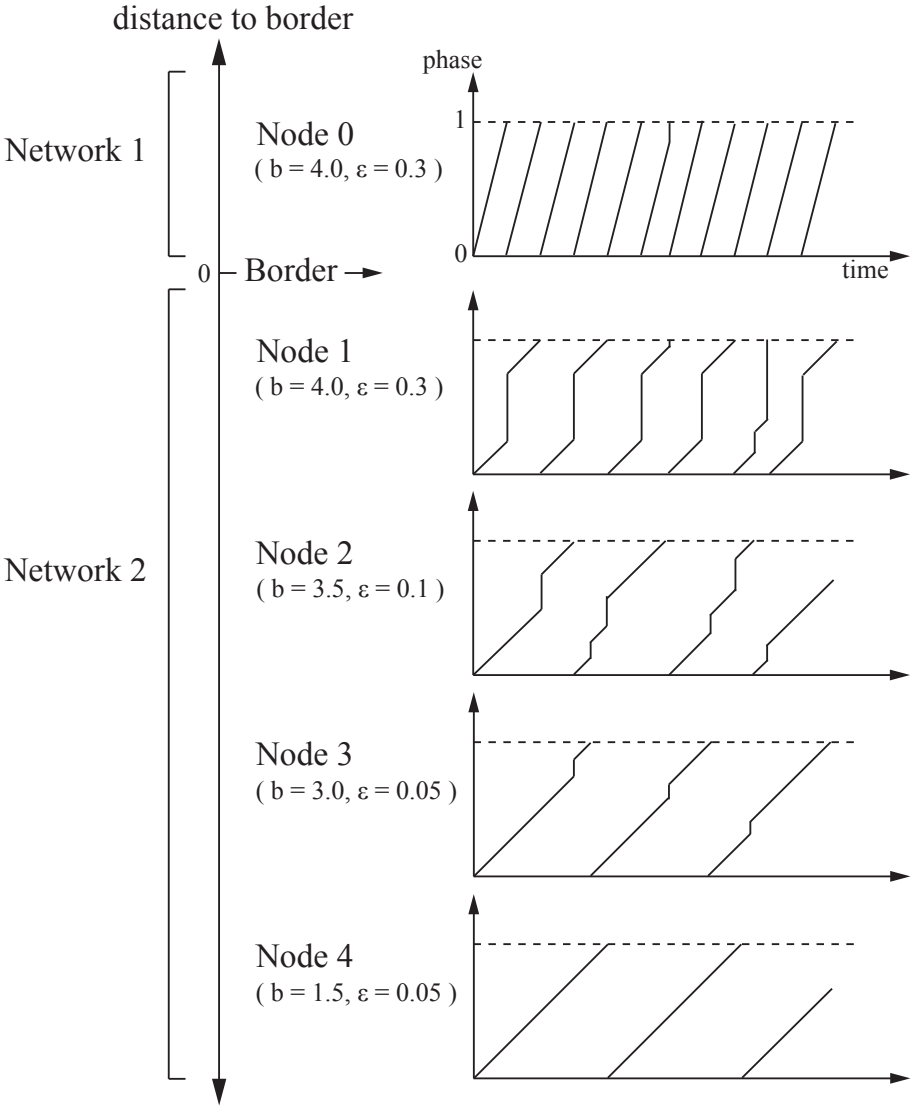


Figure 3.6: Phase transition in PCO based stepwise synchronization

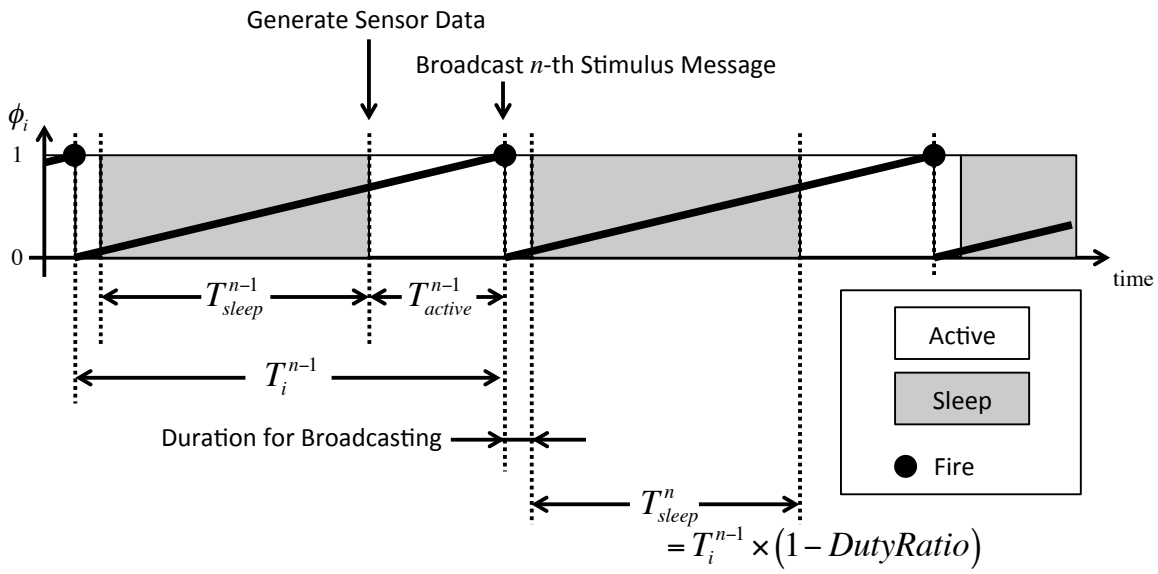


Figure 3.7: Duty cycling in proposal

messages for the duration of T_{active}^{n}.

If a node has received messages from other networks, it, i.e. a border node, adjusts b and ϵ at b_{max} and ϵ_{max}, where b_{max} and ϵ_{max} are maximum values of b and ϵ, respectively. A stimulus message emitted by a border node contains updated parameters b and ϵ, and attenuation coefficients A_b $(0 < A_b < 1)$ and A_ϵ $(0 < A_\epsilon < 1)$.

When node i receives a stimulus message in its active period, it calculates its new b and ϵ by following equations.

$$b_i^+ = \max(A_b \times b_{stim},\ b_{min}),$$
$$\epsilon_i^+ = \max(A_\epsilon \times \epsilon_{stim},\ \epsilon_{min})$$

where b_i^+ and ϵ_i^+ stand for new b and ϵ. b_{stim} and ϵ_{stim} stand for values of b and ϵ in the received stimulus message. b_{min} and ϵ_{min} are minimum values of b and ϵ. When a node receives two or more stimulus messages that have different values of b and ϵ, it uses the largest values among them. Then, node i calculates its new state and phase by Eqs. (3.3) and (3.4). Finally, the updated b_i^+ and

Table 3.1: Parameter setting

PCO	
parameter	value
b_{max}	3.0
ϵ_{max}	0.1
b_{min}	1.0
ϵ_{min}	0.02
A_b	0.7
A_ϵ	0.4

X-MAC	
parameter	value [ms]
S_{pre}	1.0
S_{ack}	1.0
S_{stim}	1.0
S_{data}	3.0
R_s	200
R_l	20

ϵ_i^+, A_b and A_ϵ are embedded as b_{stim}, ϵ_{stim}, A_b, and A_ϵ in stimulus messages emitted by the node. If the new state and new phase reach 1, node i broadcasts a stimulus message and goes to sleep.

A border node stops embedding b, ϵ, A_b and A_ϵ in a stimulus message, if it has not received any messages from other networks for a certain period of time to notify other nodes of the end of cooperation. By receiving stimulus messages without b, ϵ, A_b and A_ϵ, other nodes also stop embedding these information in stimulus messages.

We should note here that our proposal requires simple floating point arithmetic and as such the computational complexity is not high. Furthermore, each node broadcasts a stimulus message, which can be embedded in a normal message for, e.g. data gathering, only once an operational interval. Thus, the communication overhead is low. Therefore, our proposal can be implemented on a low cost and low performance sensor node and easy to deploy.

3.4 Performance Evaluation

3.4.1 Simulation Setting

We arranged 106 nodes in a 24×24 area as shown in Fig. 3.8. In Fig. 3.8, nodes in the lower-left area belong to Network 1 and the others do Network 2. Therefore, each of networks has four border nodes located in the overlapping area. A circle shown in Fig. 3.8 illustrates the communication range of the node centered at the circle. Parameters are set as summarized in Table 3.1. In addition to duty cycling based on the PCO-model, we further adopt duty cycling on the MAC layer, that is

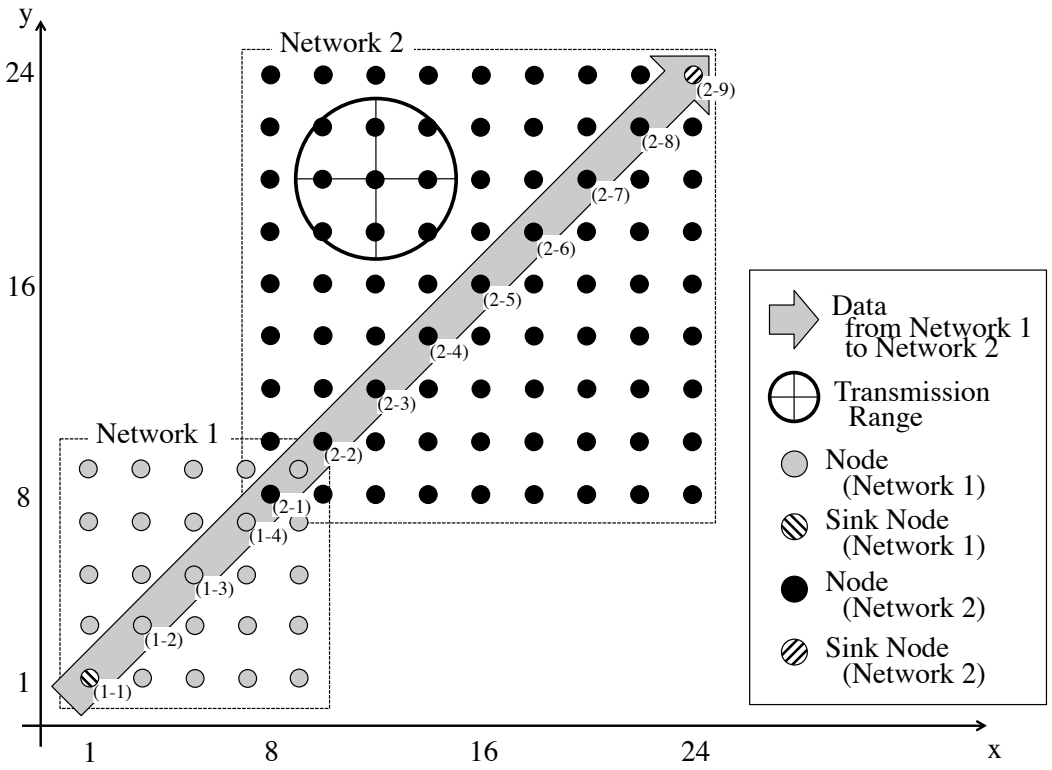


Figure 3.8: Node layout in simulation

X-MAC [28]. In Table 3.1, S_{pre}, S_{ack}, S_{stim} and S_{data} stand for durations that a node sends a Short Preamble, an ACK, a stimulus message and a data packet, respectively. We assumed that 2 wireless sensor networks operating at intervals of 10 seconds and 60 seconds co-exist and share sensor data between them. Nodes belonging to Network 1 set their intrinsic operational frequencies within the range of $[0.100, 0.101]$ at random, and nodes belonging to Network 2 set their intrinsic operational frequencies within the range of $[0.01700, 0.01717]$ at random. The operational interval between successive broadcasting is about 10 seconds in Network 1 and about 60 seconds in Network 2. Initial phases are set at random. The duty ratio is set at 0.3 at all nodes. The simulation time is 50,000 seconds in each simulation run.

In our simulation, the sink node of Network 1 sends a data message to the sink node of Network 2 by using multihop unicast communication once per 10 operational cycles. Data messages take

Table 3.2: Energy consumption model

parameter	value
Initial energy	2000 mAh
Processor active current	8 mA
Sleeping current	15 uA
Sending current	9.9 mA
Waiting and receiving current	19.7 mA

the shortest path to the receiver node following the diagonal of the networks as shown in the figure. When a node between the sink nodes is active and receives a data message, it immediately tries sending the message to a next-hop node. It transmits preambles until it receives an ACK from the next-hop node, even after the end of the PCO-based active period, i.e. expiration of timer. When the transmission of the data message is completed after the expiration of timer, the node moves to the sleep state.

For the purpose of evaluation of energy consumption, we assume that each node is equipped with an Atmel ATmega 128L processor, a Texas Instruments CC2420 radio chip and two AA batteries. The details of energy consumption model is summarized in Table 3.2.

3.4.2 Operational Frequency

We first confirm that the stepwise synchronization is accomplished. Figure 3.9 shows how nodes in Network 2 (slower network) adapt their operational frequencies. Each square corresponds to a node and the color shows the average operational interval adapted by a node during the last 10,000 seconds of simulation time. In this figure, we see that the operational interval of nodes at the border, i.e. four nodes located lower-left of the network, becomes about 20 seconds, closer to the operational interval of Network 1. On the other hand, the operational interval of other nodes become longer than that of those border nodes as the distances to the border become larger. As a result, the stepwise change in operational frequency emerges in accordance with the distance to the border.

Figure 3.10 shows the temporal change of operational intervals adapted by Nodes (2-1), (2-2), (2-3), (2-4), (2-5), and (2-6) during first 5,000 seconds. In this figure, the operational interval

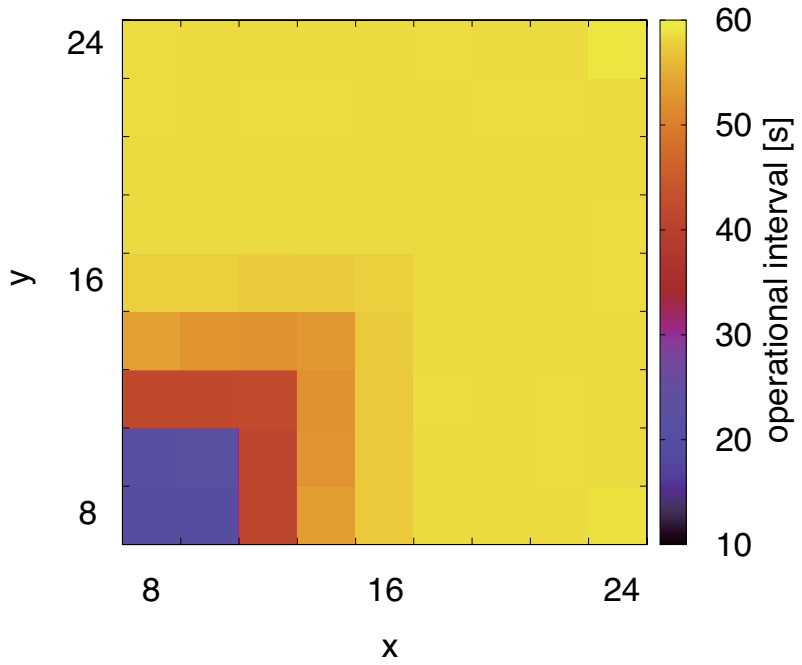


Figure 3.9: Operational interval in stepwise synchronization (Network2)

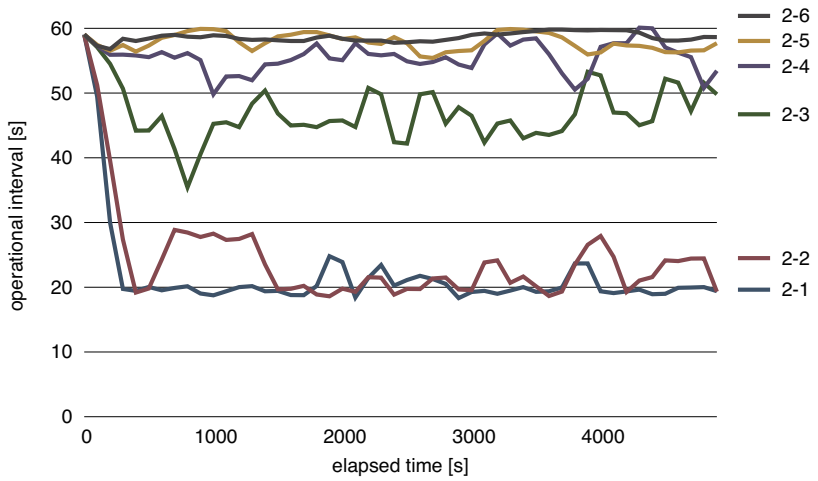


Figure 3.10: Temporal change of operational interval in stepwise synchronization

averaged in a 100 seconds time window is shown. As can be seen in Fig. 3.10, operational intervals of Node (2-1) and (2-2), i.e. border nodes, are shorten to about 20 seconds immediately after the beginning of the stepwise synchronization. By being stimulated by Node (2-2), the operational interval of Node (2-3) decreases to about 45 seconds on average and it contributes to filling the difference in operational intervals between border nodes and other nodes. The operational interval of other nodes slightly change but they are still closer to the original intervals as the distance to the border increases. A reason why operational intervals fluctuate is that stimulus messages are occasionally lost. A loss of a stimulus message prevents a node from being stimulated and adjusting an operational interval. Nevertheless, we can see the stepwise condition of operational intervals is successfully accomplished.

3.4.3 Communication Delay and Energy Consumption

We next compare two scenarios, where both networks keep their intrinsic frequencies denoted as "independent", and our proposal is adopted denoted as "proposal". As performance measures, we use communication delay which is defined as the duration between the time when a node begins to send preambles for transmission of a data message and the time when a node receives an ACK for message reception.

Figure 3.11 shows the median of the communication delay of all data messages transmitted in simulation runs. In Fig. 3.11, "1-1, 1-2" corresponds to the communication delay from Node (1-1) to Node (1-2), for example. Those nodes from Node (1-1) to Node (1-4) belong to Network 1 (faster network), and those from Node (2-1) to Node (2-9) belong to Network 2 (slower network). Node (1-4), Node (2-1) and Node (2-2) are nodes located at the border of networks. In Fig. 3.11, in the case of "independent", although all communication delays between nodes belonging to the same network are quite small, communication delay between nodes located at the border is large at about 12.2 seconds. It is because Node (1-4) has to wait for Node (2-1) located at the border of Network 2 to wake up in transmitting a data message. On the other hand, in the case that our proposal is adopted, communication delay at the border node becomes small, while communication delays between nodes belonging to Network 2 become large. It is because that they do not wake up at

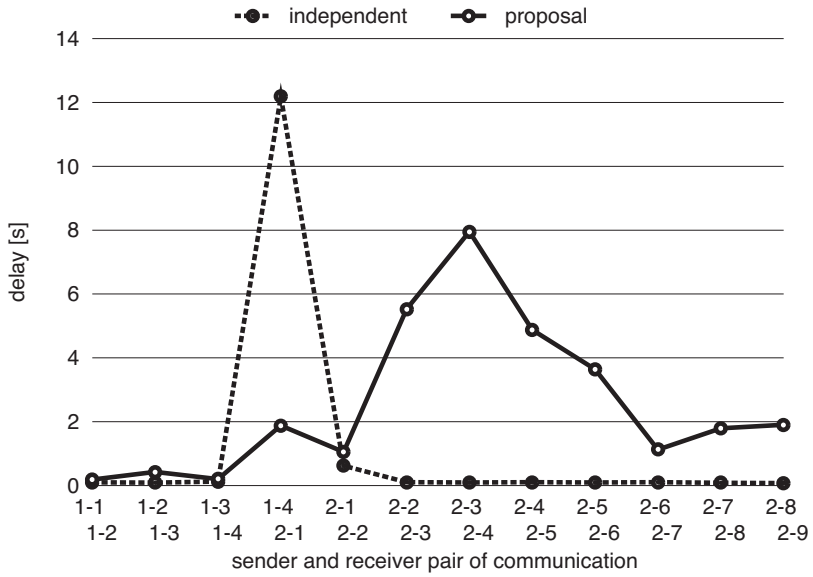


Figure 3.11: Communication delay of each hop

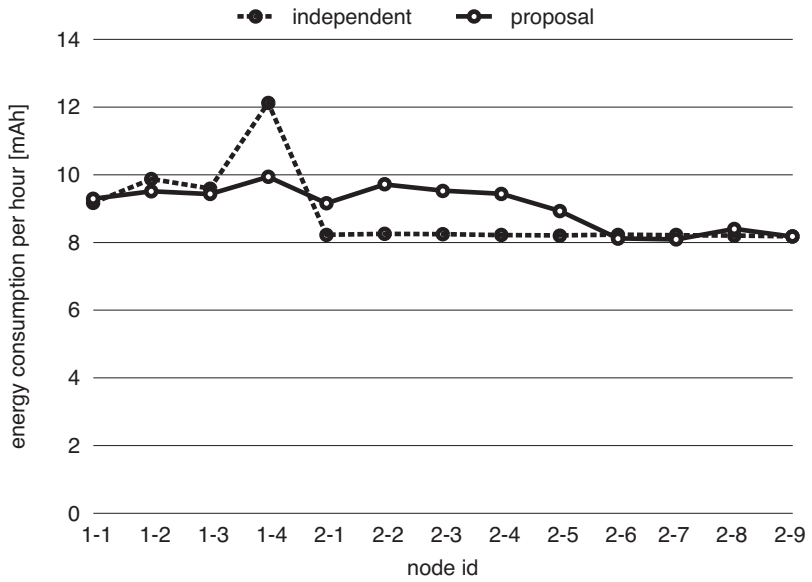


Figure 3.12: Energy consumption per hour

the same time any more for different operational frequencies as shown in Fig. 3.9. The reason that communication delay between Node (2-1) and Node (2-2) is small in the case of "proposal" is that both Node (2-1) and Node (2-2) are located at the border and they operate at similar operational frequency. As stated above, communication delay results from waiting in transmission, during which a node consumes energy. We next evaluate energy consumption, which is a major concern of a wireless sensor network.

Then, simulation results on energy consumption are summarized in Fig. 3.12. As shown in Fig. 3.12, in the case of "independent", Node (1-4), which is located at the border of Network 1, consumes the largest energy, which causes energy dissipation at the border. On the other hand, in the case that our proposal is adopted, the amount of energy consumed at the border node is reduced from 12.1 mAh to 9.9 mAh at the sacrifice of increased energy consumption at nodes in Network 2. In the case that interconnection is mediated by a gateway node [48], it consumes as much energy as a border node of "independent" in Fig. 3.12. Although the total amount of consumed energy is larger with our proposal than the case of "independent", we consider that our proposal benefits wireless sensor networks in balancing energy consumption among nodes, which leads to prolongation of the lifetime of border nodes.

3.4.4 Robustness against Loss of Control Messages

In order to confirm that our proposal accomplishes the stepwise synchronization under the influence of loss of control messages, we dropped stimulus messages at the constant probability. Figure 3.13 shows the average operational interval during the last 10,000 seconds of simulation time averaged over 10 simulation runs for each of message loss probability of 0.0 (no loss), 0.1, 0.2, 0.3, and 0.4. In Fig. 3.13, Node 0 through 24 belong to Network 1 and the others belong to Network 2. In Fig. 3.13, we can see that the stepwise synchronization is accomplished despite the loss probability of stimulus messages. In the figure, the operational interval of nodes near the border of Network 2 becomes slightly larger as the loss probability increases, but operational intervals moderately change in a stepwise fashion nevertheless.

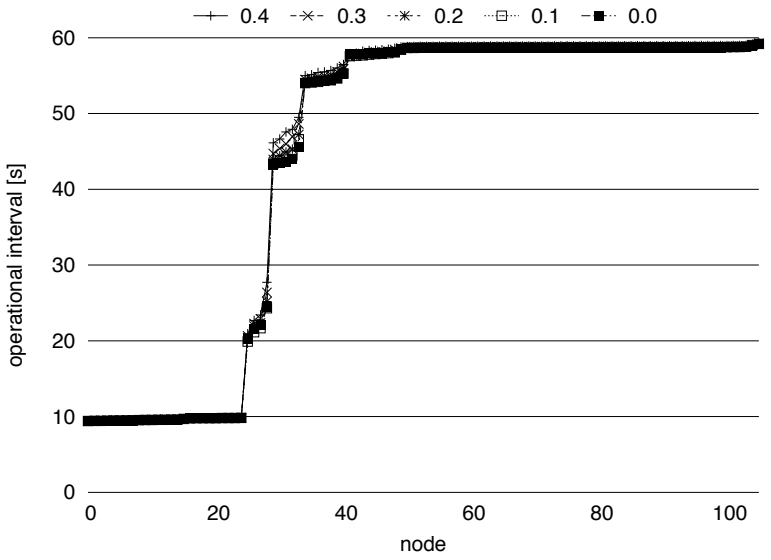


Figure 3.13: Operational interval with different packet loss probability

3.4.5 Adaptivity to Difference of Operational Frequencies of Networks

In order to see how the stepwise synchronization is accomplished among networks with largely different operational frequencies, we changed the operational frequency of Network 2 as shown in Table 3.3 and conducted 10 simulation runs for each of operational frequency settings. A_b and A_ϵ are set at 0.7 and 0.4, respectively. Figure 3.14 shows the average operational interval during the last 10,000 seconds of simulation time averaged over 10 simulation runs. In Fig. 3.14, dashed lines show original operational intervals of nodes in Network 2 for each of settings. In Fig. 3.14, the stepwise synchronization is accomplished independently of difference in operational frequencies. For example, in the case of the 5th simulation setting, where the difference of operational frequencies is largest and the operational frequency of Network 1 is about 360 times faster than that of Network 2, all border nodes of Network 2, i.e. nodes from Node 25 to Node 28, change their operational interval from 3,600 seconds to about 80 seconds. The operational interval of nodes around those border nodes become about 800 seconds. As the distance to the border increases, the operational interval of node becomes longer. In Fig. 3.14, although the operational interval of nodes distant from the border in Network 2 also becomes slightly shorter than the original operational interval, these nodes can keep their operational interval close to the original operational interval to prevent

Table 3.3: Operational frequency setting

	operational frequency [Hz] (operational interval [s])	
	Network 1	Network 2
#1	0.1 (10)	0.017 (60)
#2	0.1 (10)	0.0033 (300)
#3	0.1 (10)	0.0017 (600)
#4	0.1 (10)	0.00055 (1800)
#5	0.1 (10)	0.00027 (3600)

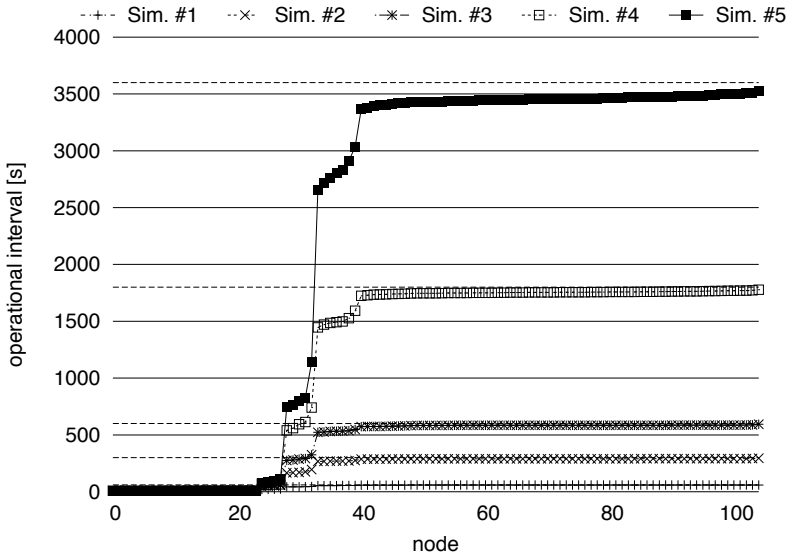
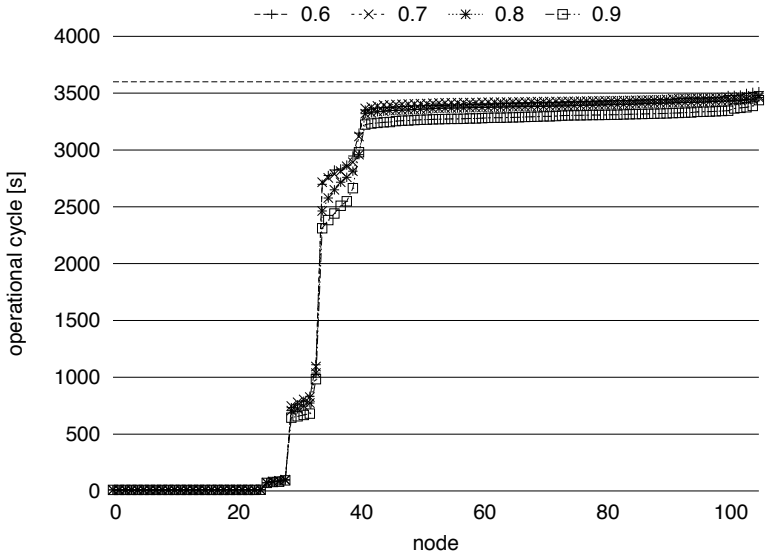


Figure 3.14: Operational interval with different operational frequency

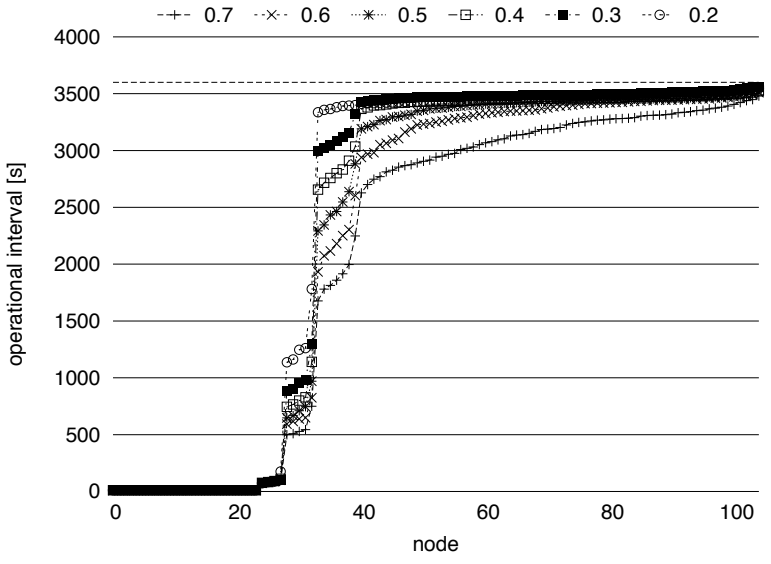
the increase of the energy consumption incurred by the frequent operation.

3.4.6 Influence of Parameter Setting

Attenuation coefficients A_b and A_ϵ influence the degree that distant nodes are stimulated by other network. In Fig. 3.15, we change either of A_b and A_ϵ while the other parameter is fixed. Operational intervals are set as #5 in Table 3.3. As shown in Fig. 3.15(a), A_b does not influence the stepwise synchronization much. A_b affects the amount of the dissipation parameter b, and b becomes larger as A_b increases. In our stepwise synchronization, a node receives a stimulus message in a short duration before it fires due to the duty cycling. Therefore, whenever a node receives a



(a) $A_\epsilon = 0.4$, A_b is changed



(b) $A_b = 0.7$, A_ϵ is changed

Figure 3.15: Operational interval (A_b or A_ϵ are changed)

stimulus message, the phase of the node is large. In the region where the phase is large, the effect of the difference of the dissipation parameter b on the phase-state function is not large as shown in Fig. 3.1. On the contrary, as A_ϵ increases, the degree that nodes change their operational frequency becomes larger as shown in Fig. 3.15(b). It is because that A_ϵ affects the amount of the phase shift on receiving a stimulus message, i.e. the amount of the stimulus ϵ, and the stimulus ϵ becomes larger as A_ϵ increases. When a node receives a stimulus message, the phase of the node reaches 1 form 0 in a shorter duration with larger ϵ and the operational interval of the node becomes shorter as shown in Fig. 3.6. It is obvious that large A_ϵ involve further nodes in stepwise synchronization. By being involved in stepwise synchronization, a node operates at shorter intervals and the frequent operation consumes larger energy. We consider that we should set A_ϵ at 0.3 or 0.4 to accomplish a stepwise synchronization, where only nodes located near the border adjust their operational frequency to bridge the gap in their intrinsic operational frequency while the remaining nodes keep their frequency closer to their intrinsic frequency.

In conclusion, independently of difference in operational frequencies, we can adopt the same pair of $A_b = 0.7$ and $A_\epsilon = 0.4$ to accomplish the stepwise synchronization. We should note here that stepwise adaptation of operational frequencies is limited to the area near border nodes and the number of nodes does not affect the stepwise synchronization or parameter setting.

3.5 Conclusion

In this chapter, to achieve smooth and moderate inter-networking between wireless sensor networks with different operational frequencies, we propose a stepwise synchronization-based inter-networking mechanism. In this mechanism, we adopt the pulse-coupled oscillator model to autonomously accomplish the stepwise synchronization. Through simulation experiments, it was shown that the delay in communication between border nodes and the energy consumption at the border nodes were reduced, but at the sacrifice of energy at nodes near the border in the slower network. Although the total amount of consumed energy is larger with our proposal than the case where both networks keep their intrinsic operational frequencies, we consider that our proposal benefits wireless sensor networks in (i) it balances energy consumption among nodes, which leads

to prolongation of the lifetime of wireless sensor networks, (ii) it can enable wireless sensor networks with largely different operational frequencies to synchronize with each other, and (iii) since the stepwise synchronization emerges as a consequence of mutual interaction between nodes and there is no deterministic rule to determine stepwise operational frequencies, it can adapt to various situations, e.g. increase in the number of networks to cooperate, cooperation among moving networks, and different degree of cooperation.

Although we verified that stepwise synchronization can be accomplished without fine tuning of parameters in the evaluated scenarios, we consider that tuning of parameters would help faster and stable synchronization. Furthermore, other conditions such as the number of networks would affect synchronization. Regarding the speed, synchronization takes time because it is self-organized. As such, our proposal cannot be directly applied to scenarios where nodes have the high mobility. We further plan to evaluate the proposal in such scenarios where there are more than three networks to cooperate, and the degree of overlapping, i.e. the number of border nodes, dynamically changes.

Chapter 4

Study on Interaction between Layered Self-organization based Control

Self-organization is considered one of key design principles to establish highly scalable, adaptive, and robust network systems to accommodate dynamic, diverse, and massive nodes and traffic. Although there are many proposals on self-organization based protocols that are useful, effective, and practical, there has never been any in-depth investigation into interaction, interference, and synergetic effects among multiple self-organization based control. In this chapter, we investigate mutual interaction among layered self-organization based control. We consider an overlay network that is constructed over an ad-hoc network, both of which adopt adaptive routing protocols based on the attractor selection model, i.e. a mathematical model of adaptive behavior of biological systems. We modified the degree of coupling by changing the way how layered self-organization control shared an objective parameter. Through simulation experiments, we showed that the upper layer-aware routing can provide the best performance at the sacrifice of the stability such as variation of paths.

4.1 Introduction

Today, billions of devices, such as PCs, mobile phones, and sensor nodes are connected to the Internet, to make our life safe, secure, and comfortable. The scale of the Internet continuously

increases in the number, heterogeneity, and mobility of devices and many new applications are emerging with help of proliferation of networked devices. It implies that the current Internet is beyond control of conventional mechanisms and protocols, which were designed 30 years ago or added in an ad-hoc manner in response to emerging needs.

First, the growing scale prevents a centralized control mechanism from managing a network system as the whole, because collection of up-to-date information from all nodes consumes considerable amount of bandwidth, energy, and time. In addition, maintenance of global information must be performed frequently, in the same order as speed of change in topology and traffic demand. Even if centralized control is distributed to nodes, they still need to frequently communicate with each other to exchange their state information and to maintain the consistency among them. Secondly, most of current mechanisms and protocols adopt complicated rules with fine-tuned parameters to achieve the optimal performance and to react to dynamic changes such as failures and movement [5, 6]. It means that assumptions on operational environment, such as the number of nodes, diversity in traffic, mobility of nodes, and frequency and size of failures, are made in defining rules and setting parameters. As a result, a network system becomes prone to unexpected events and conditions, which are more likely to occur in a future network, and it easily collapses.

To address the problems, self-organizing control mechanisms, where each node autonomously decides its behavior based on local information it observes or obtains from neighbors and the global control emerges through mutual interaction among neighboring nodes, has been attracting researchers. Self-organization based networking is an interdisciplinary research field, since self-organizing phenomena are recognized in several research fields such as mathematics, physics, chemistry, social science, and biology. Among them, those researches inspired by self-organizing behavior of biological systems are most active and promising [8–10]. Biological systems are inherently autonomous and self-organizing, where there is no centralized control unit dominating the whole, and they often exhibit scalable, adaptive, and robust properties. By being inspired by such biological systems, it is expected that network control mechanisms can achieve the scalability, adaptability, and robustness.

Although there are many successful bio-inspired self-organizing mechanisms, whose superiority to conventional mechanisms were verified through simulation and practical experiments, combination of multiple self-organizing mechanisms is not well-investigated. In [58], they analyzed combination of overlay routing and sleep scheduling in wireless sensor networks and showed that the speed of upper layer control should be as fast and faster than that of lower layer control. Analysis of layered model from an interdisciplinary viewpoint can be found in [59], but our focus in this chapter is on the influence of interdependency among layered self-organizing routing on performance and stability.

In this chapter, we show results of our investigation of interaction between two layered self-organizing control mechanisms. More specifically, as a bio-inspired self-organizing control mechanism, we consider an attractor selection model-based routing [31] and as layering we consider an overlay network constructed over an ad-hoc network. The attractor selection model is a mathematical model which is derived from non-rule based adaptation of biological systems [60]. The attractor selection model has been applied to multi-path routing in overlay networks [61], clustering in wireless sensor networks [62], clustering in mobile ad-hoc networks [63], coverage control in wireless sensor networks [64] and so on. When layered routing protocols are tightly coupled with each other, a slight change in a path in one layer would easily and strongly affect the other. It leads to global optimization where both control reach the stable and optimal solutions in some cases, but it also results in instable routing where paths continuously fluctuate in both layers. The target scenario of the chapter seems rather specific, but we believe that it becomes the first step toward understanding interaction between layered self-organization, which can be found in nature and artificial systems.

The rest of this chapter is organized as follows. First, in Section 4.2, we briefly introduce the attractor selection model, and then we explain the attractor selection-based routing protocol in Section 4.3. Next, we describe how overlay routing and ad-hoc routing are coupled with each other in Section 4.4. Then, in Section 4.5, we discuss results of our simulation experiments. Finally, we conclude the chapter in Section 4.6.

4.2 Attractor Selection Model

The attractor selection model is a mathematical model, which explains adaptive behavior of biological systems [60]. An attractor is a stable state of a dynamical system where the system converges after transient states from the initial state. In [60], bacteria, i.e. E. coli cells, synthesize two different nutrients. Since synthesis of nutrients mutually inhibit each other, a bacterial cell generates either of them at once. When the environment has sufficient amount of both nutrients, cells can live independently of nutrients it synthesize. Once the amount of either of nutrients in the environment decreases, a cell begins to synthesize the missing nutrient to compensate the lack autonomously and adaptively. A bacterial cell does not have predetermined rules for such adaptive behavior. Instead, it is explained by the attractor selection model, where noise drives adaptation.

In the attractor selection model, synthesis of nutrients is governed by gene expression of mRNA. The dynamics of gene expression of bacteria i is determined by the following equation.

$$\frac{d\vec{m}_i}{dt} = f(\vec{m}_i) \times \alpha_i + \vec{\eta}_i \tag{4.1}$$

where $\vec{m}_i$ is a vector of concentrations of mRNA corresponding to synthesis of each nutrient. When we consider two different nutrients, $\vec{m}_i = (m_{i,1}, m_{i,2})$. $f(\vec{m}_i)$ is a function which determines attractors. α_i $(0 \leq \alpha_i \leq 1)$ is a parameter called *activity* corresponding to the degree of the goodness of the current state, e.g. the growth rate of bacteria i. $\vec{\eta}_i$ corresponds to the white Gaussian noise, which is inherent to environment. When we consider two different nutrients, a dynamical system based on the attractor selection model has two stable states, i.e. attractors, where $m_{i,1} > m_{i,2}$ or $m_{i,1} < m_{i,2}$.

When gene expression, i.e. synthesis of nutrients, is suitable for the current environment, bacteria i can obtain the sufficient amount of both of nutrients. Consequently, activity α_i increases. Since the value of $f(\vec{m}_i) \times \alpha_i$ in Eq. 4.1 becomes larger than the noise term, gene expression is drawn into an attractor by being dominated by function $f(\vec{m}_i)$, and it stays at the attractor stably. Once the nutrient condition changes and the current gene expression becomes unsuitable for the new environmental condition, activity α_i decreases. With small α_i, the value of $f(\vec{m}_i) \times \alpha_i$ becomes small

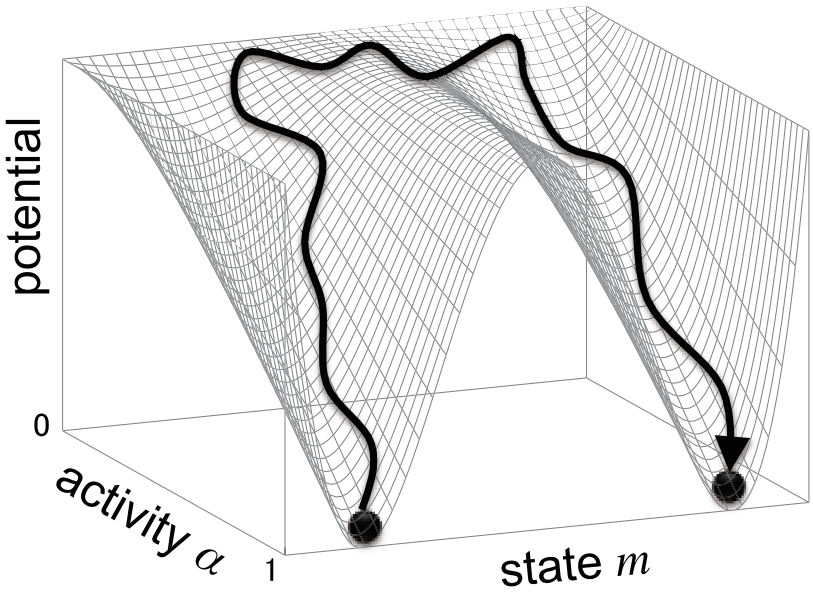


Figure 4.1: Adaptive selection of attractors in attractor selection model.

and the impact of $f(\vec{m}_i)$ on gene expression becomes relatively small, and the dynamics of gene expression is therefore governed by the noise term. If the new gene expression adapted by a noise is proper to the condition of environment, activity α_i gradually increases and the gene expression converges at an attractor.

In Fig. 4.1, we show a sketch of adaptive selection of attractors in the attractor selection model with two different nutrients. There are two attractors, $m_{i,1} > m_{i,2}$ or $m_{i,1} < m_{i,2}$, illustrated as valleys or basins. In the figure, the x-axis shows the state m, the y-axis shows the value of activity, and the z-axis shows the potential field determined by function $f(\vec{m}_i) \times \alpha_i$. Therefore, the goodness of current state determines the depth of basins. When it really is good, the activity is high and basins become deep. In other words, the degree of entrainment of attractors becomes strong. The state easily converges to one of attractors and stably remains there while having slight fluctuation caused by the noise term. On the contrary, when the environmental condition changes and the current state is not suitable any more, the activity decreases and basins become shallow. Consequently, the state is driven by the noise term more than the potential function, it randomly moves in the solution space. When the state occasionally approaches an attractor, which is suitable

for the current environmental environment, activity begins increasing and the state moves to the new attractor. This is the basic mechanism of adaptation of the attractor selection model.

4.3 Bio-inspired Adaptive Routing based on Attractor Selection Model

In this chapter, we consider interaction between routing protocols running on a wireless ad-hoc network and an overlay network, which is constructed over the ad-hoc network. As a routing protocol, nodes adopt the simplified version of MARAS [31], which is a routing protocol designed for mobile ad-hoc networks based on the attractor selection model.

4.3.1 Overview

MARAS is an on-demand routing protocol, where a path is established when it is required by a source node to send data messages to a designated destination node. Once a data message leaves a source node, it is forwarded among intermediate nodes toward the destination node. Each intermediate node selects a neighboring node to forward a data message every time a data message arrives. For the purpose of selection of forwarding node, node i maintains a list of routing information. Routing information for destination node d is composed of variables of the attractor selection model, $\vec{m}_d = \{m_{d,j} | j \in N_i\}$ and α_d. $\vec{m}_d$ is called state vector, which is a vector of state values $m_{d,j}$ corresponding to the goodness of selection of neighboring node j for destination d. α_d is the activity, which expresses the goodness of the current path to destination node d. Details of usage of them will be explained in the following sections. N_i corresponds to a set of neighboring nodes of node i.

4.3.2 Route Establishment

At the beginning of communication, source node s looks up routing information corresponding to the intended destination node d in the list of routing information. If node s dose not have the routing information for destination node d, it generates a route request (RREQ) message and broadcasts it as in the case of AODV [65]. When a node receives a RREQ message destined for a node other than itself, it records its own ID on the RREQ message and broadcasts it. In this way, the RREQ

message memorizes nodes that it has visited while being forwarded by broadcasting. When the number of forwarding reaches the maximum limitation determined in advance, the RREQ message is discarded. Since each RREQ message has a unique ID, duplicated RREQ messages are detected and discarded at an intermediate node.

When destination node d receives a RREQ message, the node generates a route reply (RREP) message and sends the message toward source node s by multi-hop unicast communication. A RREP message travels along the reverse path of the corresponding RREQ message. When a node including a source node receives a RREP message, it initializes routing information corresponding to destination node d. Regarding the state vector $\vec{m}_d$, the state value $m_{d,j}$ corresponding to the neighboring node j, from which it received the RREP message, is set at 10. In contrast, state values corresponding to the other nodes are set at 0. The activity α_d is set at 1.0, which means that the initial path is the current best. Finally, when the source node s receives a RREP message, it begins to send data messages.

4.3.3 Data Message Forwarding

When a node receives a data message destined for node d from node k, it selects node j, which has the maximum state value in $\vec{m}_d$ except node k, and forwards the data message to node j. If node k is only neighbor, the data message is discarded. A data message is forwarded by intermediate nodes while memorizing them until it reaches the destination node d or it is discarded by reaching the maximum number of forwarding.

4.3.4 Feedback and Route Maintenance

When destination node d receives a data message, it generates a feedback message and sends the message toward source node s. A feedback message is forwarded along the reverse path where the corresponding data message travelled, and eventually reaches node s.

When a node including a source node receives a feedback message at time t, it updates its

routing information for node d. First, activity α_d is derived by the following equation.

$$\alpha_d = \frac{\min_{t-T<t'\leq t} w(t')}{w(t)} \tag{4.2}$$

where $w(t)$ corresponds to the travelled hop count of the feedback message from the destination node to the node itself received at time t, and T $(T > 0)$ is a parameter.

By using the new activity α_d, the node updates state vector $\vec{m}_d$ for all neighboring nodes $j \in N_i$ by the following equation.

$$\frac{dm_{d,j}}{dt} = \frac{\alpha_d(\beta\alpha_d^{\gamma} + 1/\sqrt{2})}{1 + m_{d,max}^2 + m_{d,j}^2} - \alpha_d m_{d,j} + (1 - \alpha_d)\eta_d \tag{4.3}$$

where $m_{d,max} = \max_{k\in N_i} m_{d,k}$, and η_d is the Gaussian noise. β $(\beta > 0)$ and γ $(\gamma > 0)$ are parameters which control the influence of the activity on state values.

When the activity is high, a nonlinear dynamic system governed by Eq. 4.3 reaches an attractor, i.e. stable state, where one $m_{d,j}$ $(j \in N_i)$ has a high value and the other $m_{d,k}$ $(k \in N_i - \{j\})$ have a low value. It means that the current path is short enough and a node preferentially selects a certain node j forming the path as a next hop node. On the other hand, when the current path becomes inappropriate for node failure or link disconnection, the activity decreases. As a result, selection of next hop node will be driven by the third term of the right side of Eq. 4.3, i.e. noise. When a node occasionally selects a new good neighbor node, the activity eventually increases and it recovers preferential selection leading to the shortest path in the new condition. In summary, the attractor selection model is a kind of heuristics to find a state vector maximizing the activity, which is interpreted as the shortness of path in our routing.

In addition to feedback-based updating, the activity decays at intervals of τ $(\tau > 0)$ by the following equation and the state vector is updated by using the decayed activity accordingly, regardless of whether the node receives feedback messages in the preceding interval.

$$\alpha_d = \alpha_d - \delta \tag{4.4}$$

where δ $(\delta > 0)$ is a constant. When activity α_d becomes 0, the routing information corresponding

to destination node d is removed from a list.

4.3.5 Routing in Overlay Networks

Although MARAS is designed for mobile ad-hoc networks, it is applicable to routing in an overlay network as well. Whereas it is possible to adopt a variety of performance measures as activity, we use delay in the following discussions.

In the case of routing in an overlay network, $w(t)$ in Eq. (4.2) is newly defined as the duration that a data message sent from an overlay node takes to reach an destination overlay node, i.e. the one-way delay to a destination node in an overlay network. For this purpose, in sending and forwarding a data message, a source overlay node and intermediate overlay nodes embed a time stamp in the data message. On receiving a data message, a destination overlay node calculates communication delay from those nodes to itself and inform to them by a feedback message. Since communication delay varies in a very short period of time, the activity defined by communication delay would fluctuate very much. Since it causes unacceptably frequent changes of an overlay path, we use the smoothed activity derived by the exponential moving average of instantaneous activity values as follows.

$$\overline{\alpha}_{ON,d}(t) = A\alpha_{ON,d}(t) + (1 - A)\overline{\alpha}_{ON,d}(t - 1) \tag{4.5}$$

where $\overline{\alpha}_{ON,d}(t)$ corresponds to the activity for destination overlay node d in an overlay network at time t and $\alpha_{ON,d}(t)$ is an instantaneous value of the activity. The coefficient A $(0 < A < 1)$ is a smoothing factor which we empirically set A at $1/8$ in this chapter.

4.4 Coupling Layered Self-organization Based Routing

In this chapter, we consider a layered network model illustrated in Fig. 4.2. A lower layer is a wireless ad-hoc network, which consists of randomly distributed wireless ad-hoc nodes, and an overlay network, which is constructed on the ad-hoc network by appointing randomly chosen ad-hoc nodes as overlay nodes. The routing protocols explained in the previous section are adopted

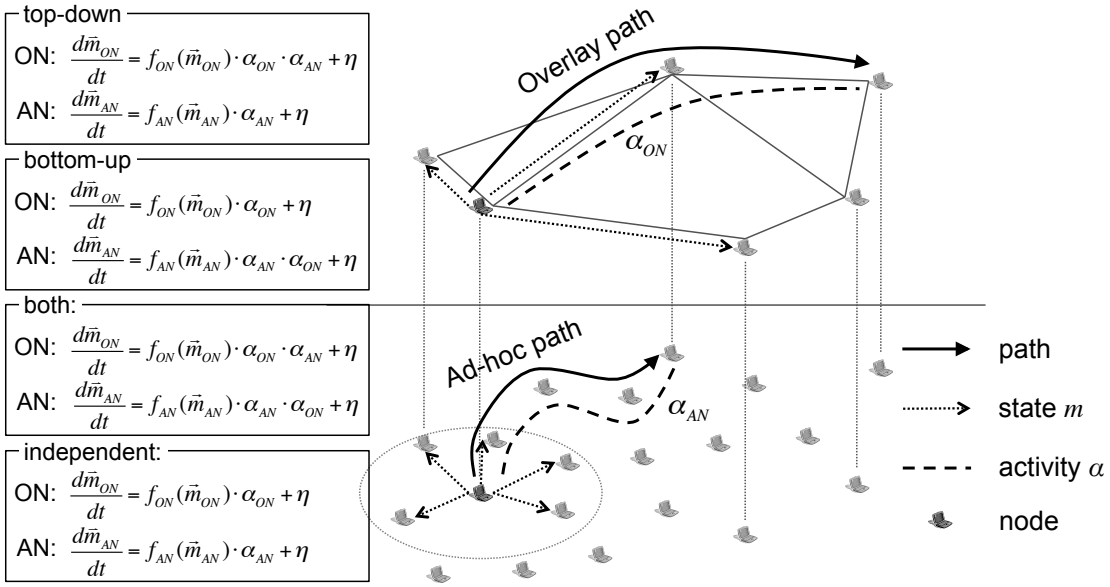


Figure 4.2: Layered routing based on attractor selection model.

on each of layers. Therefore, a node which belongs to both of an overlay network and an ad-hoc network maintains two lists of routing information, i.e. a list of state vector $\vec{m}_{ON}$ and smoothed activity $\overline{\alpha}_{ON}$ for destination overlay nodes as an overlay node and a list of state vector $\vec{m}_{AN}$ and activity α_{AN} for destination ad-hoc nodes as an ad-hoc node. As stated in previous section, in the overlay network, $w(t)$ is defined as the communication delay from the node itself to the destination overlay node. On the other hand, it is defined as the number of hops from a destination node of an ad-hoc path to the node itself.

Because of the layered structure, overlay routing and ad-hoc routing influence each other. When an overlay node intends to send data messages to another overlay node, it initiates routing procedures to find the shortest end-to-end overlay path in an overlay network. Since an overlay link is physically composed of ad-hoc nodes, it triggers ad-hoc routing to establish an ad-hoc path from an ad-hoc node corresponding to one end of an overlay link to an ad-hoc node corresponding to the other end of the overlay link. In an ad-hoc network, each node tries to establish and maintain the shortest path to a destination node, but a destination node dynamically changes due to the adaptive selection of a next hop overlay node in overlay routing. When an overlay node changes a next hop

node in an overlay network, it, as an ad-hoc node, also needs to establish or update an ad-hoc path to an ad-hoc node corresponding to the new next hop overlay node. If an ad-hoc path corresponding to a new overlay path is not stably established, the length of an ad-hoc path and the communication delay of corresponding overlay link dynamically changes due to adaptive data message forwarding by the attractor selection model. As a result, the communication delay of the end-to-end path from an source overlay node to an destination overlay node fluctuates and activity $\overline{\alpha}_{ON}$ changes accordingly. If activity $\overline{\alpha}_{ON}$ occasionally decreases, an overlay path changes and it further triggers reconstruction of ad-hoc paths. Such mutual interaction sometimes results in the shortest and stable path, but it possibly causes an unstable and fluctuating path.

To evaluate how interdependency among layered control affects performance and stability, we consider four alternatives of coupling which differ in the way of activity sharing.

- Independent: Each network tries to achieve their own goals, i.e. high activity and short or low delay path, while being influenced by behavior of the other layer implicitly. An overlay node updates its state vectors $\vec{m}_{ON}$ by using its own activity $\overline{\alpha}_{ON}$ and an ad-hoc node updates its state vectors $\vec{m}_{AN}$ by using its own activity α_{AN}.

- Ad-hoc aware overlay routing (top-down): An overlay network chooses an overlay path which improves the performance of ad-hoc routing, for example, by detouring a area where node failures occurred. An overlay node updates its state vectors $\vec{m}_{ON}$ by using the combined activity $\overline{\alpha}_{ON} \times \overline{\alpha}_{AN}$ and an ad-hoc node updates its state vectors $\vec{m}_{AN}$ by using its own activity α_{AN}. Since an overlay network is controlled at the slower speed than an ad-hoc network, an overlay network adopts the smoothed activity of an ad-hoc network $\overline{\alpha}_{AN}$ for update of state vector $\vec{m}_{ON}$ to moderate variation of the activity of an ad-hoc network. As in the case of the smoothed activity of an overlay network, the smoothed activity of an ad-hoc network $\overline{\alpha}_{AN}$ is derived by calculating the exponential moving average of instantaneous values of activity α_{AN}.

- Overlay-aware ad-hoc routing (bottom-up): An ad-hoc network chooses a physical path connecting two overlay nodes, which leads to better overlay end-to-end performance, i.e. low

end-to-end communication delay. An ad-hoc node, which also belongs to an overlay network, updates its state vectors $\vec{m}_{AN}$ by using the combined activity $\alpha_{AN} \times \overline{\alpha}_{ON}$, and an overlay node updates its state vectors $\vec{m}_{ON}$ by using its own activity $\overline{\alpha}_{ON}$.

- Tight coupling (both): Both of an overlay network and an ad-hoc network try to maximize the total performance by sharing the same goal. Both of an overlay node and an ad-hoc node, which also belong to an overlay network, update their state vectors $\vec{m}_{ON}$ and $\vec{m}_{AN}$ by using the combined activity $\overline{\alpha}_{ON} \times \overline{\alpha}_{AN}$ and $\alpha_{AN} \times \overline{\alpha}_{ON}$, respectively.

Intuitively speaking, the tight coupling, called "both", leads to the best performance among the above four alternatives. However, the global optimization with multiple criteria is harder to achieve and it would take longer time than the other couplings. Furthermore, the tight control is vulnerable to failures and unexpected events. Going back to biological systems, they are not fully optimized, incomplete, and redundant. However, such non-optimality, incompleteness, and redundancy are the source of adaptability and robustness.

4.5 Simulation Experiment

To evaluate how interdependency among layered control affects performance and stability, we conducted simulation experiments with the network simulator NS-3 [66].

4.5.1 Setting

We arranged 198 immobile ad-hoc nodes in a 400×400 m area randomly and 2 immobile ad-hoc nodes in corners of the area, i.e. (0, 0) and (400, 400), and then constructed an ad-hoc network of those 200 nodes. In the ad-hoc network, each node adopted IEEE 802.11 as MAC protocol and could communicate with any nodes within the communication range of 60 m. The nodes locating at (0, 0) and (400, 400) and randomly chosen 18 ad-hoc nodes were appointed as overlay nodes. They belonged to both of an ad-hoc network and an overlay network. An overlay network was constructed by establishing 50 overlay links between randomly chosen pairs of 20 overlay nodes.

Table 4.1: Parameter setting (left: ON, right: AN).

parameter	value
β	10
γ	5
δ	0.05
T	30
τ	30

parameter	value
β	10
γ	5
δ	0.1
T	10
τ	3

Table 4.2: Message size (left: ON, right: AN).

message type	size [Byte]
RREQ	16
RREP	12
Feedback	16
Data header	16
Data payload	512

message type	size [Byte]
RREQ	24
RREP	20
Feedback	20
Hello	24

In all simulation experiments, we appointed the node locating at (0, 0) as a source overlay node and the node locating at (400, 400) as a destination overlay node, respectively.

In order to manage a list of neighboring nodes, each ad-hoc node broadcasted HELLO messages at regular intervals of 1 s. A node registered a node, from which it received a HELLO message, in its list of neighboring nodes, and removed a node from the list if it did not receive two successive HELLO messages from the node. In an overlay network, in order to maintain physical paths corresponding to each overlay link, each overlay node sent HELLO messages at regular intervals of 1 s to all overlay nodes to which it had overlay links. Note that an overlay node did not remove an overlay link between itself and another overlay node even if it did not receive HELLO messages from the overlay node for a long time, since HELLO messages in an overlay network are adopted for the sake of maintenance of physical paths of overlay links.

A source node began to generate data messages at regular intervals of 0.25 s and send it toward the destination node at the time 150 s, and it stopped generating data messages at the time 700 s. At the time 200 s, we randomly chose 20 ad-hoc nodes out of all nodes except nodes belonging to an overlay network, and removed them to emulate node failures. We set parameters of attractor selection model for each networks as shown in Table 4.1. The size of each message is shown in

Table 4.2. An interval of routing information updating in an overlay network is set at 10 times as long as that in an ad-hoc network by updating the routing information for every 10 feedback messages. At the beginning of a simulation run, all of ad-hoc and overlay nodes did not have any route information. We generated 100 random topologies and evaluated four alternatives of coupling on each of the topologies.

4.5.2 Results and Discussion

First, we show how interdependency among layered control affects performance and stability in path recovery from path disruptions due to node failures at the time 200 s. We consider that a path recovers from a disruption and converges on a new path, if 40 successive data messages, which equivalent to 10 s, are forwarded along the same physical path from the source overlay node to the destination overlay node. We define a path recovery phase as a period between the time when node failures occur, i.e. the time 200 s, and the time when the path converges on a new path. Figure 4.3 shows the number of variation of overlay paths during the path recovery phase. An overlay path is considered to change if a data message is forwarded along an overlay path that is different from the path along which a previous data message is forwarded. In Fig. 4.3, results of 100 topologies are arranged along the x-axis in the ascending order of the number of variation of overlay paths. In addition, Figure 4.4 shows the standard deviation of the hop count of whole physical paths from a source overlay node to a destination overlay node during the path recovery phase, which indicates how large fluctuation of paths is. Furthermore, Figure 4.5 shows the ratio between the number of data messages that a destination overlay node received and the number of data messages that a source overlay node sent during the path recovery phase. As in the case of Fig. 4.3, in these figures, results of 100 topologies are arranged along the x-axis in the ascending order of y-axis.

As can be seen in these figures, the coupling of “independent”, where each network tries to achieve their own goal, results in the smallest number of overlay path variation, the smallest fluctuation of paths, and highest delivery ratio of data messages. The coupling of “bottom-up”, where an ad-hoc node that also belongs to an overlay network takes into account the activity of an overlay network in derivation of a state vector, follows “independent”, and achieves small overlay path

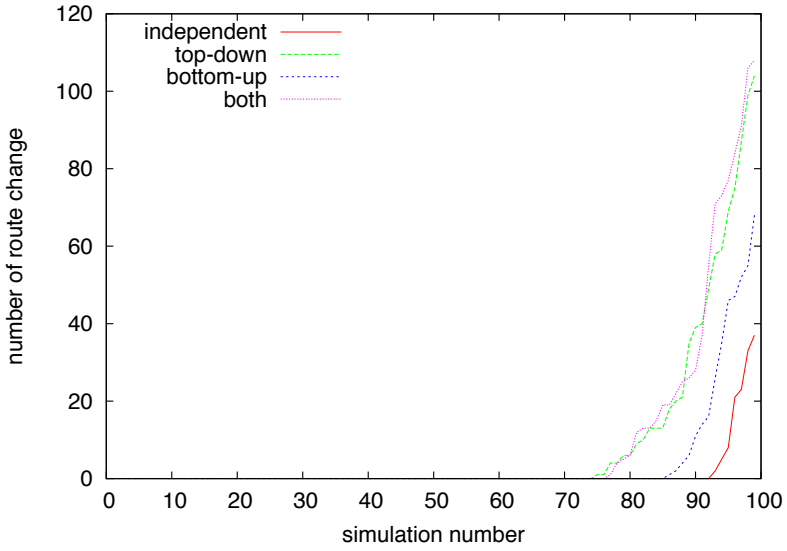


Figure 4.3: Variation of overlay path in path recovery

variation and physical path fluctuation, and high delivery ratio. In contrast, both of the coupling of "top-down", where an overlay node takes into account the activity of an ad-hoc network in derivation of a state vector, and the coupling of "both", where both of overlay and ad-hoc nodes take into account the activity of the other network in derivation of a state vector, result in high overlay path variation and physical path fluctuation, and then "both" results in the worst. Note that "top-down" achieves better delivery ratio of data messages than "bottom-up".

In an overlay network, since the activity is smoothed by the exponential moving average, instantaneous increase of end-to-end communication delay has a small effect on the activity used in state vector calculation, and then overlay paths dose not immediately change in response to the instantaneous perturbation. Therefore, in the case where an overlay network is operating independently of an ad-hoc network, i.e. "independent" and "bottom-up", an ad-hoc network alone looks for new ad-hoc paths to compensate the decrease of activity caused by node failures and converges on a new ad-hoc path before an overlay network begins path exploration. If the end-to-end communication delay largely fluctuates by reason that an ad-hoc network cannot find a better path and converge for a long time, the activity of an overlay network gradually decreases, then an overlay network

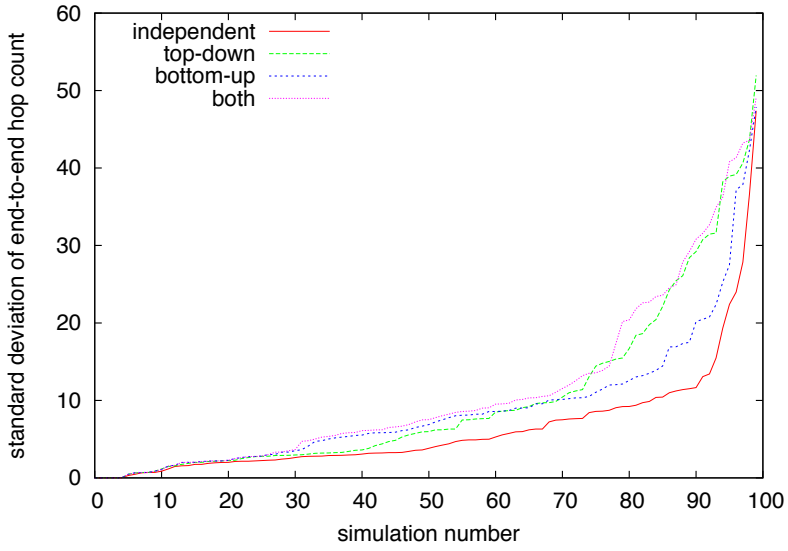


Figure 4.4: Standard deviation of path length in path recovery

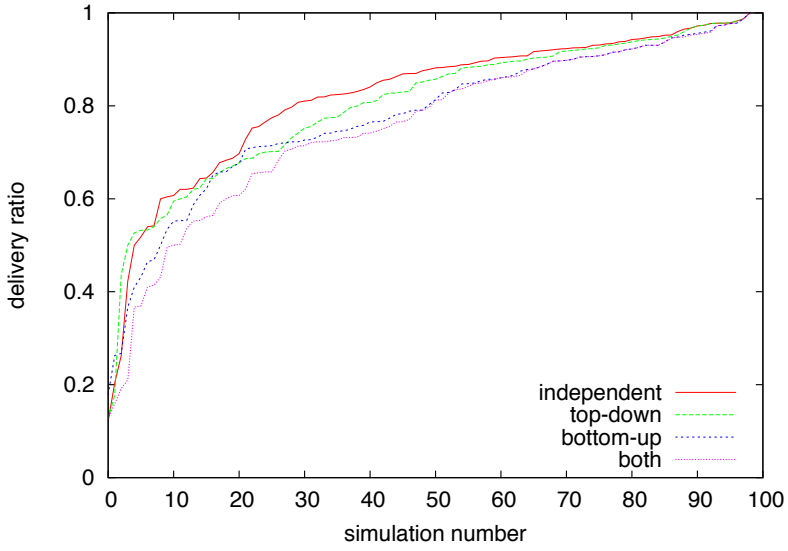


Figure 4.5: Delivery ratio in path recovery

eventually begins to look for new overlay paths to increase the activity, i.e. smaller delay. A change in an overlay path would cause adaptive path establishment in an ad-hoc network to shorten the delay of each overlay link. Consequently, paths converge in both of an overlay network and an ad-hoc network. Thus, "independent" results in the most moderate convergence and achieves the highest delivery ratio. In contrast, in "bottom-up", small overlay activity drives an ad-hoc network to keep searching for better paths. Because of the aggressive search and resultant fluctuation in an ad-hoc network, the communication delay observed by an overlay network also fluctuates which further triggers path exploration in the overlay network. As a result, compared with "independent", "bottom-up" results in higher overlay path variation and physical path fluctuation, and lower delivery ratio.

On the other hand, in the case where an overlay node takes into account the activity of an ad-hoc network as in "top-down" and "both", the decrease of the activity of an ad-hoc network in the path recovery phase makes an overlay network search for detour paths to avoid the region affected by failed nodes in more aggressive manner than "independent" and "bottom-up". The activity defined for an overlay network, i.e. delay, is inherently affected by the goodness of ad-hoc paths, since the upper-layer protocol is served by the lower-layer protocol in the layered protocol architecture. Therefore, multiplying the lower-layer activity and the upper-layer activity in upper-layer control strengthens the influence of lower-layer control very much. The aggressive search leads to high variation of overlay paths, which further causes fluctuation of physical paths in an ad-hoc network. As a result, overlay path considerably fluctuates. Comparing "top-down" and "both", the former suffers less from the strong interdependency. Since an ad-hoc network can move toward convergence without the direct influence of small activity of an overlay network, an overlay network can find overlay paths consisting of good and stable ad-hoc paths. On the contrary, in "both", path exploration in an ad-hoc network and an overlay network affects each other. A path from a source overlay node to a destination overlay node considerably fluctuates and it results in low delivery ratio of data messages.

Next, we show time series of end-to-end communication delay between a source overlay node and a destination overlay node in Fig. 4.6. In Fig. 4.6, the communication delay is averaged over data messages sent at the same timing in 100 simulation runs and the obtained averages are

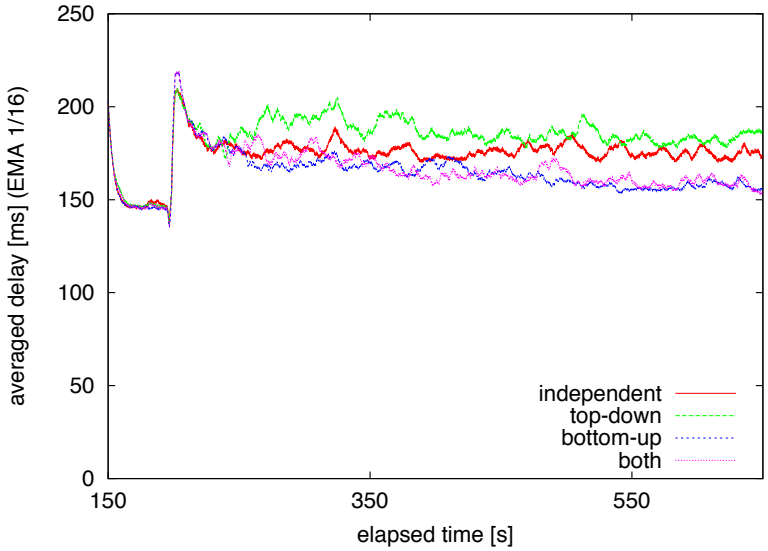


Figure 4.6: Averaged end-to-end communication delay

further smoothed by the exponentially moving average with the smoothing factor of $1/16$. In our simulation experiments, a source overlay node began to send data messages at the time 150 s and randomly chosen nodes were removed at the time 200 s. Although communication delay right after a source node begins data message emission is large because of the congestion caused by RREQ broadcasting for route establishment and ARP messages for resolution of MAC address of neighboring nodes, communication delay before node failures are rather stable around almost the same value independently of the coupling of networks. Once node failures occur at the time 200 s, communication delay increases temporarily independently of the coupling during the path recovery phase, and then gradually decreases as time passes.

In “bottom-up”, the communication delay decreases the fastest and converges on the smallest value among all, because an ad-hoc network finds paths minimizing overlay communication delay, and as such an overlay network is likely to satisfy with the new overlay paths. In “both”, communication delay is as small as in “bottom-up”, while communication delay fluctuates slightly more than “bottom-up” for aggressive path exploration in an overlay network. In contrast, in “independent”,

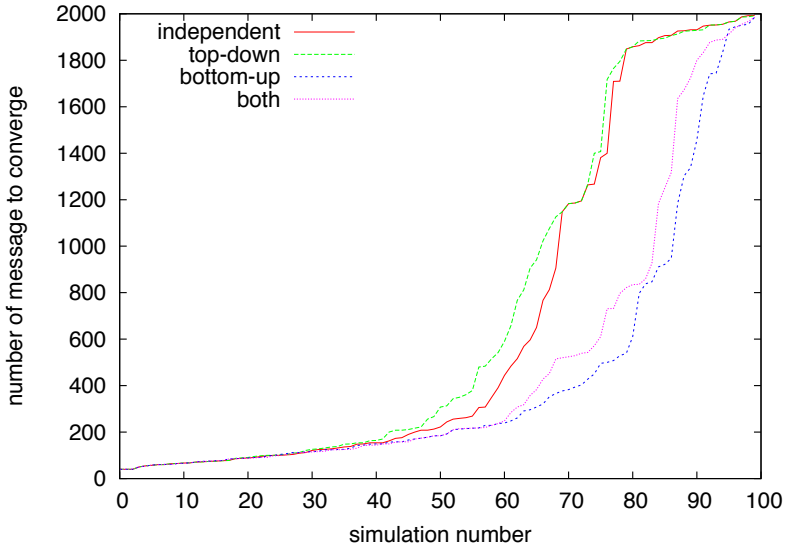


Figure 4.7: Convergence time

communication delay is the most stable by being kept around the same level. However, independent path recovery cannot achieve the small communication delay as in "bottom-up" and "both". In "top-down", in spite of the temporal decrease of communication delay after node failures, the delay increases again. Furthermore, the delay stayed at the highest level even after convergence. It is because an overlay network tries to minimize not only the communication delay of itself but also the physical hop length of an ad-hoc network.

Finally, as an indirect measure of recovery speed, the number of data messages that a source overlay node sends in the path recovery phase is shown in Fig. 4.7. From the figure, it can be seen that the coupling of "bottom-up", where communication delay decreases the fastest in Fig. 4.6, converges on new paths in the shortest time, followed by the coupling of "both", where the delay converges on the small value too. In contrast, the coupling of "top-down" needs the longest time to converge on new paths.

In summary, the coupling of "independent" achieves the most moderate and stable path recovery, while the communication delay on new paths is not the smallest among four couplings. On the other hand, the coupling of "bottom-up" converges on the paths with the shortest communication

delay the fastest among four couplings at the sacrifice of the stability and the delivery ratio during path recovery. However, we can compensate the loss by retransmission triggered by the event that a source node does not receive a corresponding feedback message for certain duration of time.

4.6 Conclusion

In this chapter, to investigate mutual interaction among layered self-organization based control, we consider an overlay network constructed on an ad-hoc network both of which adopt an attractor selection model-based routing mechanism. We evaluated the influence of different degree of coupling by changing the way how layered control share an objective parameter, i.e. the activity. Through simulation experiments, we showed that upper layer-aware routing can provide the best performance in end-to-end communication delay.

For future work, we plan to consider scenarios where a topology dynamically changes due to the mobility of ad-hoc nodes. In addition, we are going to consider combinations of other self-organizing protocols such as clustering and scheduling.

Chapter 5

Conclusion

Self-organization is one of key design principles of future networks to overcome the difficulty in managing a large number of heterogeneous nodes and their traffic under dynamically changing environment. Since a monolithic network with a single self-organizing control mechanism cannot satisfy a variety of requirements, we need to employ multiple self-organizing control mechanisms and coordinate them to operate in cooperative manner.

First, we propose a self-organizing video coding rate control mechanism for wireless camera networks, where nodes cooperate with each other to adjust video coding rate not to exceed the network capacity while obtaining high quality images of targets. In camera sensor networks, despite its usefulness in applications such as remote surveillance and monitoring, there is a major challenging issue, where the capacity of wireless sensor networks is limited to accommodate huge traffic such as video images generated by cameras even if they adopt well-known techniques on QoS control for video transmission in wireless networks. To address the problem, we proposed a video coding ratio control mechanism to suppress the video coding rate to avoid wasting the wireless network capacity in transmission of irrelevant and redundant video data. In our mechanism, by adopting a reaction-diffusion model, each node adjusts the video coding ratio adaptably without global knowledge. Through simulation and practical experiments, we verified that our proposal can avoid network congestion and keep the quality of video image from a camera having the target high.

Next, we propose a mechanism for multiple sensor networks to cooperate and be integrated

for smooth communication among networks to offer a unified service to users, that is stepwise synchronization. In our stepwise synchronization, sensor nodes located near the border of sensor networks adjust their operational frequencies to bridge the gap in their intrinsic operational frequencies. To accomplish the stepwise synchronization, we adopted the pulse-coupled oscillator model and strengthen the degree of entrainment at border nodes to intensively shift the operational frequency toward that of the other network while the degree of entrainment is weakened as the distance to the border increases. As a result, the operational frequencies of nodes near the border are adjusted to somewhere between the original operational frequencies of wireless sensor networks. Through simulation experiments, it was shown that the delay in communication between border nodes was reduced, and the energy consumption was balanced among nodes, which leads to prolongation of the lifetime of wireless sensor networks.

Finally, we further investigate the influence of interaction or interdependency between layered self-organizing mechanisms on performance and stability of the whole network system. We considered a scenario where an ad-hoc network and an overlay network constructed on the ad-hoc network, both of which adopt an attractor selection model-based routing mechanism while they have different goals, coexist and influence each other. To evaluate how interdependency among layered self-organizing control affects performance and stability, we considered four alternatives of coupling which differ in the way of activity sharing, that is underlay aware coupling, upper layer aware coupling, tight coupling and no coupling. Through simulation experiments, we verified that the upper layer aware coupling can provide the best performance in end-to-end communication delay. Intuitively speaking, the tight coupling leads to the best performance among the above four alternatives including the no coupling case. However, we found that the upper layer aware coupling achieve the best performance because the global optimization with multiple criteria is harder to achieve.

Through above studies, we proposed control mechanisms for self-organizing cooperation in wireless networks and showed some considerations on cooperation among self-organizing control mechanisms, which is essential to realize future networks suffering from complicated, unpredictable, and dynamically changing nature of application's behavior as well as that of wireless

network itself. Since our studies could cover only limited area of self-organizing cooperative networks, we need further exploration, where various coupling of control mechanisms, a variety of environmental changes, and heterogeneity in nodes, wireless links, and applications are considered. Although a lot of problems are left as open issues, we believe that those findings and discussion in this thesis will contribute to design and realization of future networks.

Bibliography

[1] MEXT Global COE Program, "Center of excellence for founding ambient information society infrastructure." http://www.ist.osaka-u.ac.jp/GlobalCOE (accessed on June 1, 2011).

[2] K. Ducatel, M. Bogdanowicz, F. Scapolo, J. Leijten, and J. Burgelman, "Scenarios for ambient intelligence in 2010," *IST Advisory Group Final Report*, Feb. 2001.

[3] D. Preuveneers, J. Van den Bergh, D. Wagelaar, A. Georges, P. Rigole, T. Clerckx, Y. Berbers, K. Coninx, V. Jonckers, and K. De Bosschere, "Towards an extensible context ontology for ambient intelligence," *Lecture Notes in Computer Science*, pp. 148–159, Oct. 2004.

[4] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, pp. 2787–2805, Oct. 2010.

[5] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications*, vol. 30, pp. 2826–2841, Oct. 2007.

[6] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Boloni, and D. Turgut, "Routing protocols in ad hoc networks: A survey," *Computer Networks*, vol. 55, pp. 3032–3080, Sept. 2011.

[7] B. Latré, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," *Wireless Network*, vol. 17, pp. 1–18, Jan. 2011.

[8] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Computer Networks*, vol. 54, pp. 881–900, Apr. 2010.

[9] M. Meisel, V. Pappas, and L. Zhang, "A taxonomy of biologically inspired research in computer networking," *Computer Networks*, vol. 54, pp. 901–916, Apr. 2010.

[10] T. Nakano, "Biologically inspired network systems: A review and future prospects," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, pp. 630–643, Sept. 2011.

[11] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393–422, Mar. 2002.

[12] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, pp. 2292–2330, Aug. 2008.

[13] F. Dressler, I. Dietrich, R. German, and B. Kruger, "A rule-based system for programming self-organized sensor and actor networks," *Computer Networks*, vol. 53, pp. 1737–1750, July 2009.

[14] I. Chlamtac, M. Conti, and J. J.-N. Liu, "Mobile ad hoc networking: imperatives and challenges," *Ad Hoc Networks*, vol. 1, pp. 13–64, July 2003.

[15] S. Sesay, Z. Yang, and J. He, "A survey on mobile ad hoc wireless network," *Information Technology Journal*, vol. 3, pp. 168–175, May 2004.

[16] H. Yamamoto, K. Hyodo, N. Wakamiya, and M. Murata, "Implementation and evaluation of a reaction-diffusion based coding rate control mechanism for camera sensor networks," *Technical Report of IEICE* (IN2008-6), vol. 108, pp. 31–36, May 2008.

[17] H. Yamamoto, K. Hyodo, N. Wakamiya, and M. Murata, "Implementation and evaluation of a reaction-diffusion based coding rate control mechanism for camera sensor networks," in *Proceedings of 2nd ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC)*, (California, USA), Sept. 2008.

[18] H. Yamamoto, K. Hyodo, N. Wakamiya, and M. Murata, "A reaction-diffusion based coding rate control mechanism for camera sensor networks," *Sensors*, vol. 10, pp. 7651–7673, Aug. 2010.

[19] A. Ganz, Z. Ganz, and K. Wongthavarawat, *Multimedia Wireless Networks: Technologies, Standards and QoS*. Springer, Sept. 2003.

[20] M. Watson, M. Luby, and L. Vicisano, "Forward error correction (FEC) building block," *RFC 5052*, Aug. 2007.

[21] B. W. Wah, X. Su, and D. Lin, "A survey of error-concealment schemes for real-time audio and video transmissions over the internet," in *Proceedings of the 2000 International Conference on Microelectronic Systems Education (MSE 2000)*, (Taipei, Taiwan), pp. 17–24, Dec. 2000.

[22] A. M. Turing, "The chemical basis of morphogenesis," *Royal Society of London Philosophical Trans. Series B*, vol. 237, pp. 37–72, Aug. 1952.

[23] H. Yamamoto, N. Wakamiya, and M. Murata, "Proposal and evaluation of a stepwise synchronization-based inter-networking mechanism for wireless sensor networks," *Technical Report of IEICE* (USN2010-4), vol. 110, pp. 17–20, May 2010.

[24] H. Yamamoto, N. Wakamiya, and M. Murata, "Proposal and evaluation of an inter-networking mechanism using stepwise synchronization for wireless sensor networks," *Technical Report of IEICE* (USN2010-40), vol. 110, pp. 13–18, Jan. 2011.

[25] H. Yamamoto, N. Wakamiya, and M. Murata, "An inter-networking mechanism using stepwise synchronization for wireless sensor networks," in *Proceedings of the 1st Workshop on Bio-inspired Models and Technologies for Ambient Information Society (BioAmbIS)*, (Boston, USA), pp. 17–21, Dec. 2010.

[26] H. Yamamoto, N. Wakamiya, and M. Murata, "An inter-networking mechanism with stepwise synchronization for wireless sensor networks," *Sensors*, vol. 11, pp. 8241–8260, Aug. 2011.

[27] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proceedings of IEEE INFOCOM 2002*, (New York, USA), pp. 1567–1576, June 2002.

[28] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble mac protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys)*, (Boulder, USA), pp. 307–320, Oct. 2006.

[29] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, pp. 1645–1662, Mar. 1990.

[30] H. Yamamoto, N. Wakamiya, and M. Murata, "Study on interaction between layered self-organization based control," in *Proceedings of 4th international symposium on applied sciences in biomedical and communication technologies (ISABEL)*, (Barcelona, Catalonia, Spain), Oct. 2011.

[31] N. Asvarujanon, K. Leibnitz, N. Wakamiya, and M. Murata, "Robust and adaptive mobile ad hoc routing with attractor selection," in *Proceedings of the 4th International workshop on Adaptive and DependAble Mobile Ubiquitous Systems (ADAMUS)*, (Berlin, Germany), July 2010.

[32] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, pp. 921–960, Mar. 2007.

[33] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Advances in Multimedia*, vol. 2009, May 2009.

[34] Y. Charfi, N. Wakamiya, and M. Murata, "Challenging issues in visual sensor networks," *Wireless Communication*, vol. 16, pp. 44–49, Apr. 2009.

[35] ZigBee Alliance, "Zigbee specification," June 2005. ZigBee Document 053474r06, Version 1.0.

[36] J. Li, C. Blake, D. S. D. Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *Proceedings of 7th Annual International Conference on Mobile Computing and Networking (MobiCom 2001)*, (Rome, Italy), pp. 61–69, July 2001.

[37] Ozturk, T. Hayashi, T. Yamasaki, and K.Aizawa, "Content-aware control for efficient video transmission of wireless multi-camera surveillance systems," in *Proceedings of 1st ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2007)*, (Vienna, Austria), pp. 394–395, Sept. 2007.

[38] X. Zhu, E. Setton, and B. Girod, "Rate allocation for multi-camera surveillance over an ad hoc wireless network," in *Proceedings of 24th Picture Coding Symposium (PCS 2004)*, (California, USA), Sept. 2004.

[39] Y. Chen and T. C. Henderson, "S-nets: Smart sensor networks," in *Proceedings of 7th International Symposium on Experimental Robotics (ISER 2000)*, (Hawaii, USA), pp. 85–94, Dec. 2000.

[40] M. Durvy and P. Thiran, "Reaction-diffusion based transmission patterns for ad hoc networks," in *Proceedings of IEEE INFOCOM 2005*, (Florida, USA), pp. 2195–2205, Mar. 2005.

[41] K. Hyodo, N. Wakamiya, E. Nakaguchi, M. Murata, Y. Kubo, and K. Yanagihara, "Experiments and considerations on reaction-diffusion based pattern generation in a wireless sensor network," in *Proceedings of 1st IEEE International Workshop: From Theory to Practice in Wireless Sensor Networks (t2pWSN 2007)*, (Helsinki, Finland), June 2007.

[42] A. Yoshida, K. Aoki, and S. Araki, "Cooperative control based on reaction-diffusion equation for surveillance system," in *Proceedings of 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005)*, (Melbourne, Australia), pp. 533–539, Sept. 2005.

[43] T. Nakamura, N. Mine, E. Nakaguchi, A. Mochizuki, M. Yamamoto, K. Yashiro, C. Meno, and H. Hamada, "Generation of robust left-right asymmetry in the mouse embryo requires a self-enhancement and lateral-inhibition system," *Dev. Cell*, vol. 11, pp. 495–504, Oct. 2006.

[44] "MPEG.ORG." http://www.mpeg.org/ (accessed on April 1, 2010).

[45] Z. Chen and K. N. Ngan, "Recent advances in rate control for video coding," *Signal Processing: Image Communication*, vol. 22, no. 1, pp. 19–38, 2007.

[46] P. Kersch, R. Szabo, and Z. L. Kis, "Self organizing ambient control space - an ambient network architecture for dynamic network interconnection," in *Proceedings of the 1st ACM Workshop on Dynamic Interconnection of Networks (DIN)*, (Cologne, Germany), pp. 17–21, Sept. 2005.

[47] E. Poorter, B. Latre, I. Moerman, and P. Demeester, "Symbiotic networks: Towards a new level of cooperation between wireless networks," *International Journal of Wireless Personal Communications*, vol. 45, pp. 479–495, June 2008.

[48] S. Jung, A. Chang, and M. Gerla, "Comparisons of ZigBee personal area network (PAN) interconnection methods," in *Proceedings of IEEE International Symposium on Wireless Communication Systems 2007 (ISWCS)*, (Trondheim, Norway), pp. 337–341, Oct. 2007.

[49] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal, "Firefly-inspired sensor network synchronicity with realistic radio effects," in *Proceedings of the 3rd ACM International Conference on Embedded Networked Sensor Systems (SenSys)*, (San Diego, USA), pp. 142–153, 2005.

[50] Y.-W. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 1085–1099, May 2005.

[51] J. Degesys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: Self-organizing desynchronization and TDMA on wireless sensor networks," in *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, (Massachusetts, USA), pp. 11–20, Apr. 2007.

[52] D. Lucarelli and I.-J. Wang, "Decentralized synchronization protocols with nearest neighbor communication," in *Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys)*, (Maryland, USA), pp. 62–68, Nov. 2004.

[53] P. Goel and B. Ermentrout, "Synchrony, stability, and firing patterns in pulse-coupled oscillators," *Physica D: Nonlinear Phenomena*, vol. 163, pp. 191–216, Mar. 2002.

[54] Y. Kuramoto, "Collective synchronization of pulse-coupled oscillators and excitable units," *Physica D: Nonlinear Phenomena*, vol. 50, pp. 15–30, May 1991.

[55] S. Kashihara, N. Wakamiya, and M. Murata, "Implementation and evaluation of a synchronization-based data gathering scheme for sensor networks," in *Proceedings of IEEE International Conference on Communications, Wireless Networking (ICC)*, (Seoul, Korea), pp. 3037– 3043, May 2005.

[56] A. Mutazono, M. Sugano, and M. Murata, "Evaluation of robustness in time synchronization for sensor networks," in *Proceedings of the 2nd International Conference on Bio-Inspired Models of Network (BIONETICS)*, (Budapest, Hungary), pp. 89–92, Dec. 2007.

[57] N. Wakamiya and M. Murata, "Synchronization-based data gathering scheme for sensor networks," *IEICE Transactions on Communications*, vol. 88, no. 3, pp. 873–881, 2005.

[58] N. Wakamiya and M. Murata, "Attractor composition-based self-adaptation in layered sensor-overlay networks," in *Proceedings of 2009 International Symposium on Nonlinear Theory and its Applications (NOLTA)*, (Sapporo, Japan), Oct. 2009.

[59] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, pp. 255–312, Jan. 2007.

[60] A. Kashiwagi, I. Urabe, K. Kaneko, and T. Yomo, "Adaptive response of a gene network to environmental changes by fitness-induced attractor selection," *PLoS ONE*, vol. 1, Dec. 2006.

[61] K. Leibnitz, N. Wakamiya, and M. Murata, "Biologically inspired self-adaptive multi-path routing in overlay networks," *Communications of the ACM*, vol. 49, pp. 62–67, Mar. 2006.

[62] E. Sakhaee, N. Wakamiya, and M. Murata, "Layered attractor selection for clustering and data gathering in wireless sensor networks," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, (Sydney, Australia), pp. 1–6, Apr. 2010.

[63] G. Nishikawa, F. Ooshita, H. Kakugawa, and T. Masuzawa, "A stable clustering algorithm for mobile ad hoc networks based on attractor selection," in *Proceedings of 3rd International Conference on Bio-Inspired Models of Network, Information, and Computing Systems (BIONETICS)*, (Hyogo, Japan), pp. 38:1–38:6, Nov. 2008.

[64] T. Iwai, N. Wakamiya, and M. Murata, "Error-tolerant coverage control based on bio-inspired attractor selection model for wireless sensor networks," in *Proceedings of International Workshop on Intelligent Communication Networks (IntelNet)*, (Bradford, UK), pp. 723–729, June 2010.

[65] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," *RFC 3561*, July 2003.

[66] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*, pp. 15–34, Springer, 2010.