Ph.D. Degree Program

2020 年 3 月，関西大学審査学位論文

Mobile Sensing Cluster Based on Swarm Intelligence with Multiple Autonomous Mobile Devices

Kansai University

Graduate School of Science and Engineering Major

Civil, Environment and Applied Systems Engineering

17D6008　Eiji NII

Contents

List of Figures

List of Tables

Abstract

In the near future as alternatives to humans, such autonomous mobile devices as robots and unmanned aerial vehicles are expected to search for and actuate diverse emergent events whose location and number in occurrences are unknown. When an autonomous mobile device searches for and actuates an unknown event, it seeks events based on sensing the physical information emitted by them, such as smell and temperature, and when reaching an event, it processes them based on its actuating function. Typical examples of unknown events include, in a disaster damaged structures, rescueers, and target resources in the resource search. Based on the property of the event, an autonomous mobile device is required to finish the search for and actuate a larger number of events within a limited time.

To achieve the above issue, most of existing methods have improved the performance of the functions in a single device. However, the single device scheme is restricted by its performance; to overcome such restrictions, a multiple-device scheme is necessary. Consider multiple autonomous mobile-device schemes that search for and actuate multiple unknown events. The approach is divided into independent and cooperative schemes. In the independent search scheme, each device independently searches for and actuates unknown events. Although such schemes have high parallelism to search for and actuate events, their searching and actuating capacities depend on capacity of a single device. In the cooperative scheme, each device collaboratively searches for and actuates unknown events by sharing its information. Although this scheme creates high capability to search for and actuate events, searching for and actuating multiple unknown events is sequentially processing, and so it spends too much time on searching for and actuating multiple events. Therefore, the scheme in searching for and actuating multiple unknown events with multiple autonomous mobile devices is required that it appropriately balances parallelism and collaboration.

Therefore, this thesis proposes a Mobile Sensing Cluster (MSC), which expands Particle Swarm Optimization (PSO) and dynamically forms multiple swarms with multiple autonomous mobile de-

vices to achieve appropriate balancing parallelism and collaboration when searching for and actuating multiple unknown events.

Chapter 1

Introduction

In the near future as alternatives to humans, autonomous mobile devices, such as robots and unmanned aerial vehicles, are expected to be used to search for and actuate diverse emergent events whose locations and numbers in occurrences are unknown[1][2]. When an autonomous mobile device searches for and actuates an unknown event, it seeks an event based on sensing such physical information emitted by the event as smell and temperature. When it reaches the event, it processes the event based on its actuating function. Typical examples of an unknown event include, in a disaster, a damaged part of a structure, rescueers, and a target resource in a resource search. Based on the event's property, an autonomous mobile device is required to finish the search for and actuate more events within a limited time.

To achieve the above issue, most of existing methods focus on improving the performance functions in a single device [3], such as driving function, sensing function, etc. However, the single device scheme is restricted by its performance, and to overcome such restrictions, a multiple-device scheme is necessary. Consider multiple autonomous mobile-device schemes that search for and actuate multiple unknown events. The approach is divided into independent and cooperative schemes. In the independent search scheme, each device independently searches for and actuates unknown events. Although such schemes have high parallelism to search for and actuate events, their searching and actuating capacities depend on capacity of a single device. In the cooperative scheme, each device collaboratively searches for and actuates unknown events by sharing its information. Although this scheme creates high capability to search for and actuate events, searching for and actuating multiple unknown events is sequentially processing, and so it spends too much time on searching for and actu-

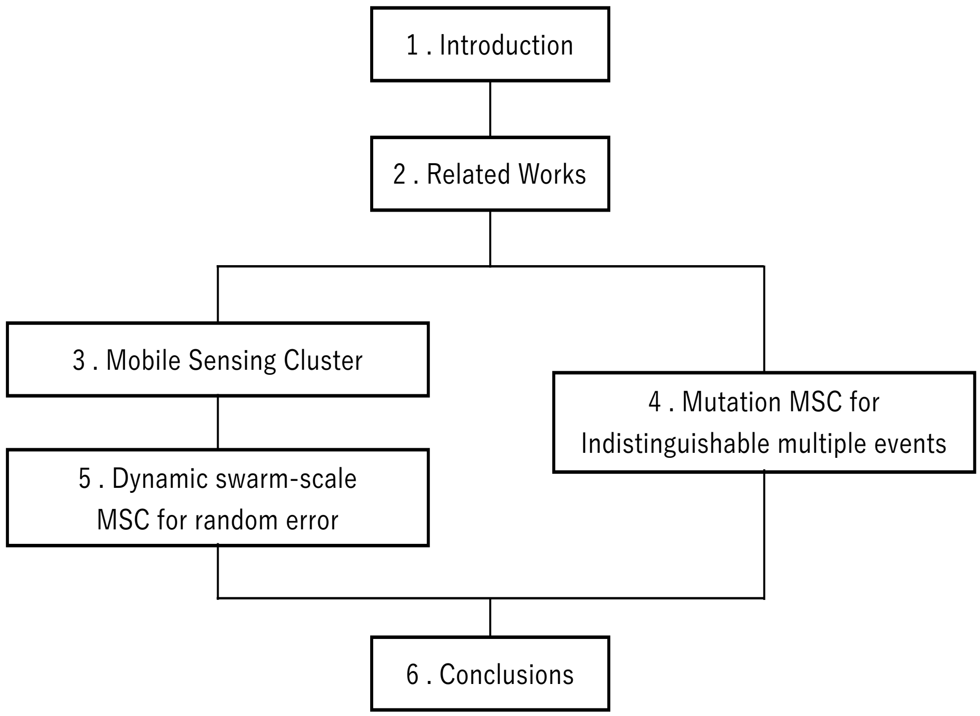


Figure 1.1: Outline.

ating multiple events. Therefore, the scheme in searching for and actuating multiple unknown events with multiple autonomous mobile devices is required that it appropriately balances parallelism and collaboration.

This thesis proposes a Mobile Sensing Cluster (MSC) to search for and actuate a large number of events whose location and number in occurrences are unknown for a limited time.

MSC expands Particle Swarm Optimization (PSO)[4] to achieve appropriate balancing parallelism and collaboration in searching for and actuating multiple unknown events and can dynamically form multiple swarms. In MSC with wireless communications, each devise shares searching for and actuating information among its neighbors and forms multiple swarms with devicesj to search for and actuate multiple unknown events. In this thesis, three schemes are proposed for MSC and applied to the following assumptions for events:

- The strength of the physical information emitted by an event increases accordingly as approaching event, and the events are distinguishable from each other based on its emitting physical information such as radio waves, which are artificially structuralized for data communications and event's identify is inserted into the structures.

- The strength of the physical information emitted by an event monotonically increases accordingly as approaching event, and the events are indistinguishable from each other based on its emitting physical information such as natural phenomenon, which is unstructuralized and includes no information that can distinguish events.
- The strength of the physical information emitted by an event includes random error, for example, the error is fading of radio waves, and fluctuates complexly by the error and the events are distinguishable from each other based on its emitting physical information.

Fig.1.1 shows the outline of this thesis. Chapter 2 describes related work, and chapter 3 proposes an MSC for the strength of the physical information emitted by an event that monotonically increases according as approaching an event and a distinguishable event from each other. In chapter 4, a mutation MSC for the strength of the physical information emitted by an event that monotonically increases accordingly as approaching an event but an indistinguishable event from each other is proposed. Chapter 5 proposes a dynamic swarm-scale MSC for the strength of the physical information from events with random error and a distinguishable event from each other. Finally in chapter 6, conclusions are shown.

Chapter 2

Related Works

This chapter focuses on previous research in swarm intelligence and swarm robotics.

2.1 Swarm Intelligence

Swarm intelligence (SI), which is a kind of an artificial intelligence, is an optimization algorithm based on the behavior of animals or insects [5][6][7]. An SI system usually consists of a group of simple individuals autonomously controlled by a plain set of rules and local interaction [8][9]. A swarm can complete the tasks that a complex individual can do with high robustness, flexibility, and low cost [10][11].

As an example of SI, algorithms based on schools of fish, ant colonies, and bee colonies are shown below.

2.1.1 Particle swarm optimization

Particle swarm optimization (POS) [12][13], inspired by the swarm behavior of flocks of birds and schools of fish, is a mathematical search model based on multiple particles. Each particle has a location and a velocity, and its own location is evaluated using a fitness function. The velocity of each particle is derived by its personal and global bests(Fig.2.1). The former is its best previous location, and the latter is the best previous location of all the particles [14][15][16].

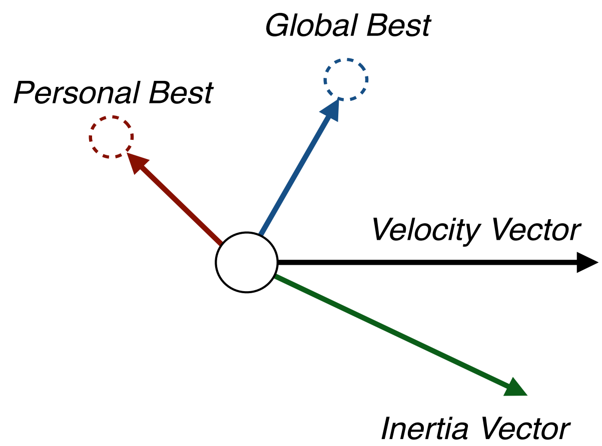


Figure 2.1: Update rule of PSO.

2.1.2 Reynolds flocking model

The Reynolds flocking model [17], which simulates the swarming behavior of birds, was introduced in 1987. Each agent moves based on the following three rules [18][19](Fig.2.2):

- alignment: agents adjust their velocity to that of their neighbor agents;
- cohesion: agents are attracted to the average position of their neighboring agents;
- separation: agents are repulsed by their neighboring agents.

2.1.3 Ant colony optimization

Ant colony optimization (ACO) presents a main class of stochastic search algorithms that was introduced for solving difficult optimization problems [20][21][22]. ACO was inspired by the foraging behavior of ant colonies. It is based on the indirect communication among ants mediated by trails of a chemical substance called pheromone, which allows them to solve the shortest path problems between their nests and food sources [23][24][25].

ACO calculates approximate solutions in traveling salesman problems and is especially effective for graphs that change dynamically [26]. The algorithm is expected to be applied to routing networks and traffic systems, etc.

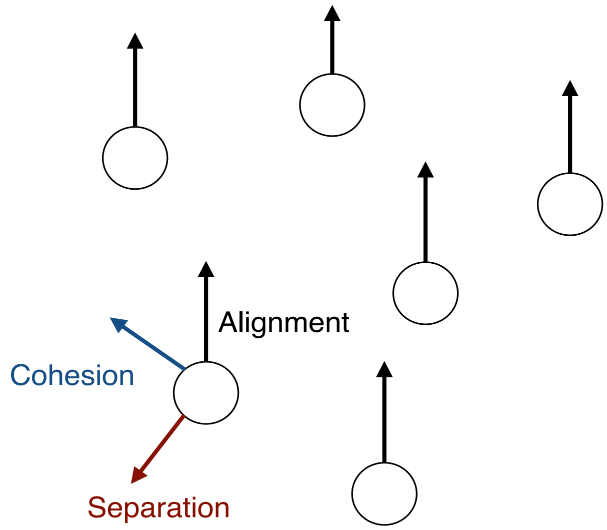


Figure 2.2: Three vectors of Reynolds Flocking Model.

2.1.4 Artificial bee colony

Artificial bee colony (ABC) is an algorithm that simulates the swarm intelligence integration of a bee's colony in foraging activities [27][28]. In ABC, bees are divided into three groups based on the division of labor: honey bees, followers, and scouts. The honey bee randomly searches for nectar within the defined domain and shares such information with the followers who select the best nectar in the vicinity. Scout bees are formed by honey bees that abandoned the nectar because they had already collected more than a predetermined number of cycles or only collected a small amount of honey [29][30].

2.2 Consensus Problem

Multiple agent systems, which cooperatively control arbitrary systems by multiple agents, are expected to be used in the field of sensor networks or to control autonomous robots. In such systems, the velocity of robots and the sensing data values converge to an arbitrary value called a consensus problem [31][32]. Only obtaining a consensus among multiple agents, the systems address the formation of a single swarm [33].

2.3 Genetic Algorithm and Mutation

The genetic algorithm (GA) is a meta-heuristic search algorithm that references natural selection and a genetic mechanism of nature [34][35]. As a computational model that simulates the biological evolution process of Darwin's genetic selection theory, it is a completely new global optimization algorithm. Its operation manages the objects of every member of the group and creates a new generation by choosing, crossing, and mutating [36][37].

Mutation prevents a collapse into a local solution by randomly replacing a chromosome's genetic information [38].

2.4 Swarm Robotics

2.4.1 Definition of swarm robotics

Swarm robotics is a new approach to the coordination of multi-robot systems that consist of large numbers of mostly simple physical robots [44]. A desired collective behavior emerges from the interaction among the robots and their interaction with the environment. This approach emerged in the field of artificial swarm intelligence as well as the biological study of insects, ants, and other fields in nature where swarm behavior occurs.

Swarm robotics research studies the design of a large amount of relatively simple robots, their physical bodies, and their controlling behaviors. The individuals in the swarm are generally uncomplicated, small, and inexpensive to exploit a large population [39]. A key component of the system is its communication between the agents in the group, which is normally local, and guarantees that the system is scalable and robust [40].

2.4.2 Comparison to a single robot system

To complete a complex task, a single robot must be designed with a complicated structure and control modules that increase the resulting design, construction, and maintenance costs [41][42]. A single robot is especially vulnerable because a small broken part might affect the whole system, and predicting the consequences is difficult. Swarm robotics can achieve the same ability through inter-group cooperation and exploiting the reusability of simple agents and low construction and maintenance costs [43].

A single robot is inspired from human behaviors by comparing the corresponding nature species of these research areas from which social animals inspired swarm robotics. Due to the restrictions of current technology, simulating human interactions using machines or computers is difficult; applying the mechanisms in animal groups is easier. This realization suggests a bright future for swarm robotics for dealing with complex and large-scale problems.

2.4.3 General model of swarm robotics

The swarm robotics model is a key component of cooperative algorithms that control the behaviors and interactions of all individuals. In it, the robots in the swarm require such basic functions as sensing, communication, motioning, etc.

The model is divided into three modules based on the functions utilized to accomplish the following three kinds of behavior: information exchange, basic, and advanced behavior. Information exchange is inevitable when the robots cooperate with one another, and is the core part for controlling swarm behaviors. The basic behavior of individuals include functions such as motioning and local planning which is one of most significant differences of swarm robotics than the multi-agent and sensor network systems. The advanced behavior is a task decomposition, task allocation, adaptive learning, etc. Among the three modules, information exchange plays the most critical role in the model. The robots in the swarm exchange information and distribute it to the whole swarm through autonomous behavior that results in swarm-level cooperation [45].

2.4.4 Existing projects

2.4.4.1 Swarm-bots project

Swarm-bots is defined as an artifact composed of a swarm of several mobile robots(called s-bots) that can operate both autonomously and as a group, and s-bots is a comprehensive study on autonomous self-assembly with a new collective and mobile reconfigurable robotic system [46][47]. Each s-bots is a fully autonomous mobile robot capable of performing based tasks such as autonomous navigation, perception of the environment and grasping of object. In addition to these features, an s-bots is able to communicate with other s-bots and physically connect to them in flexible ways, thus forming the swarm-bots. The project developed both simulation and entity robots and presented their results on the two platform [48][49].

2.4.4.2 Swarmanoid project

The Swarmanoid project has extended the work done in the Swarm-bots project to three dimensional environment. The project is an innovative swarm robotics system composed of three different robot types with complementary skills: foot-bots are small autonomous robots specialized in moving on both even and uneven terrains, capable of self-assembling and of transporting object or other robots; hand-bots are autonomous robots capable of climbing some vertical surfaces and manipulating small objects; and eye-bots are autonomous flying robots that can attach to an indoor ceiling, capable of analyzing the environment from a privileged position to collectively gather information inaccessible to foot-bots and hand-bots [50].

Swarmanoid exploits the heterogeneity and complementarity of its constituent robot types to carry out complex tasks in large, three-dimensional, man-made environments. Humanoid robots are usually assumed to be the most efficient robot type for man-made environments. One of the goals of the swarmanoid project was to refute this assumption. The system has no centralized control and relies on continued local and nonlocal interactions to produce collective self-organized behavior.

2.4.4.3 Pheromone robotics project

The Pheromone Robotics Project, started in 2000, is coordinated by Professor David. The project aims to provide a robust, scalable approach for achieving the swarm level behaviors using a large number of small-scale robots in surveillance, reconnaissance, hazard detection, path finding, payload conveyance and small-scale actuation [51]. The team exploited the notion of a virtual pheromone, and implemented the simple beacons and directional sensors mounted on each robot. The virtual pheromones only facilitate simple communication and coordination with little on-board processing [52].

2.4.4.4 iRobot swarm project

iRobot Swarm Project is projected by MIT for cooperating over 100 robots. The goal of the project is to develop the distributed algorithms for robotic swarms composed of hundreds of individual robots robust to complex real-world environment and tolerant to the addition or failure of any number of individuals. The project team has developed a global monitoring device and an automatic charging station. The most of work of the project was done by Mclurkin and his colleagues[53].

2.4.4.5 E-puck education robot

The main goal of E-puck education robot project is to develop a miniature mobile robot for education use. The robots have several features specialized for such purpose. The robots have a clean mechanical structure simple to understand, operate and maintain. The robots are cheap and flexible, and can cover a large spectrum of educational activities thanks to a large potential in sensors, processing power and extensions [54]. Researches based on e-puck project have already exceeded 60 by the end of 2010. The potential educational fields include mobile robotics, real-time programming, embedded system, signal processing, image or sound feature extraction, *humanemachine* interaction or collective system.

2.5 Comparison between search algorithms in related works

In this section, we compare the search algorithm in related works from the point of view of a searching unknown multiple events in real environment with following four items.

- Cooperative search : whether the algorithm can cooperate among multiple agents in searching an event.
- Collision avoidance : whether the algorithm consider collisions among agents.
- Continuous search : whether the algorithm can continuously search for multiple unknown events.
- Parallel search control : whether the algorithm can search for multiple events in parallel with emerged multiple swarms.

In Fig.2.1, the comparison between search algorithms in related works with the above items is shown.

Brute force search algorithm, such as Breath First Search[55] or Depth First Search[56], can search cooperatively with multiple agents, and it can consider the collision avoidance among agents.

However, the algorithm search all area, where events occur, therefore, its search area should be preset and constrained. Furthermore, it does not consider a continuous search and parallel search control.

PSO search for an event cooperatively among multiple agents by forming a swarm. However, since PSO is a mathematical search model, it does not consider physical restrictions, which are collisions between agents and the range of communication among them, and cannot search for multiple events continuously. In addition, PSO does not show the parallel search mechanism with multiple swarm.

The Reynolds flocking model has no function to search for an event because its algorithm maintains a swarm's form, which is organized by multiple agents.

ACO and ABC can search an event cooperatively with other agents. The both algorithms is a mathematical search model, so it does not consider collisions among agents. ABC can continuously search for multiple events. ACO and ABC don't have the parallel search control, so the both algorithms cannot search in parallel.

As described above, all the search algorithms in the comparison are short of the capabilities to search for multiple unknown events.

Table 2.1: Requirements to search for multiple solutions and comparison between search algorithm in related works.

	Cooperative among agents	Collision avoidance	Continuous serach	Parallel search control
Brute force search	○	○	×	×
PSO	○	×	×	×
Reynold Flocking Model	×	×	×	×
ACO	○	×	×	×
ABC	○	×	○	×

Chapter 3

Mobile Sensing Cluster

In this chapter, we propose MSC, and we discuss the following two mechanisms that comprise MSC:

- a search and actuation mechanism based on PSO for unknown events using wireless communications as the interaction between mobile devices;
- a dynamic multiple-swarming mechanism that extends PSO to achieve appropriate balancing of parallelism and collaboration in searching for and actuating multiple unknown events.

The search and actuation mechanism based on PSO utilizes wireless communications and forms swarm with multiple autonomous mobile devices. The dynamic multiple-swarming mechanism divides a swarm into multiple swarms or aggregates it with multiple swarms, based on searching for and actuating multiple unknown events.

3.1 MSC assumption

MSC assumes that an autonomous mobile device has the following functions:

- a drive function and can autonomously move;
- a sensor function with which it estimates its location;
- a wireless communication function to share searching and actuating information with its neighbors;

- a sensor function for physical information emitted by events, each of which can distinguish and sense the physical information's strength;
- an actuating function for events with which it can process or capture them;
- a recognition function that can determine whether it has reached an event. When it has, it can change from searching to its actuating function.
- changing from actuating to searching when it cannot sense physical information from event actuation: that is, when the event actuation has been processed or captured.

We assume the following physical information is emitted by an event:

- the physical information's strength emitted by an event monotonically increases as approaching an event.
- the physical information from an event includes information with which each device can be distinguished.

3.2 Search and actuation mechanism based on PSO using wireless communications

This section describes the PSO algorithm, the general MSC model based on wireless communications, the selection leader, the collision-avoidance control, and the continuous search control.

3.2.1 PSO algorithm

The PSO algorithm, which is a distributed algorithm and works in each particle, is formulated as follows:

$$\begin{aligned} v_i(t+1) = wv_i(t) + c_1 r_1 (x_i^{Pbest}(t) - x_i(t)) \\ + c_2 r_2 (x^{Gbest}(t) - x_i(t)) \end{aligned} \tag{3.1}$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \tag{3.2}$$

where t is an iteration, $v_i(t)$ is the velocity vector of particle i at iteration t, w is the weight of inertia vector, $x_i^{Pbest}(t)$ is a personal best location at iteration t, $x_i^{Gbest}(t)$ is a global best at iteration t, c_1 is a selfish parameter, c_2 is a social parameter, r_1 and r_2 are uniformly distributed random numbers, and $x_i(t)$ is the location of particle i at iteration t.

As described above, the particle in the PSO algorithm updates the vector and the location based on the personal and global bests to approach a solution: an event. In Eq.(3.1), the personal best creates selfish behavior based on the particle's information by the particle itself, and the global best creates unselfish behavior based on information from the entire swarm.

3.2.2 General MSC model

To search for and actuate unknown events in the real world, based on the PSO algorithm, the general MSC model works in each autonomous mobile device and is formulated as follows:

$$\begin{aligned} v_i(t+1) = wv_i(t) + pb_i(t)(x_i^{Pbest}(t) - x_i(t)) \\ +lb_i(t)(x^{Lbest}(t) - x_i(t)) \end{aligned} \tag{3.3}$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \tag{3.4}$$

where t is time, $v_i(t)$ is the velocity vector of device i at time t, $pb_i(t)$ is the weight of the personal best, $lb_i(t)$ is the weight of the local best, $x_i^{Pbest}(t)$ is the personal best's location, and $x_i^{Lbest}(t)$ is the best location of the neighbor devices, which will be described in Eq.(3.6).

The personal best shows a location where an autonomous mobile device has determined based on its own sensing of the physical information's strength from the event(Fig.3.1):

- If an autonomous mobile device determines that it is approaching an event, its personal best is randomly updated around the current moving direction.

- Otherwise, the personal best is randomly updated around the opposite direction to the current moving direction:

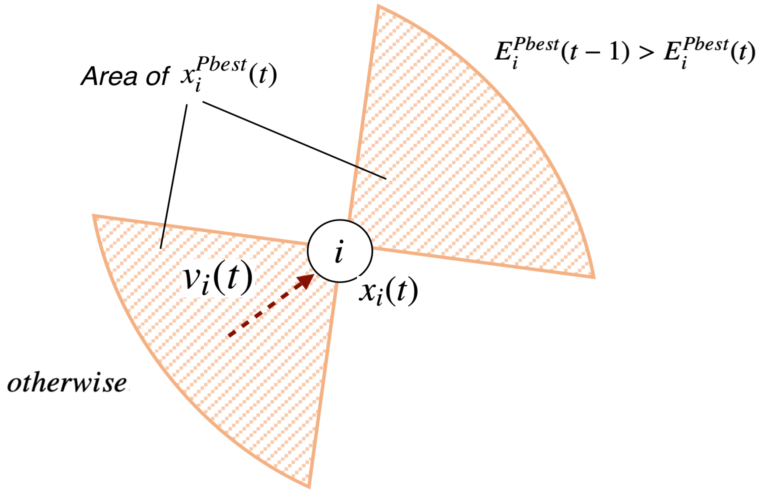


Figure 3.1: Area of the personal best.

$$x_i^{Pbest}(t) = \begin{cases} ||v_i(t-1)||(\cos(\alpha+\beta), \sin(\alpha+\beta)) + x_i(t) \\ \quad if \;\; E_i^{Pbest}(t-1) > E_i^{Pbest}(t) \\ -||v_i(t-1)||(\cos(\alpha+\beta), \sin(\alpha+\beta)) + x_i(t) \\ \quad otherwise. \end{cases} \tag{3.5}$$

Here $E_i^{Pbest}(t)$ is the personal best evaluation value of device i at time t, α is the angle of $v_i(t-1)$ with axis x, and β is a random angle in $[-\theta, \theta]$.

The personal best evaluation value corresponds to the distance between a device and an event, and the distance is derived by the device based on its sensing of the physical information's strength from an event. The details in the personal best evaluation value are described in section 3.2.3.

MSC assumes that the device interacts with each other by wireless communications. Therefore, the interaction among devices is restricted by the range of wireless communications and to local areas. Accordingly, in Eq.(3.3) of MSC, the third item is not the global best; it is the local best. The local best, which is the location of a neighbor nearest an event, is formulated:

$$x_i^{Lbest}(t) = X\left(l | l = \min_{j \in neighbor_i(t)} E_j^{Pbest}(t)\right), \tag{3.6}$$

where $X(l)$ is location of device l and $neighbor_i(t)$ is a set of devices that are found at t as neighbor devices of device i. The distance between a neighbor and an event is expressed by the local best evaluation value whose details are also described in section 3.2.3.

3.2.3 Evaluation value

To derive the personal and local bests, MSC defines three evaluation values: personal best, local best, and self-evaluation values.

- The personal best evaluation, whose value shows the distance from the nearest event in all the events discovered by the device, is derived as follows:

$$E_i^{Pbest}(t) = \min_{k \in discovery_i(t)} E_i^K(t), \tag{3.7}$$

  where $discovery_i(t)$ is a set of discovered events by device i at t and $E_i^K(t)$ is an evaluation value showing the distance from event K based on the sensing of physical-information strength from K in device i at t.

- The local best evaluation value shows the minimum distance in a set of the distance between a neighbor and an event and is derived based on the self-evaluation value, which shows the distance to an event in each device:

$$E_i^{Lbest}(t) = \min_{j \in neighbor_i(t)} E_j(t), \tag{3.8}$$

  where $E_i^{Lbest}(t)$ is the local best evaluation value of device i at t and $E_j(t)$ is a self-evaluation value of device j at t.

- A self-evaluation value shows the distance to an event. If the personal best evaluation value is less than the personal best evaluation values of the neighbors in the wireless communications range, the self-evaluation value is the personal best evaluation value; otherwise, it is the sum of the local best evaluation value and the distance to the local best, which is a location of a neighbor that corresponds to the local best:

$$E_i(t) = \begin{cases} E_i^{Pbest}(t) \\ \quad if \;\; E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} E_j^{Pbest}(t) \\ E_i^{Lbest} + C_i^{Lbest}(t) \\ \quad \text{otherwise.} \end{cases} \tag{3.9}$$

Here $E_i(t)$ is the self-evaluation value of device i at t, and $C_i^{Lbest}(t)$ is the distance to the local best of device i at t.

3.2.4 Selection of a leader to form a swarm

MSC selects a device with a minimum personal best value for an event as a swarm's leader, which only moves based on the personal best value. The devices other than the leader (followers) just move based on the local best value; that is, the leader selfishly moves to an event, and the followers follow the leader, and a swarm that searches for an event is formed by the leader and the followers. To realize emergence of the above behavior and formation of a swarm, the weights of the personal and local best values are derived:

$$pb_i(t) = \begin{cases} 1 & if \;\; E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} E_j^{Pbest}(t) \\ 0 & otherwise. \end{cases} \tag{3.10}$$

$$lb_i(t) = \begin{cases} 0 & if \;\; E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} E_j^{Pbest}(t) \\ 1 & otherwise. \end{cases} \tag{3.11}$$

Based on Eqs.(3.10) and (3.11), the leader dynamically changes, and a swarm is always formed by selecting the nearest device to an event as the leader.

3.2.5 Collision-avoidance control

MSC extends collision avoidance in the Reynolds flocking model. All devices have collision-avoidance vectors that repel other devices. A collision-avoidance vector is derived from the distance between itself and its neighbor devices. The vector, which becomes a strong repulsion vector as a

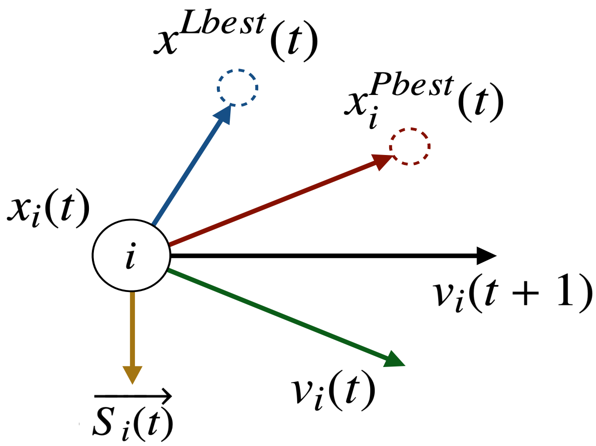


Figure 3.2: The vector and location update rules, including the collision-avoidance vectors.

device moves closer to its neighbor, is derived:

$$\overrightarrow{S_i(t)} = c_3^i \sum_{j \in n} \frac{\overrightarrow{V_{ji}(t)}}{\|V_{ji}(t)\|(d_{ij}(t))^k}, \tag{3.12}$$

where c_3^i is the avoidance weight of device i, $\overrightarrow{V_{ji}(t)}$ is the velocity vector to device i from device j, n is a set of the neighbor devices of device i, d_{ij} is the distance between devices i and j, and k is the avoidance degree.

The vector and location update rules, including the collision-avoidance vectors, are described(Fig.3.2):

$$\begin{aligned} v_i(t+1) = wv_i(t) + pb_i(t)(x_i^{Pbest}(t) - x_i(t)) \\ +lb_i(t)(x^{Lbest}(t) - x_i(t)) + \overrightarrow{S_i(t)} \end{aligned} \tag{3.13}$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \tag{3.14}$$

where $S_i\vec{(}t)$ is the collision-avoidance vector.

3.2.6 Search and actuation phases

MSC repeatedly switches the behavior between the search and actuation phases. In the search phase, as described above, devices search for events by communicating with other neighbor devices based on Eqs.(3.3) and (3.4). If the device senses the strength of the physical information above a threshold, it determines that it has reached an event and enters the actuation phase.

In the actuation phase, to stay within a range where the physical information is strong above a threshold, the device decelerates and adjusts the distance among its neighbors to evenly disperse them. The velocity vector in Eq.(3.3) and the collision-avoidance weight in Eq.(3.12) are derived:

$$c_3^i = \begin{cases} c_3^{Search} & if \;\; E_i > T \\ c_3^{Search}/n & otherwise. \end{cases} \tag{3.15}$$

$$v_i(t) = \begin{cases} \dfrac{v_i(t)}{|v_i(t)|} M^{upper} & if \;\; |v_i(t)| > M^{upper} \\ v_i(t) & otherwise, \end{cases} \tag{3.16}$$

where c_3^{Search} is the separation weight in the search phase, n is an integer, T is a threshold entering the actuation phase, and M^{upper} is the upper limit of the velocity per second.

In the actuation phase, if a device becomes unable to sense the physical information from an event within a certain period, it determines that the actuation of an event is completed. Then, to search for other events, it discards the current evaluation values and returns to the search phase.

3.2.7 Continuous search mechanism

If multiple events exist, continuous search and actuation are required for multiple events after a swarm has completely searched and actuated an event. When PSO completes its search for a solution, it ends its search; it does not search continuously.

If a device does not continuously sense the physical information during a period in the actuation phase, MSC discards the personal and local best evaluation values, switches behavior to the search phase, and restarts to search for other events.

3.2.8 Wireless communications among multiple mobile devices

MSC uses wireless communications to interact with devices based on the information among them. Each device advertises the following information and shares it among neighboring devices with wireless communications:

- self-location
- personal best evaluation value
- self-evaluation value.

Each device utilizes the above interactive information among neighbors for updating their locations, their best evaluation values, and the leader's selection.

3.3 Dynamic multiple-swarming mechanism

MSC dynamically forms multiple swarms to search for and actuate multiple events in parallel. The dynamic multiple-swarm mechanism selects a leader for individual events to realize emergence of the division behavior in swarms. Furthermore, to realize emergence of the behavior that boosts the swarm's division and formation of an impartial swarm size among multiple swarms, MSC introduces an event-crowd degree for deriving the personal best evaluation value and a neighbor-crowd degree for deriving the local best evaluation value.

3.3.1 Multiple leaders for dividing a swarm into multiple swarms

As described in section 3.2.4, only one device is selected as a leader in a swarm. The dynamic, multiple-swarming mechanism selects multiple leaders to search for and actuate multiple events. To divide a swarm into multiple swarms based on multiple events, a leader is selected for individual events:

- Each device selects an event with a minimum personal best value in all the sensing events.
- Each device advertises the selecting event and its personal best evaluation value to its neighbors.

- Each device scans the advertisements from its neighbors and compares its personal best evaluation value for the selected event with those of all its neighbors.

- The devices with minimum personal best evaluation value behave as leaders.

According to the above selection, the weights of the personal and local best values are revised:

$$pb_i(t) = \begin{cases} 1 & if \;\; E_i^{Pbest(K)}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest(K)}(t)\} \\ 0 & otherwise. \end{cases} \tag{3.17}$$

$$lb_i(t) = \begin{cases} 0 & if \;\; E_i^{Pbest(K)}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest(K)}(t)\} \\ 1 & otherwise. \end{cases} \tag{3.18}$$

Here $E_i^{Pbest(K)}(t)$ is a personal best evaluation value of device i for event K at t.

The event-crowd degree derives a personal best value to control the number of devices in a swarm. The event-crowd degree for event K accords with the number of neighboring devices in a swarm that is approaching event K. By applying the event-crowd degree to the personal best evaluation value, that is, by adding the event-crowd degree to the personal best evaluation value, the personal best evaluation value for events approached by many devices becomes high; another new leader can be selected to search for other events in a swarm that divides it into multiple swarms. The event-crowd degree and personal best evaluation value to which the event-crowd degree is applied are derived:

$$D_i^k(t) = \{x | x \in neighbor_i(t), P^k(x, t)\} \tag{3.19}$$

$$E_i^{Pbest(K)}(t) = \min_{k \in discovery_i(t)} \{E_i^{Pbest(k)}(t) + c_4 |D_i^K(t)|\}, \tag{3.20}$$

where $P^K(x, t)$ is a set of devices approaching event K at t, $D_i^K(t)$ is a set of event-crowd degrees for event K of device i, $|D_i^K(t)|$ is the number of elements of $D_i^K(t)$, and c_4 is a parameter of a coefficient of the event-crowd degree.

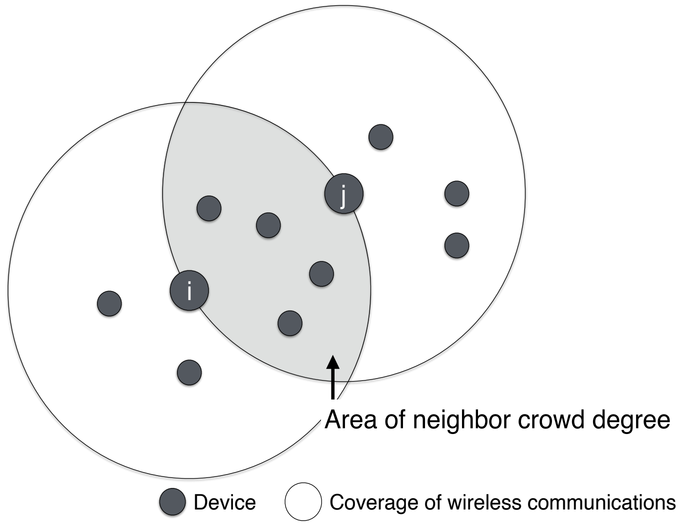


Figure 3.3: Area of neighbor-crowd degree.

3.3.2 Impartial swarm size among multiple swarms

To optimize the search and actuation mechanism based on multi-swarms, the swarm size, which is the number of devices that form a swarm, must be impartial among multiple swarms. To make the number of followers uniform among multiple swarms, by using the neighbor-crowd degree, the local best evaluation value is derived. The neighbor-crowd degree accords with the number of devices between itself and its neighboring device (Fig.3.3). If the neighbor-crowd degree for a neighbor is large, that is, the swarm among the neighbors is crowded, the device follows another device with a lower neighbor-crowd degree. The local best evaluation value with the neighbor-crowd degree is derived so that the above behavior emerges:

$$N_i^j(t) = \{x | x \in neighbor_i(t) \cap neighbor_j(t)\} \tag{3.21}$$

$$E_i^{Lbest}(t) = \min_{j \in neighbor_i(t)} \{E_j(t) + c_4 |N_i^j(t)|\}, \tag{3.22}$$

where $N_i^j(t)$ is a neighbor-crowd degree of device i for neighbor device j at t and $|N_i^j(t)|$ is the number of elements of $N_i^j(t)$.

3.4 Simulation evaluation

This simulation validates the effectiveness of each proposal mechanisms, which are described in chapter3.

Table 3.1: Simulation parameters of evaluation for MSC.

Parameters	Values
Simulator	ns3
Simulation time (sec)	5000
Number of trials for each simulation scenario	10
Number of devices	10～40
Number of events	10～40
Initial location of devices	Origin (0,0), Radius 50 m
	Uniform distribution
Initial location of events	Origin (70,70), Radius 50 m
	Uniform distribution
Actuation capacity of an event	300
Update cycle of velocity vector (sec)	0.1
T (dBm) in Eq.3.15	-50.6262
w	0.5
pb	1
lb	1
c_3^{Search}	25
$c_3^{Capture}$	5
M_{Search}^{upper}(m/sec)	1
$M_{Capture}^{upper}$(m/sec)	0.3
k in Eq.(3.12)	2
$\theta(^\circ)$ in Eq.(3.5)	30
Period without sensing physical information from event	
to discard evaluation values in actuation phase (sec)	1
Coefficient of neighbor-crowd degree c_4	-10

Table 3.1: (cont'd) Simulation parameters of evaluation for MSC.

Parameters	Values
Wireless communications	IEEE802.11b
Transmission power (dBm)	17.0206
Path loss (dB)	$L_0 + 10n\log_{10}(\frac{d}{d_0})$
Path loss at reference distance L_0 (dB)	-46.6777
Reference distance d_0 (m)	1
Propagation loss factor n	3
Radio wave reach distance (m)	250
Effective actuation radius (m)	5
D_c(m)	1

3.4.1 Simulation specifications

The simulation parameters are listed in Table 3.1. The devices and events are defined:

- A device is equipped with an IEEE802.11b interface and periodically advertises its information (section 3.2.8).
- An event is equipped with an IEEE802.11b interface and periodically advertises a beacon.
- Each device distinguishes among events by MAC addresses in the beacon advertised from events.

Each device receives information from its neighboring devices and beacons from events and derives the following three evaluation values:

- Personal best evaluation value (E_i^{Pbest})

  Based on Eqs.(3.19)(3.20), a personal best evaluation value is defined:

$$E_i^{Pbest(K)}(t) = \min_{k \in discovery_i(t)} \{|RSSI_i^k(t)| + c_4|D_i^K(t)|\}, \tag{3.23}$$

  where $RSSI_i^k(t)$ is the Received Signal Strength Indicator (RSSI), whose unit is dBm, of a beacon received by device i from event K at t and $discovery_i(t)$ is a set of events from which device i receives beacons at t. If a device cannot receive a beacon from any event, let the personal best evaluation value be a positive infinity.

- Local best evaluation value (E_i^{Lbest})

  Based on Eqs.(3.21) and (3.22), a local best evaluation value is defined:

$$E_i^{Lbest}(t) = \min_{j \in neighbor} \{E_j(t) + c_4 |N_i^j(t)|\}. \tag{3.24}$$

- Self-evaluation value (E_i)

  Based on Eq.(3.9), a self-evaluation value is defined:

$$E_i(t) = \begin{cases} E_i^{Pbest(K)}(t) \\ \quad if \;\; E_i^{Pbest(K)}(t) < \min_{j \in neighbor_i(t)} \{E_j^{Pbest(K)}(t)\} \\ E_i^{Lbest} + |RSSI_i^{Lbest}(t)| \\ \quad otherwise, \end{cases} \tag{3.25}$$

  where $RSSI_i^{Lbest}(t)$ is the RSSI of the information that device i received from a local best device at t.

Each device derives the above evaluation values based on beacons from the event and interactive information among neighbors, moves to the updated location with the evaluation values, and advertises its information with the updated location and evaluation values. Each device performs a sequence in a 1-sec period. If the distance between devices becomes lower than a threshold (D_c), it judges that a collision has occurred. Assuming that the devices failed due to a collision, the colliding devices stop advertising their information.

An event has an actuation capacity, which is the room necessary to complete an event's capture. The device in the actuating phase decreases 1 actuation capacity per one second. When the actuation capacity of an event becomes 0, it disappears from the simulation field.

Table 3.2: Comparison method of evaluation for MSC.

	Pbest	Lbest	Collision avoidance	Continuous search	Dynamic multi-swarming
Method 1	○	×	○	×	×
Method 2	○	×	×	○	×
Method 3	○	×	○	○	×
Method 4	○	○	○	○	×
Proposed method	○	○	○	○	○

3.4.2 Comparative evaluation

The proposed and comparative methods and their functions are shown in Table 3.2. The circles present a method with the shown mechanism in the item of Table 3.2, and the crosses present a method without it. Methods 1, 2, and 3 only search based on the personal best. Additionally, in method 1, devices have a collision avoidance control, in method 2, they have a continuous search control mechanism, and in method 3, they have above both control mechanisms. In method 4, the devices search for and actuate events by forming a single swarm based on personal and local bests. In the proposed method, devices form multiple swarms based on the dynamic multiple swarming mechanism.

The simulation results are investigated with comparative items as shown below:

- Capture ratios for events
- Turnaround times to capture all events
- Collision ratios
- Transition in capture ratios

The turnaround time to capture of all the events presents the time to capture all events. If the devices cannot complete searching for and actuating all the events, the turnaround time is given by the simulation time. The collision ratio is a ratio of the number of devices that collided. The transition in the capture ratio is the change in the capture ratio in time.

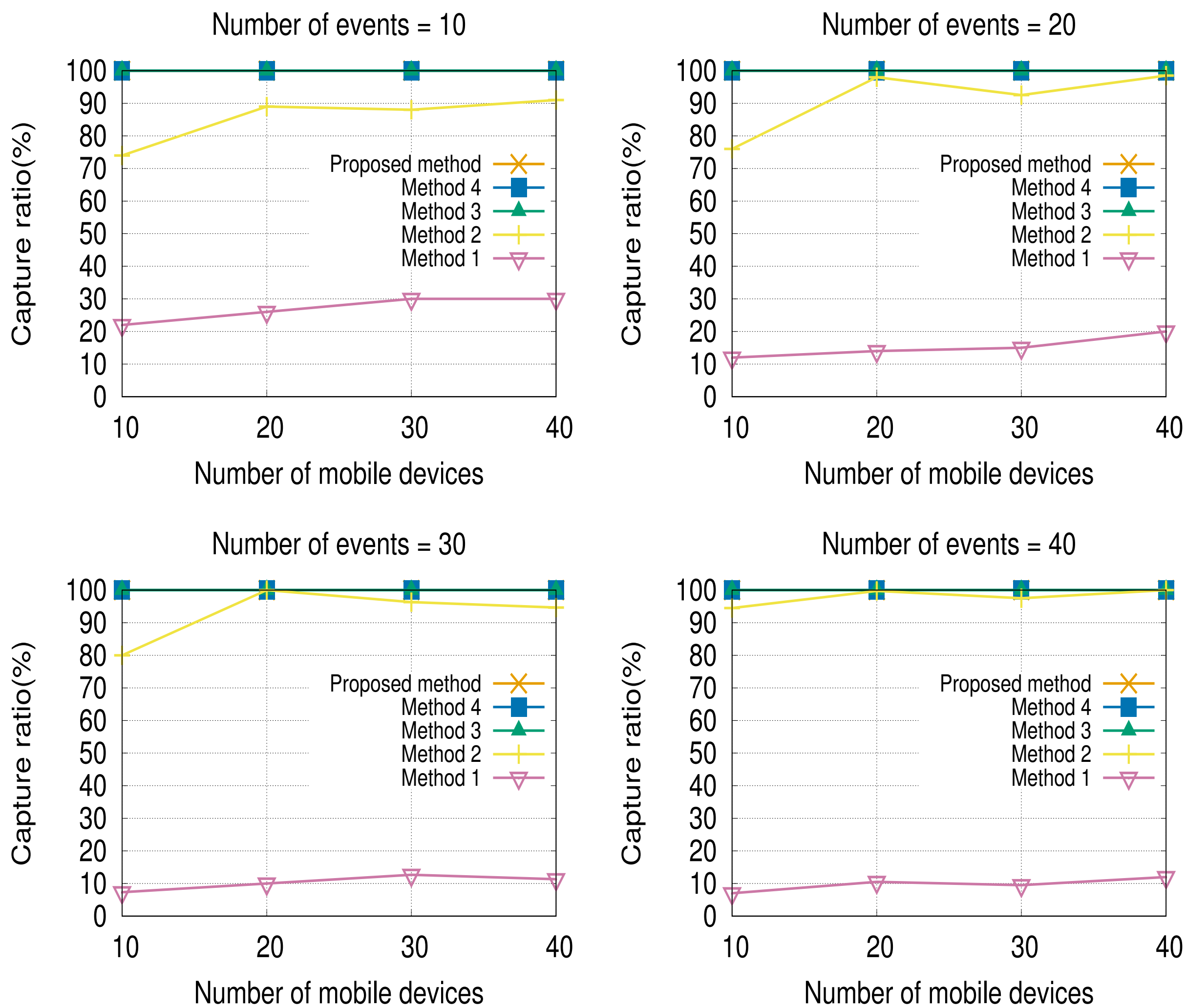


Figure 3.4: Capture ratio.

3.4.3 Simulation results

The simulation results, which are the total result of searching and actuating, the dependence of turnaround times on searching times and the dependence of turnaround times on actuating times, are shown below.

3.4.3.1 Total result of searching and actuating

The capture ratios are shown in Fig.3.4. Methods 3, 4, and the proposed method completely searched for and actuated all the events in all cases of the number of devices. Method 1 has no

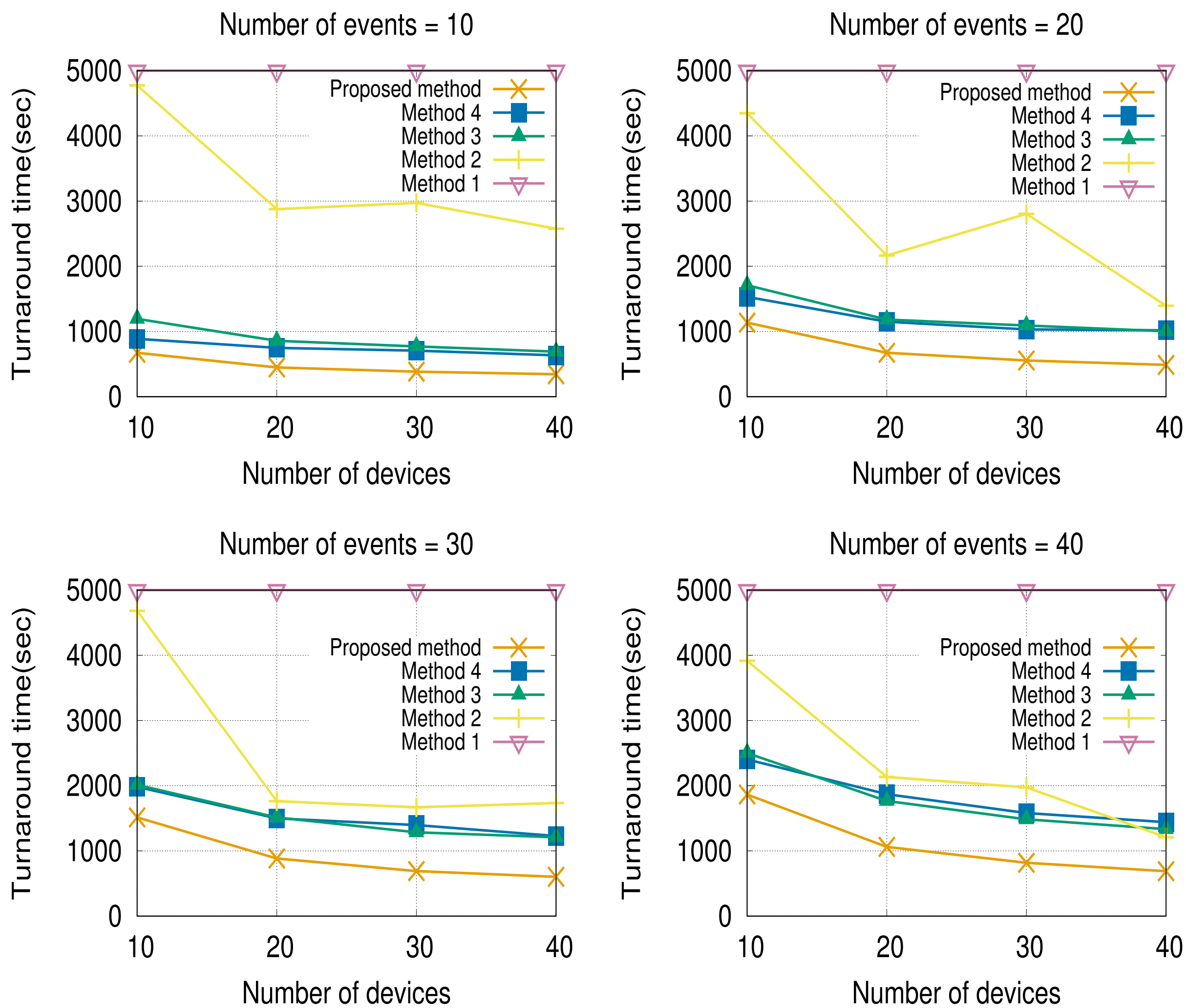


Figure 3.5: Turnaround times to capture all events.

continuous search mechanism, therefore, it failed to complete to the search, and only actuated 10～30% events. Its result is the worst among all the methods, because each device in it stops searching for and actuating after one event. Method 2 does not have a collision avoidance mechanism and does not complete its search for and its actuation of all the events in a small number of devices or a small number of events. It is supposed that the devices concentrate in a small number of events when the number of events is small, as the result, the number of active devices greatly decrease due to collision between them.

The turnaround times to capture of all the events are shown in Fig.3.5. Method 1 does not complete

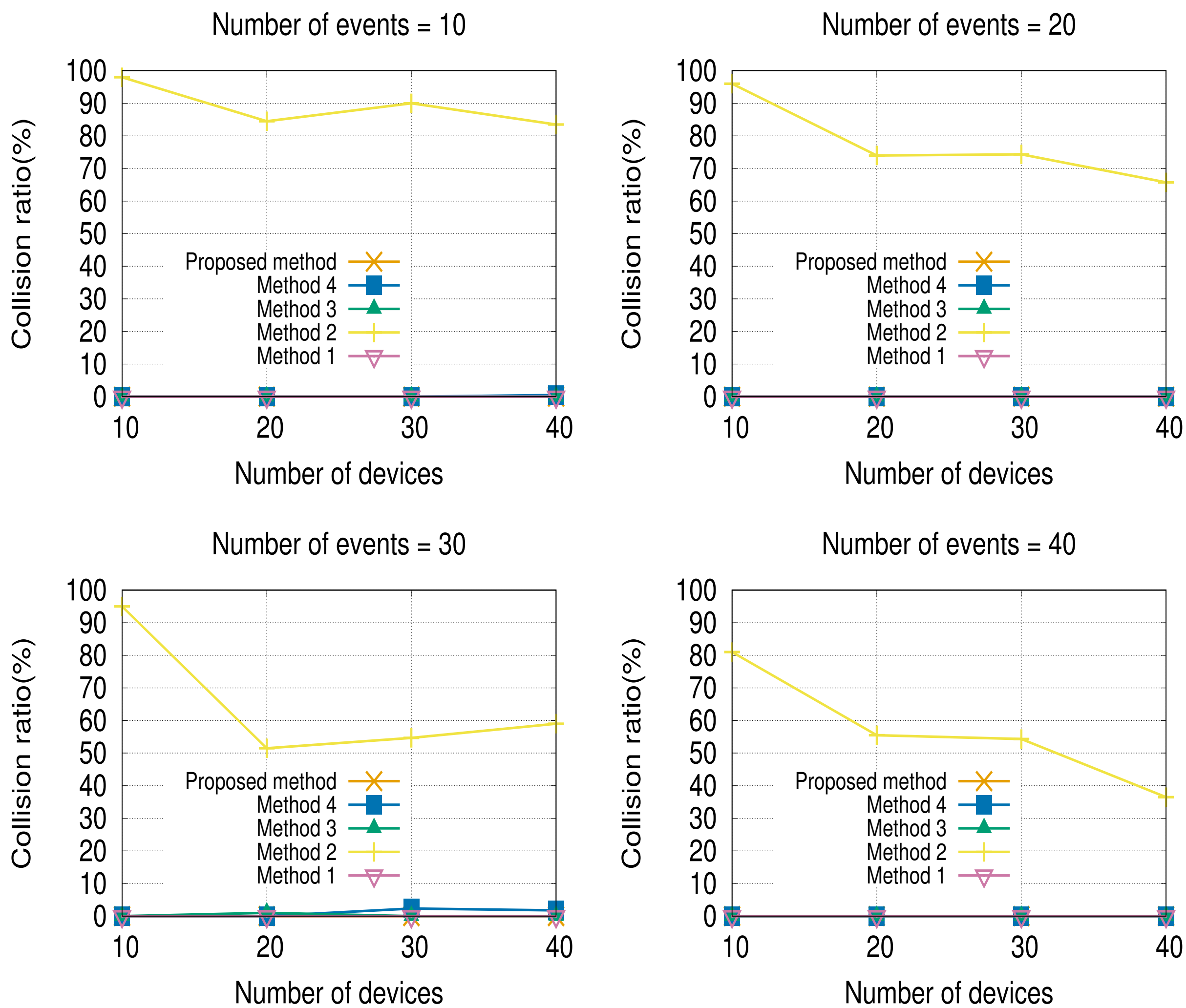


Figure 3.6: Collision ratio

its search for and its actuation of all the events. Method 2 requires a longer turnaround time in most cases than methods 3 and 4. The longer turnaround time in method 2 is caused by a decrease in the number of devices due to collisions. The turnaround time in our proposed method is the smallest in all the methods and the cases. Only the proposed method has a dynamic multiple swarming mechanism that can divide a swarm into multiple swarms and avoid swarms of unbalanced size. Therefore, the dynamic multiple swarming mechanism decreased the turnaround times for the proposed method and outperformed the other methods when searching for and actuating multiple events. Detailed evaluations and discussions are shown in sections 3.4.3.2 and 3.4.3.3.

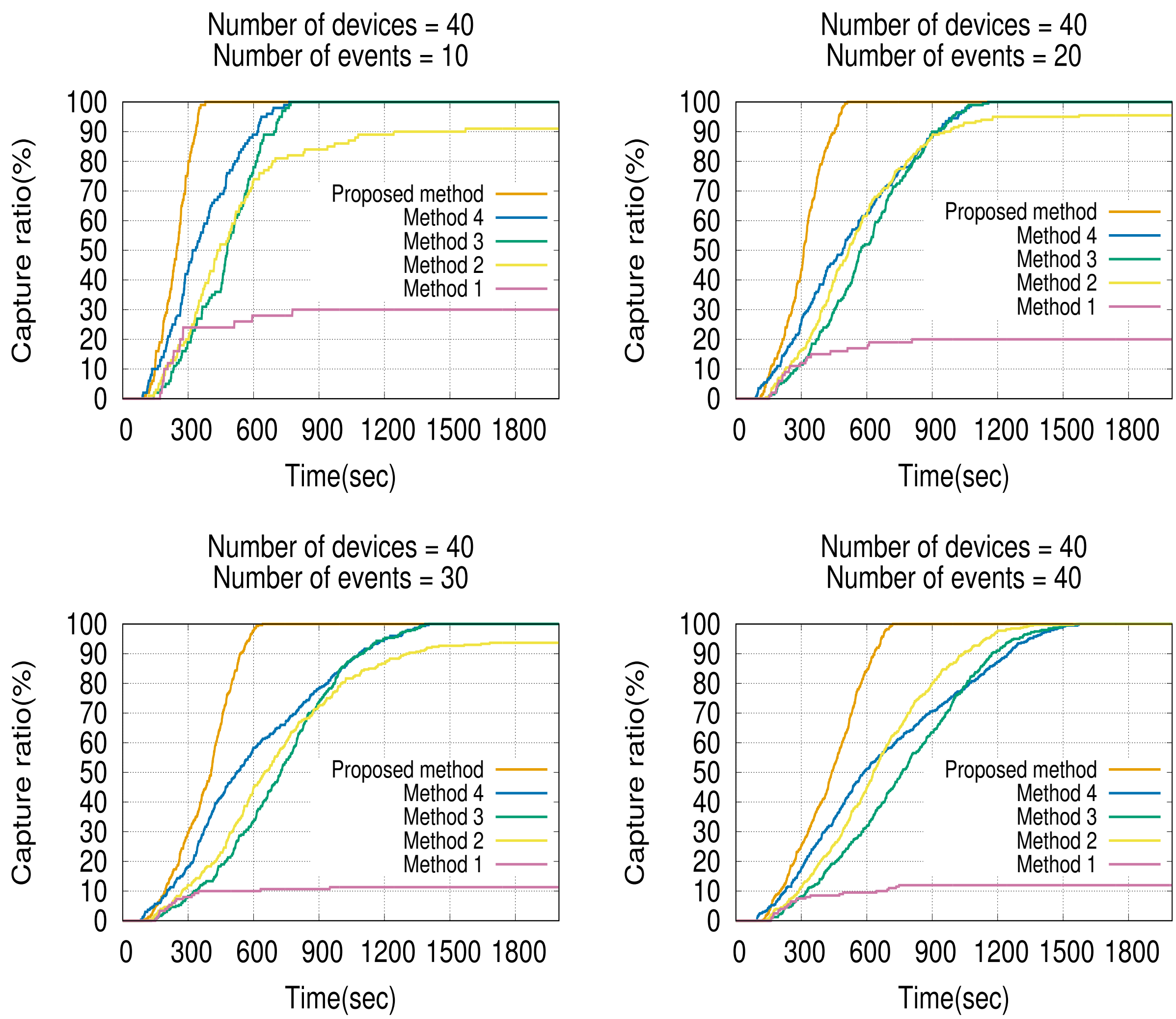


Figure 3.7: Transition in capture ratios with times for 40 nodes.

The collision ratio (Fig.3.6) in the methods with the collision avoidance mechanism is 0% in most cases. On the other hand, in method 2, which does not have the collision avoidance mechanism, the collision ratio increased as a natural result, and most devices collided with other devices. Therefore, the collision avoidance mechanism is effective.

The transition in the capture ratio for 40 devices is shown in Fig.3.7. The proposed method completes its search for and its actuation of all the events in the shortest time in all cases, and time difference increases with more events.

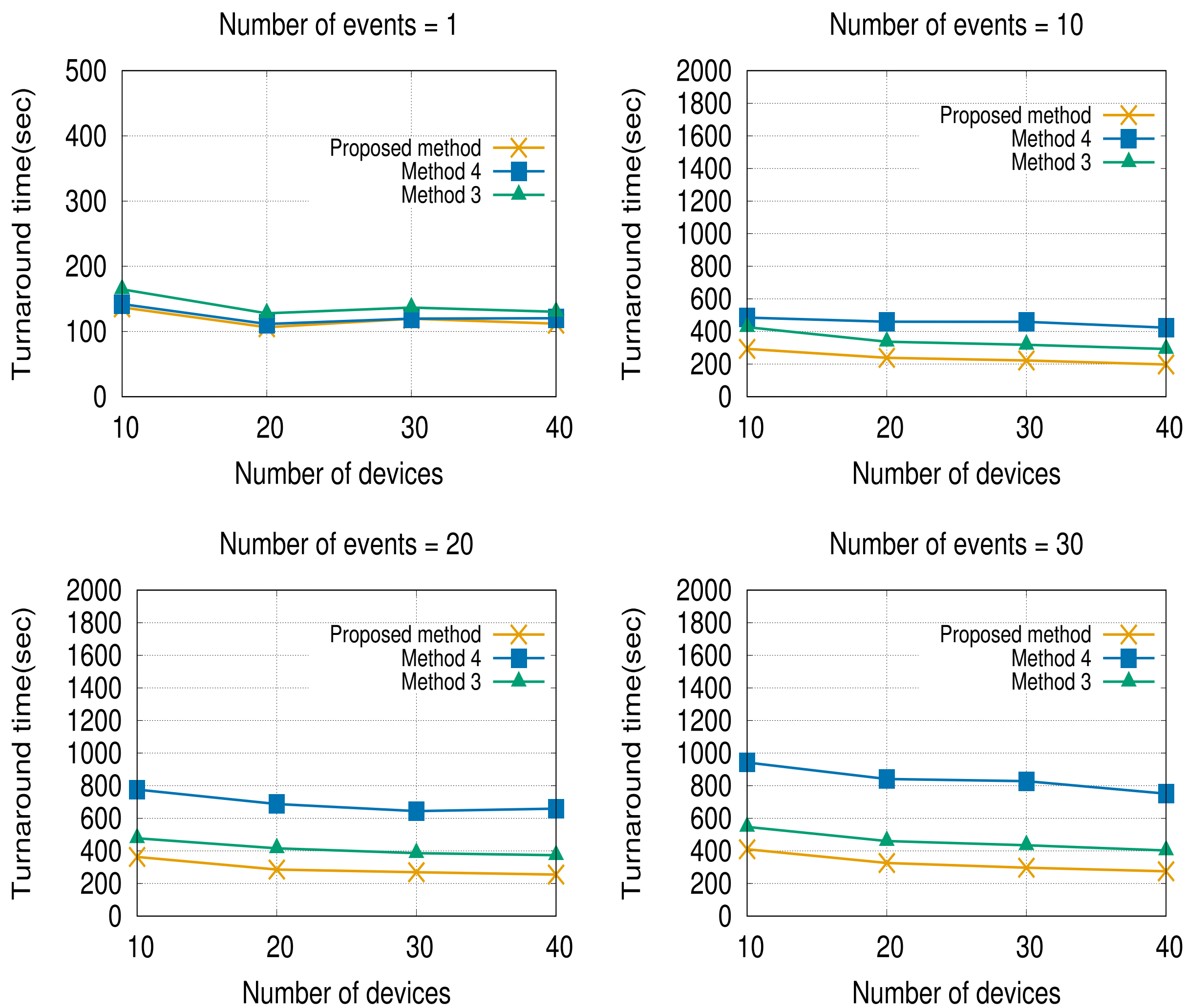


Figure 3.8: Turnaround times for capture capacity 1.

3.4.3.2 Dependence of turnaround times on searching times

Here, the dependence of the turnaround time on the searching times is shown. In order to consider a scenario that only consists of the searching time, let the actuation capacity of all the events be 1. In the parameters, methods 3, 4 and the proposed method are compared in term of the turnaround times and the transition in the capture ratios that changes in time. The number of events is 1, 10, 20, and 30. The other simulation parameters are identical as in Table 3.1.

The turnaround times are shown in Fig.3.8, and the transitions in the capture ratios are shown in Fig.3.9. When the number of events is 1, the turnaround times in method 4 and the proposed method

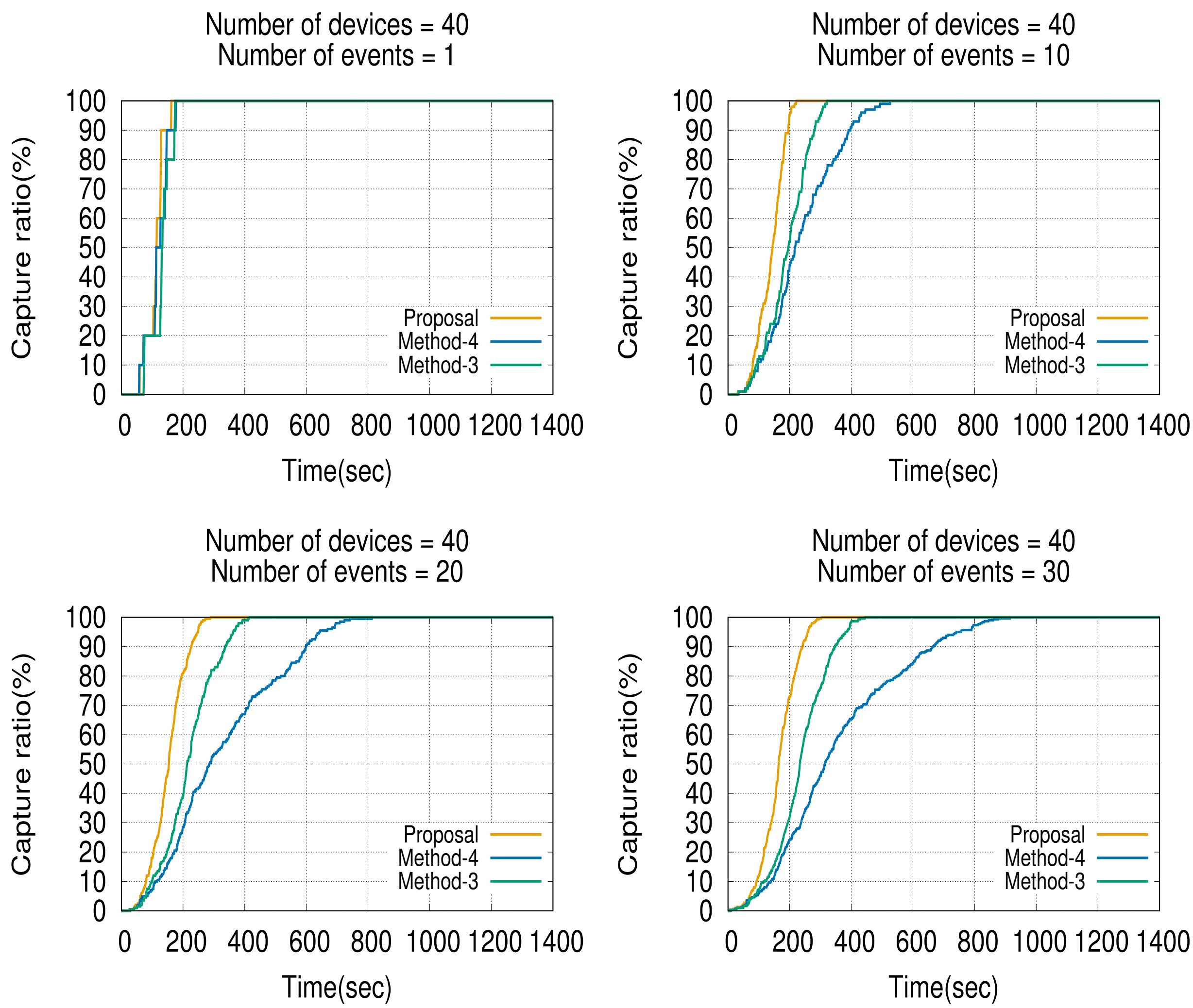


Figure 3.9: Transition in capture ratios with times for capture capacity 1

are smaller than those in method 3. As the number of events increases, the turnaround times in method 3 are smaller than those in method 4, and in the proposed method they are the smallest among all the methods, and the difference in the times between each method increases according as the number of events increases.

In method 3, which utilizes only the personal best, each device behaves as a leader and independently and selfishly searches for an event. On the other hand, in method 4 and the proposed method, which both utilize the personal and local bests, the devices form a swarm. When the number of events is 1, the search for an event requires one swarm. Therefore, when searching for one event, method

4 and proposed method, whose devices form swarms and cooperate with the others, are superior to method 3 where each device is independent and splits off on its own. As the number of events increases, the search for multiple events requires a parallel search for each one. In method 4, which has no dynamic multiple swarming mechanism, devices form only one swarm that sequentially searches for multiple events. On the other hand, in method 3, each device independently splits off, and in the proposed method with a dynamic multiple swarming mechanism, devices form multiple swarms, and both methods can search for multiple events in parallel. Therefore, for searching for multiple events, method 3 and the proposed method are superior to method 4. For searching for each event, a search based on a swarm is superior to the independent and splitting off searches. The proposed method is superior to method 3 because the proposed method searches cooperatively with other devices and has high parallelism to search by dividing a swarm into multiple swarms. In all the cases, the proposed method is superior to the others, and as the number of events increases, the time difference increases between the proposed method and the others. Therefore, in the search phase, the proposed method with the dynamic multiple swarming mechanism can appropriately divide a swarm into multiple swarms and balance the swarm sizes among multiple swarms.

3.4.3.3 Dependence of turnaround times on actuating times

Here, we show the dependence of the turnaround times on the actuating times. The number of devices and events is 30, and the actuation capacity is varied from 100 to 1000. In addition, we plot results with the actuation capacity of 1 in the same graph. The other simulation parameters are identical as Table 3.1.

The dependence of the turnaround times on the actuation capacity is small(Fig.3.10). When the actuation capacity is small, the turnaround time in method 3 is smaller than that in method 4. As the actuation capacity of the events increases, the value in method 4 is smaller than that in method 3. The turnaround time in the proposed method is always the smallest. The turnaround time of method 3 increases as the actuation capacity increases. The proposed method maintained its superior performance in most cases.

When the actuation capacity is small, the differences in the actuation time to an event is small. Therefore, in multiple events with small actuation capacity, actuation with splitting off is more effective than actuation with one aggregation, and method 3 is superior to method 4. As the actuation

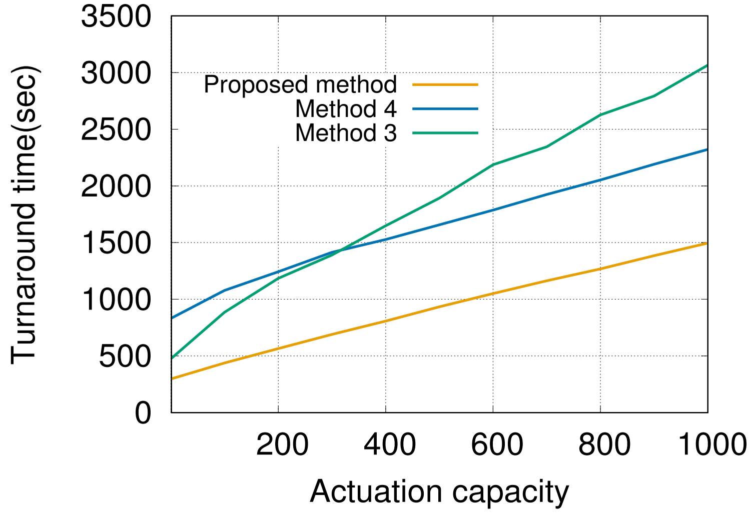


Figure 3.10: Dependence of turnaround times on capture capacity.

capacity of the events increases, actuation of an event requires more devices. Therefore, actuation with one aggregation is more effective than actuation with splitting off, and method 4 is superior to method 3. The turnaround time in the proposed method is always the smallest and superior to the others. In the actuation phase, the proposed method with the dynamic multiple swarming mechanism can also appropriately divide a swarm into multiple swarms and balance the swarm size among them.

3.5 Conclusions of MSC

In this chapter, assuming that the physical information from an event monotonically increases as approaching an event and that each event can be distinguished based on its physical information, we proposed an MSC with the following mechanisms based on wireless communications to reduce the time to search and actuate and the actuation of multiple events whose location and number in the occurrences are unknown:

- a mechanism that forms a swarm based on PSO;
- a mechanism that dynamically selects the nearest device to an event as a swarm leader;
- a mechanism that avoids collisions between devices;
- a mechanism that continuously searches for and actuates multiple events;

- a mechanism that dynamically divides a swarm into multiple swarms and balances the swarm size among multiple swarms.

MSC's effectiveness was shown by the following simulation results:

- MSC can search for and actuate multiple unknown events in a short time.
- In both searching and actuating performances, MSC is overwhelmingly superior to the independent parallel and collective methods.

Chapter 4

Mutation Mobile Sensing Cluster

In chapter 3, the MSC assumed that each event can be distinguished based on the physical information emitted by it. The physical information which corresponds to that assumption is radio waves for data communications. Since they are artificially structuralized for data communications and an event's identity is inserted into the structures, events can be distinguished by radio waves for data communications. However, in a real environment, most physical information emitted by a natural phenomenon is unstructuralized and includes no information that can distinguish events. Therefore, such physical information does not correspond to our assumption. In such indistinguishable physical information for events, when a device is in an area where physical information from multiple events overlap, it senses a strength that physical information from multiple events overlaps with each other. In the situation, MSC searches for events by a single swarm because a device senses a strength that physical information from multiple events overlaps with each other. Therefore, searching for and actuating multiple events in MSC becomes sequential and requires more time.

In this chapter, we propose a mutation MSC, which is an MSC with a mutation mechanism, to resolve the above issue caused by indistinguishable physical information for events. A mutation MSC is composed of the following three mechanisms:

- a search and actuation mechanism based on PSO
- a swarm local division mechanism
- a swarm global division mechanism

The search and actuation mechanism based on PSO is the mechanism proposed in chapter 3 . As mentioned above, in indistinguishable physical information for events, the dynamic multiple swarming mechanism in MSC does not work because it compares personal best evaluation values per event, and the swarm does not divide into multiple swarms for multiple events. Therefore, the above two swarm division mechanisms based on the mutation mechanism are introduced to MSC instead of the dynamic multiple swarming mechanism.

The swarm local division mechanism, when multiple events exist around a swarm, causes followers to mutate into leaders and a swarm to divide into multiple swarms in narrow areas. The swarm global division mechanism, when multiple events are widely scattered, the swarm global division mechanism causes normal device to mutate into heretic device which is newly introduced to the mutation MSC, and leave the swarm, which divides into multiple swarms and diffuse widely.

4.1 Assumption in Mutation MSC

The assumption for autonomous mobile devices is identical as that in MSC. In indistinguishable physical information for events, since a mutation MSC cannot assume that an event can be distinguished by its emitting physical information, it accordingly assumes that the device senses the strength that physical information from multiple events overlaps with each other.

4.2 Search and actuation mechanism based on PSO

The search and actuation mechanism based on PSO consists of the following mechanism:

- a mechanism that forms a swarm based on PSO;
- a mechanism that dynamically selects the nearest device to an event as a swarm leader;
- a mechanism that avoids collisions between devices;
- a mechanism that continuously searches for and actuates multiple events.

See chapter 3 that scrutinizes the above mechanisms.

4.3 Swarm local division mechanism

When multiple events are located around a swarm, the swarm local division mechanism divides it into multiple swarms to search for and actuate indistinguishable multiple events in parallel. The mechanism assumes that some followers in a swarm are located near events that are different from the approaching event of swarm, which are located in the back of the swarm. In the assumption, the followers cannot be aware of any nearby events due to indistinguishable events. Therefore, to divide a swarm, the mechanism causes the followers in the back of the swarm to behave as follows:

- The follower recognizes his location in the swarm.
- When the follower recognizes that he is at the back of the swarm, he mutates into a leader and leaves.

Location recognition in the swarm utilizes the neighbor-crowd degree, which accords with the number of devices between a device and its neighboring devices (Fig.3.3). The neighbor-crowd degree is formulated as follows. The neighbor-crowd degree and the updated rule of the local best evaluation value (including the neighbor-crowd degree) are shown below:

$$N_i^j(t) = \{x | x \in neighbor_i(t) \cap neighbor_j(t)\}, \tag{4.1}$$

where $N_i^j(t)$ is the neighbor-crowd degree of device i for neighbor device j at t.

If the neighbor-crowd degree is high, a device recognizes that it is located at the back of the swarm. Mutation into a leader is applied to the follower at back of the swarm who is expected to be near the event that is different from the swarm's approaching event. As mentioned above, the determination that the follower is located at the back of the swarm can be achieved by the neighbor-crowd degree. The determination that the follower is expected to be located near an event, which is different from the approaching swarm event, is achieved by comparing the personal and local best evaluation values. The personal best evaluation value corresponds to the distance from an event as a leader. On the other hand, the local best evaluation value corresponds to the indirect distance from the event based on a swarm as a follower. If the personal best evaluation value is smaller than the local best evaluation value, it remains unclear whether the event for each evaluation value is different. But the follower is

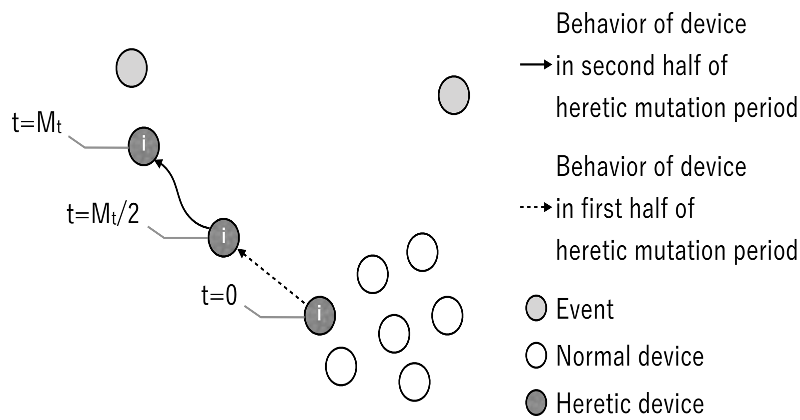


Figure 4.1: Behavior of a heretic device.

expected to be near the event and mutate into a leader. Then, the mutation causes the swarm to divide. To increase the mutation from the follower to the leader as a backward location in the swarm, the local best evaluation value is degraded by the neighbor-crowd degree (Eq.(4.2)). The above behavior is formulated as follows:

$$E_i^{Lbest}(t) = \min_{j \in neighbor_i(t)} \{E_j(t) + c_4 |N_i^j(t)|\}, \tag{4.2}$$

$$pb_i(t) = \begin{cases} 1 & if \;\; E_i^{Pbest}(t) < \min_{j \in neighbor_i(t)} \{E_j^{Pbest}(t)\} \\ & or \;\; E_i^{Pbest}(t) < E_i^{Lbest}(t) \\ 0 & otherwise. \end{cases} \tag{4.3}$$

$$lb_i(t) = \begin{cases} 0 & if \;\; E_i^{Pbest}(t) < \min_{j \in neighbor_i(t)} \{E_j^{Pbest}(t)\} \\ & or \;\; E_i^{Pbest}(t) < E_i^{Lbest}(t) \\ 1 & otherwise. \end{cases} \tag{4.4}$$

Based on Eqs.(4.3) and (4.4), even if the mutation increases from the follower to leader and the swarm excessively divides into multiple swarms, those that search for the same event will approach each other and eventually merge. Therefore, an appropriate number of swarms based on mutation can be maintained for multiple events.

4.4 Swarm global division mechanism

The swarm local division mechanism is effective when events are located around the swarm. However, it is unsuitable when multiple events are widely scattered. Therefore, we introduce the swarm global division mechanism in order for the swarm to be divided into multiple swarms that search for and actuate widely scattered multiple events. The mechanism spatially diffuses some devices, ignores the swarm's behavior, and increases opportunities to search for widely scattered multiple events in parallel.

Consequently, the mechanism introduces a device that ignores swarm behavior, repulses its neighbors, and goes far from the swarm. This is called a heretic device, in contrast to a normal device that is a leader or a follower. A normal device, which mutates into a heretic device based on heretic mutation probability M_p, behaves as a heretic device during heretic mutation period M_t. In the heretic device, the mechanism works to realize emergence of behavior that ignores the swarm behavior, impulses to the neighbors, and goes far from the swarm:

- To ignore the swarm behavior, a device does not search for events based on Eqs.(4.3) and (4.4); it only searches for events based on its personal best.
- To repulse its neighbors and go far from the swarm, the device increases its collision avoidance vectors.

In the first half of the heretic mutation period, the heretic device increases the collision avoidance vectors to repulse its neighbors, leaves the swarm, and searches for events based on its personal best to ignore swarm behavior. The above behavior is shown as a dotted arrow in Fig.4.1. In the second half of the heretic mutation period, the heretic device restores a collision avoidance vector to the state before the mutation, assuming that it sufficiently leaves the swarm, and searches for events based on its personal best (solid arrow in Fig.4.1).

If the heretic mutation period expires, the heretic device mutates into a normal device and behaves as a leader or a follower based on Eqs.(4.3) and (4.4). After the heretic mutation period, the device, which is mutating from a heretic into a normal device, is located far away from the swarm, and events around the device are likely to be different from the approaching events with the swarm. Therefore, the local best in the device is degraded because the device moves far away from the swarm and is likely to be a leader based on Eq.(4.3). Furthermore, the event that the device approaches as a leader

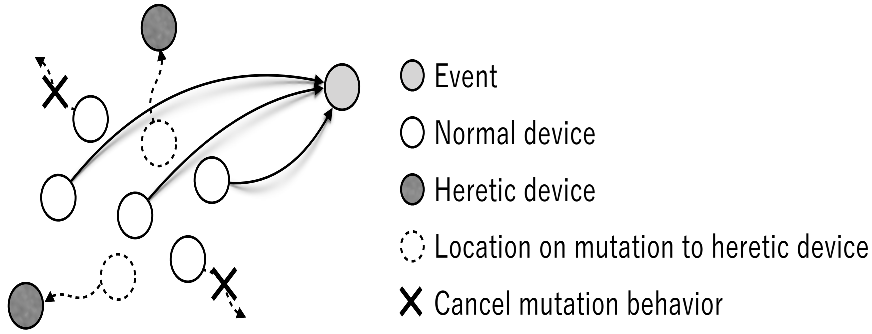


Figure 4.2: Restriction on number of heretic devices.

is probably different from the event toward which the swarm is approaching. We expect the device that is mutating from a heretic to a normal device to become a leader and form another swarm.

Each device independently determines its mutation to a heretic device based on the heretic mutation probability. Consequently, the number of heretic devices might temporarily and accidentally increase. In such cases, many devices leave the swarm and are dispersed randomly, causing the search performance to deteriorate. Therefore, the mechanism sets an upper limit for the number of heretic devices to prevent a decrease in the number of devices that comprise a swarm and increase in the number of heretic devices. The restriction on mutation into a heretic device is executed in each device:

- The device adds the information whether it is a normal or a heretic device to the interaction information, which includes its location, its personal best evaluation value, and its self-evaluation value, and distributes it to its neighbors.
- The device derives the number of heretic devices among its neighbors based on receiving interaction information from its neighbors and maintains that number.
- The device determines the mutation into heretic devices by the heretic mutation probability and a comparison between its maintained number of heretic devices and their upper limit.

As shown in Fig.4.2, the devices mutating into heretic devices are restricted to particular devices by the above procedure in each device.

We formulated the increase in the collision avoidance vector to repulse the neighbors and leave the swarm by collision avoidance weight c_3:

$$c_{3i}(t) = \begin{cases} c_3^{Heretic} & if \;\; P(M_p|H_i(t) \leqq M_u) \\ c_3^{Search} & if \;\; x_i(t) - x_i(t - A_t) > A_d \\ c_3^{Capture} & otherwise, \end{cases} \tag{4.5}$$

where $P(\cdot)$ is a probability, $H_i(t)$ is the number of heretic devices in the neighbor of device i at time t, and M_u is the upper limit of the heretic devices.

We redefine Eqs.(4.3) and (4.4) to realize emergence of the mutation behavior of the swarm local division mechanism and the swarm global division mechanism in a device:

$$pb_i(t) = \begin{cases} 1 & if \;\; P(M_p|H_i(t) \leqq M_u) \\ & or \;\; E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest}(t)\} \\ & or \;\; E_i^{Pbest}(t) < E_i^{Lbest}(t) \\ 0 & otherwise. \end{cases} \tag{4.6}$$

$$lb_i(t) = \begin{cases} 0 & if \;\; P(M_p|H_i(t) \leqq M_u) \\ & or \;\; E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest}(t)\} \\ & or \;\; E_i^{Pbest}(t) < E_i^{Lbest}(t) \\ 1 & otherwise. \end{cases} \tag{4.7}$$

4.5 Simulation evaluation

This simulation validates the effectiveness of mutation MSC.

Table 4.1: Simulation parameters of evaluation for mutation MSC.

Parameters	Values
Simulator	ns3
Simulation time (sec)	5000
Number of trials for each simulation scenario	10
Number of devices	10～30

Table 4.1: (cont'd) Simulation parameters of evaluation for mutation MSC.

Parameters	Values
Number of events	10～30
Initial location of devices ($m \times m$)	(30,30), Uniform distribution
Initial location of events ($m \times m$)	(100,100), Uniform distribution
Update cycle of velocity vector (sec)	0.1
Actuation capacity of an event	300
T (dBm) in Eq.3.15	-50.6262
A_t (sec)	10
A_d (m)	1
w	0.5
pb	1
lb	1
c_3^{Search}	50
$c_3^{Capture}$	100
M_{Search}^{upper} (m/sec)	1
$M_{Capture}^{upper}$ (m/sec)	0.3
k in Eq.(3.12)	2
$\theta(^\circ)$ in Eq.(3.5)	30
Evaluation discard time of continuous control (sec)	1
Coefficient of neighbor-crowd degree c_4	-10
M_p	1/1000
M_t(sec)	200
$c_3^{Heretic}$	100000
Wireless communications	IEEE802.11b
Transmission power (dBm)	17.0206
Path loss (dB)	$L_0 + 10n\log_{10}(\frac{d}{d_0})$
Path loss at reference distance L_0 (dB)	-46.6777
Reference distance d_0(m)	1
Propagation loss factor n	3

Table 4.1: (cont'd) Simulation parameters of evaluation for mutation MSC.

Parameters	Values
Radio wave reach distance (m)	250
Effective actuation radius (m)	5
D_c(m)	1

4.5.1 Simulation specifications

The simulation parameters are listed in Table 4.1 and the devices and events are defined:

- a device is equipped with an IEEE802.11b interface and periodically advertises its information (section 3.2.8).
- an event emits indistinguishable physical information for an event.

Each device receives interaction information from its neighbors. The RSSI of the information from neighboring devices and the beacons from events are treated as a dBm value. Each device also derives the following three evaluation values:

- personal best evaluation value (E_i^{Pbest})

$$E_i^{Pbest}(t) = E_i^{dir}(t) \tag{4.8}$$

- local best evaluation value (E_i^{Lbest})

$$E_i^{Lbest}(t) = \min_{j \in neighbor_i(t)} \{E_j(t) + c_4 |N_i^j(t)|\} \tag{4.9}$$

- self-evaluation value (E_i)

$$E_i(t) = \begin{cases} E_i^{Pbest}(t) \\ \quad if\ \ E_i^{Pbest}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest}(t)\} \\ \quad or\ \ E_i^{Pbest}(t) < E_i^{Lbest}(t) \\ E_i^{Lbest}(t) + |RSSI_i^{Lbest}(t)| \\ \quad otherwise \end{cases} \tag{4.10}$$

Here $RSSI_i^{Lbest}(t)$ is the RSSI of the information received by device i from a device treated as a local best device at t.

Each device derives the above evaluation values based on beacons from the event and interactive information between neighbors, moves to the updated location with the evaluation values, and advertises its information with the updated location and evaluation values. Each device performs the sequence in 1-sec periods. If the distance between devices becomes lower than a threshold (D_c), it judges that a collision has happened. Assuming that the devices fail due to a collision, the colliding devices stop advertising their information.

An event has an actuation capacity, which is the room necessary to complete an event's capture. The device in the actuating phase decreases 1 actuation capacity per sec. When the actuation capacity of an event becomes 0, it disappears from the simulation field.

Assuming that a device cannot distinguish events by physical information, it can only sense the strength of overlapped physical information from multiple events. Therefore, such strength by distance is shown as follows [57]:

$$E_i^{dir}(t) = \log \frac{1}{2} \prod_{j=1}^{n} d_{ij}^2(t). \tag{4.11}$$

Eq.(4.11) monotonically increases as the product of the distance to the power of 2 to each event. On the other hand, the strength of the physical information monotonically decreases as the n-th power of the distance to the events. Therefore, Eq.(4.11) monotonically decreases as the product of the strength of the overlapped physical information from multiple events and monotonically decreases as the sum of the strength of the physical information from multiple events. That is, the personal best evaluation value monotonically decreases as the strength of the overlapped physical information, which is sensed by a device, from multiple events based on Eq.(4.11).

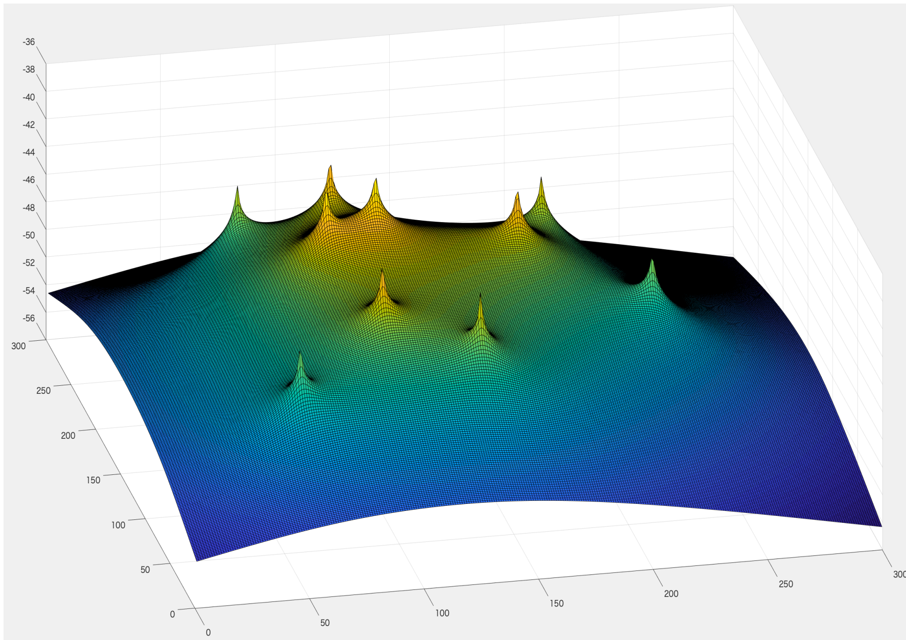


Figure 4.3: Example of distribution of physical information emitted by events.

The distribution of the strength of the physical information based on Eq.(4.11) is shown in Fig.4.3. The strength monotonically increases based on its approach to an event.

4.5.2 Comparative evaluation

The proposed and comparative methods and their functions are shown in Table 4.2. The circles present that a method has the mechanism in item of Table 4.2, and the crosses present that it does not. In method 1, the devices independently and scatteredly search for and actuate events only based on their personal best. In method 2, devices search for and actuate events based on the MSC described in chapter 3. In the proposed method, devices search for and actuate events based on mutation MSC that introduces swarm local and global division mechanisms.

The comparative item is the turnaround time to search and actuate. If the devices cannot complete searching for and actuating all events, the turnaround time is given by the simulation time.

Table 4.2: Comparison methods of evaluation for mutation MSC.

	Pbest	Lbest	Swarm local division	Swarm global division
Method 1	○	×	×	×
Method 2	○	○	×	×
Proposed method	○	○	○	○

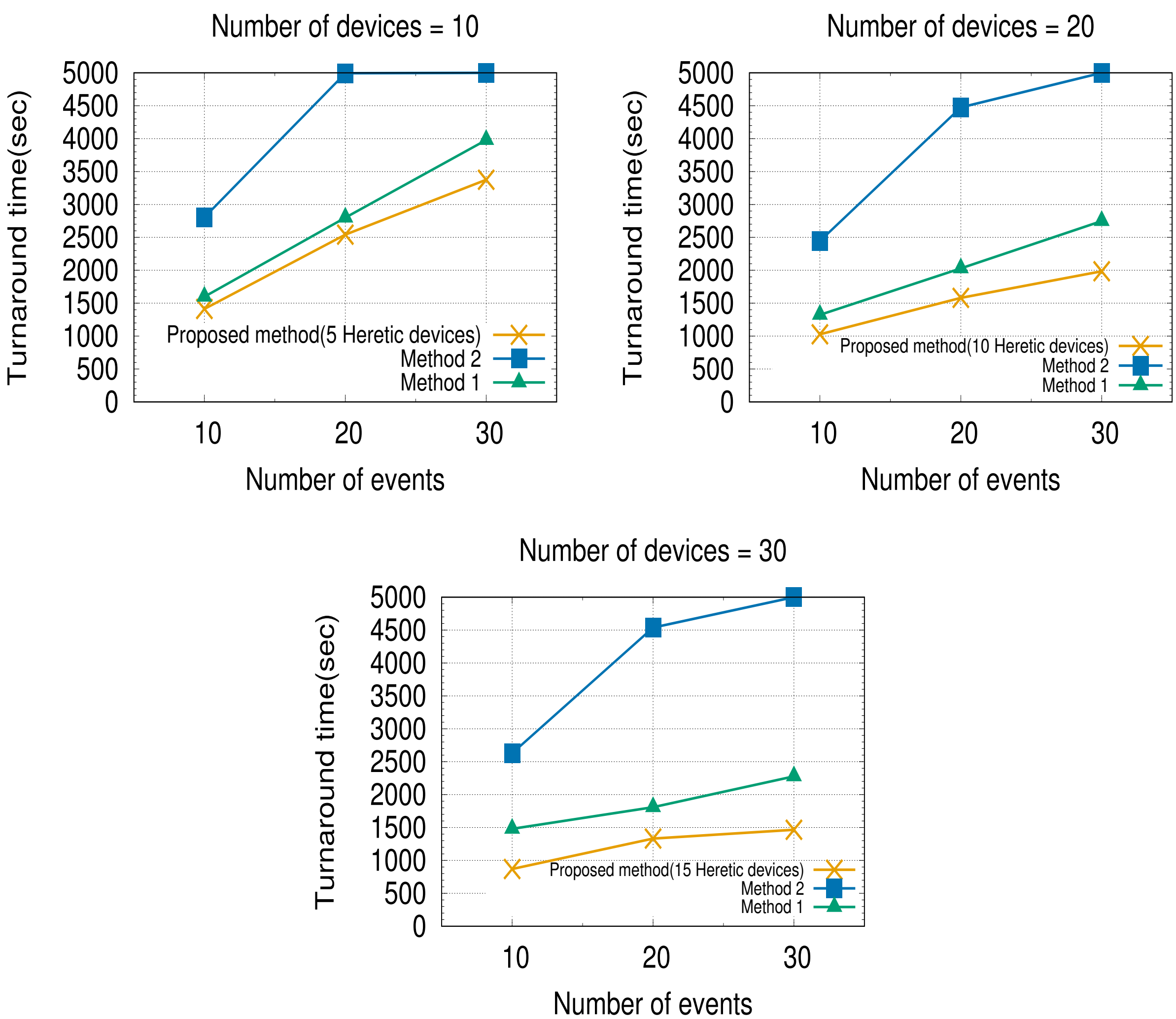


Figure 4.4: Turnaround time.

4.5.3 Simulation results

The simulation results, which are the total search and actuation results, the dependence of turnaround times on M_u and M_p and the dependence of turnaround times on field size, are shown below.

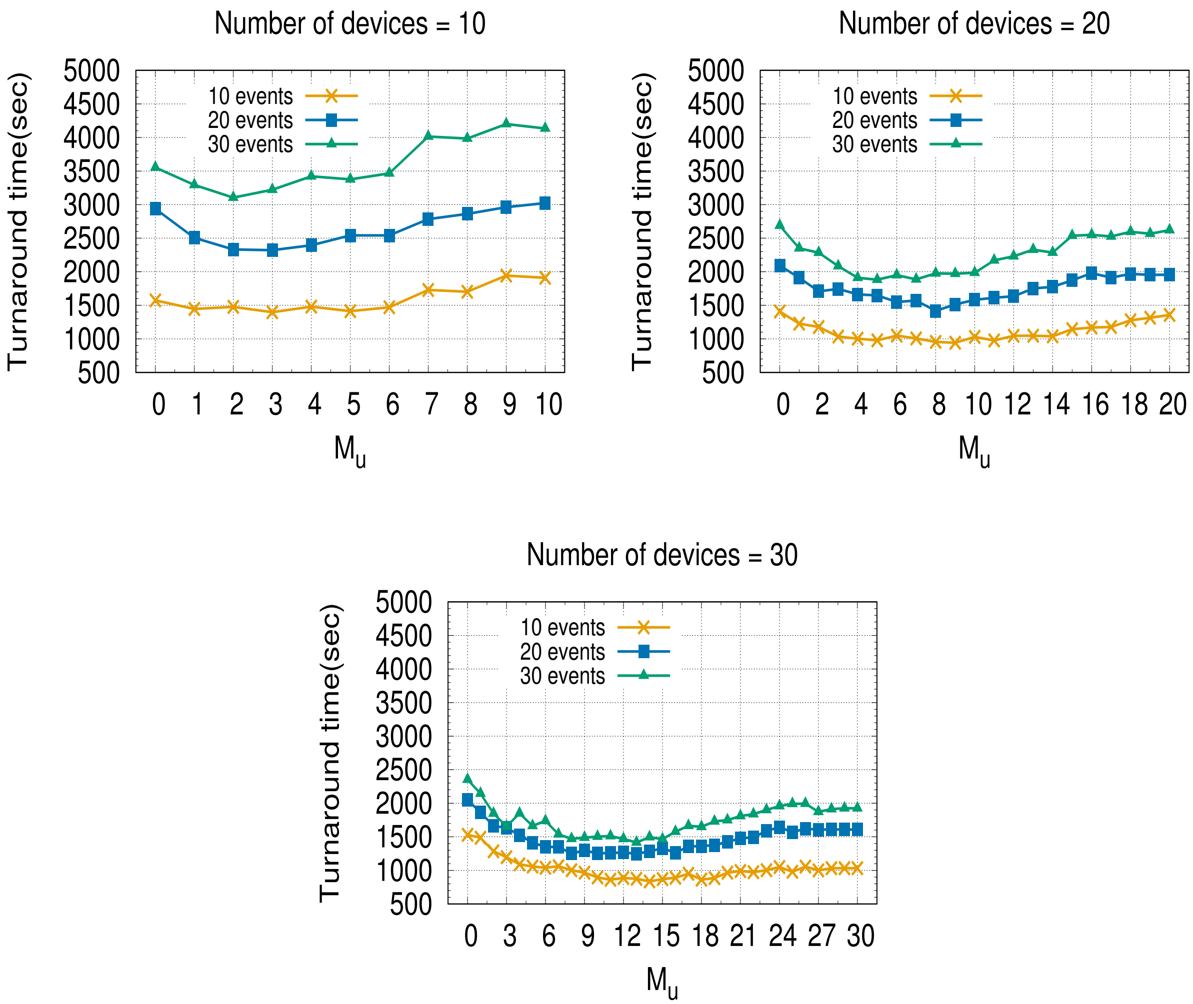


Figure 4.5: Dependence of turnaround times on M_u.

4.5.3.1 Total search and actuation results

The results are shown in Fig.4.4. In all the methods, the turnaround time decreased as the number of devices increased and rose as the number of events increased. Method 2 requires more time to search for and actuate all the events than the others. Method 1 decreases the turnaround times more than method 2. The proposed method decreased the turnaround times more than others, and the difference of the turnaround times between the proposed method and the others increased as the number of devices and events increased.

In method 2, the devices form a single swarm and search for and actuate events. Since it does the

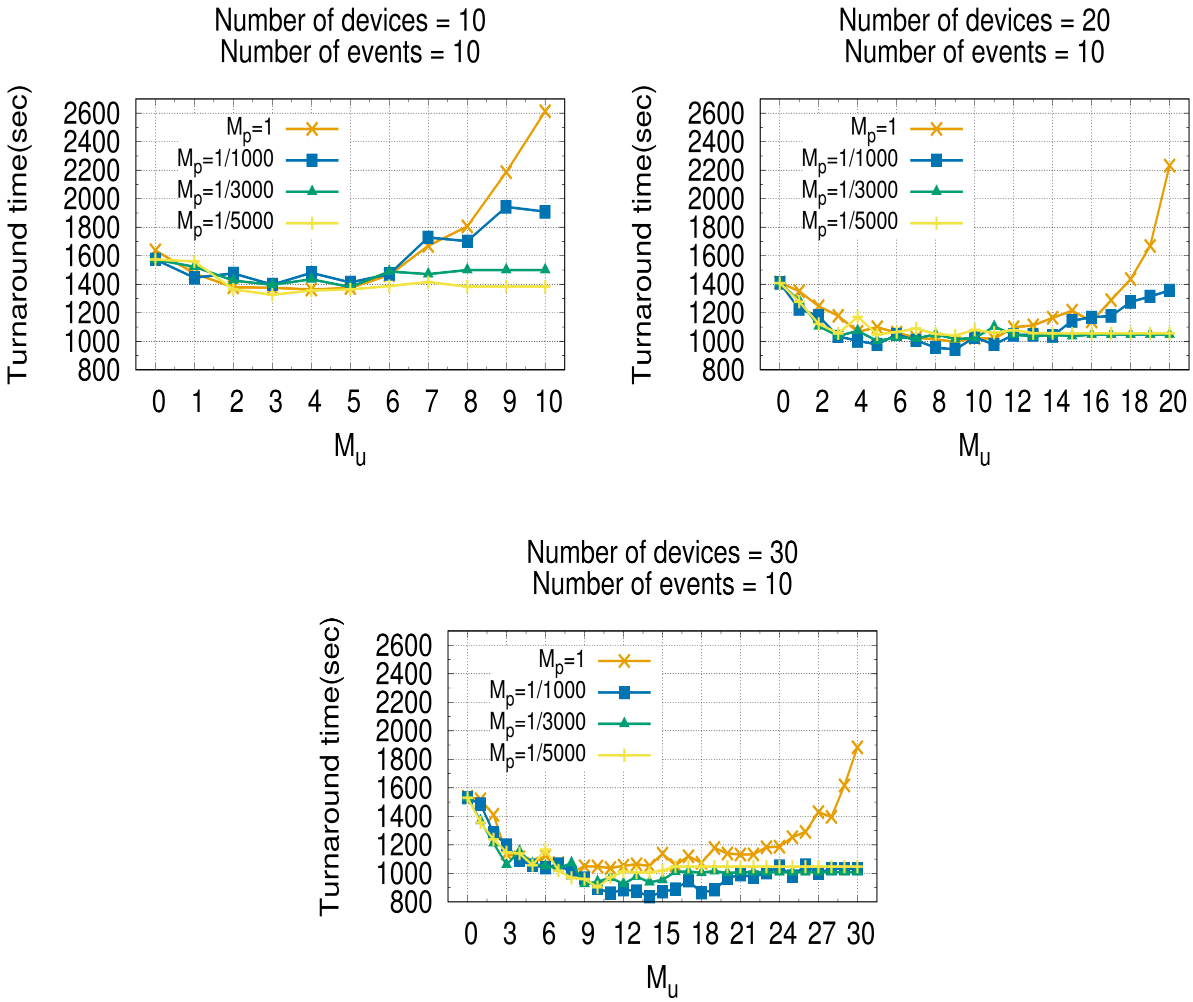


Figure 4.6: Dependence of turnaround times on M_p.

search and actuation of multiple events one by one, the turnaround time largely increases. In method 1, the devices independently and scatteredly search for and actuate events; that is, they search for and actuate them in parallel. But because they approach their nearest neighbor event, the parallel operation is restricted to a local area in their neighborhood. On the other hand, in the proposed method, the devices form local multiple swarms in local areas by the swarm local division mechanism, and some devices are scattered widely and form swarms with the swarm global division mechanism; devices form multiple swarms in wide areas. Consequently, the turnaround time in the proposed method decreases more than those in methods 1 and 2.

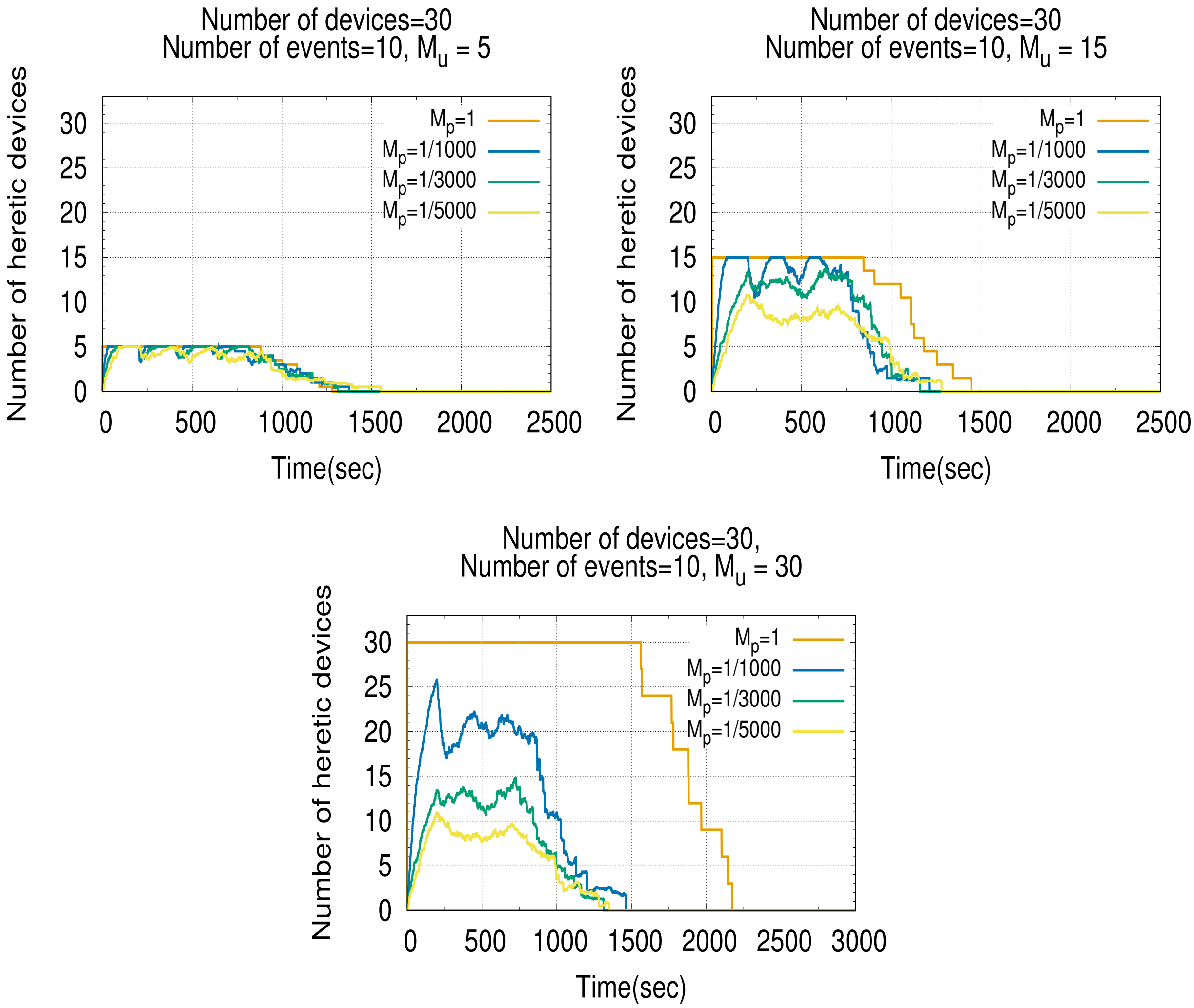


Figure 4.7: Dependence of number of heretic devices in time on M_u and M_p.

4.5.3.2 Dependence of turnaround times on M_u and M_p

The dependence of the turnaround times on the upper limit for heretic devices M_u is shown in Fig.4.5. The turnaround time depends on M_u, and the dependency is generally convex downward with M_u.

The dependence on M_u for the four cases of heretic mutation probability M_p are shown in Fig.4.6. The dependence of turnaround time on M_p is generally convex downward regardless of the number of devices or M_p, like in Fig.4.5. Therefore, M_u is a dominant parameter that minimizes the dependence of the turnaround time. Because the devices randomly and independently mutate to the heretic devices

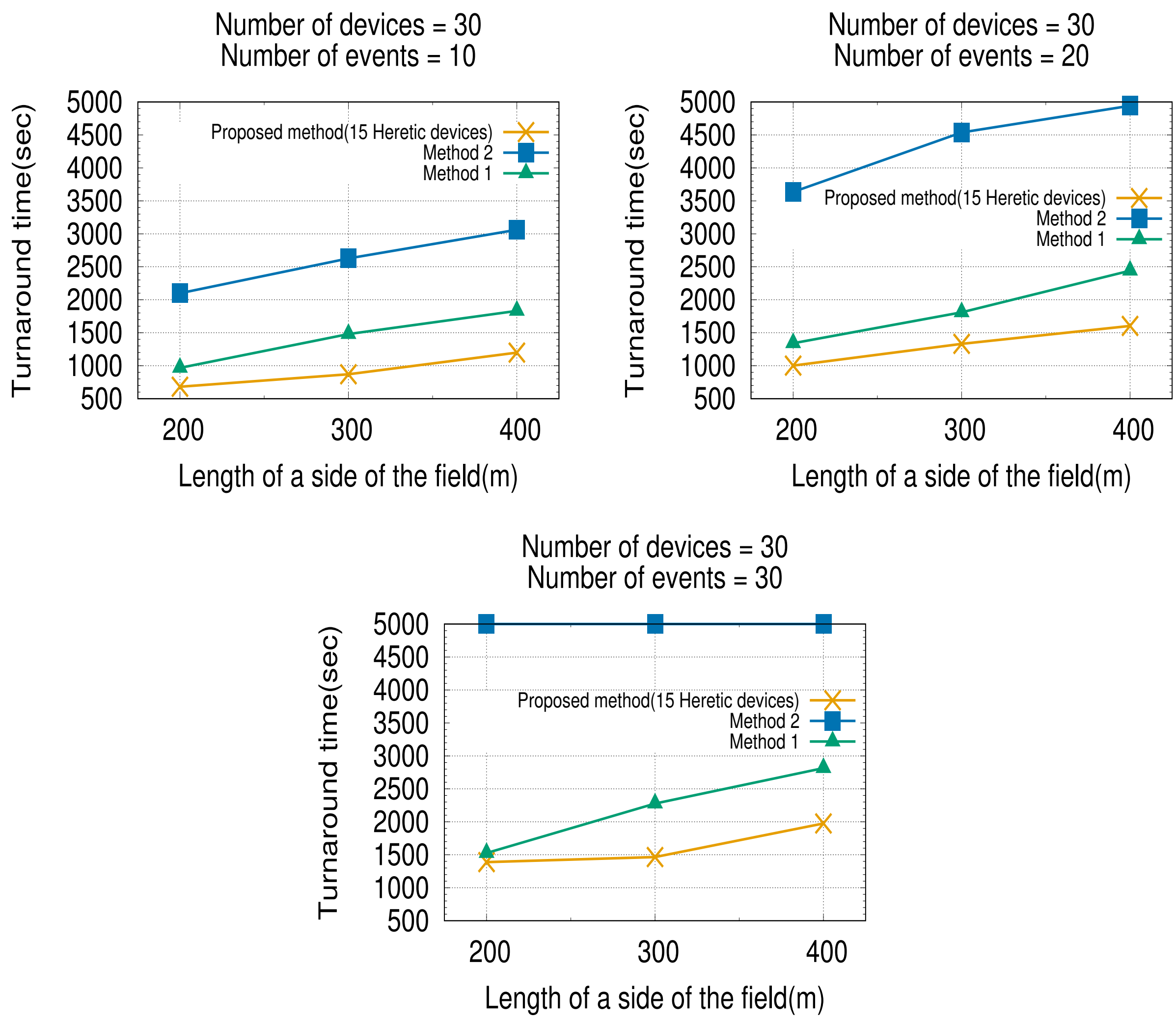


Figure 4.8: Dependence of turnaround times on field size.

based on M_p, the number of heretic devices might increase temporarily and accidentally. Fig.4.7 shows the transition in the number of heretic devices based on M_p. M_u prevents a rapid increase in the number of heretic devices by M_p. Therefore, based on Figs. 4.6 and 4.7, M_u prevents too many devices from leaving the swarms and being scattered, and our proposed method reduces the temporary degradation in the search performances. M_u decreases the turnaround times and controls the search and actuation performances in the proposed method.

4.5.3.3 Dependence of turnaround times on field size

The dependence of the turnaround times in the proposed and comparative methods on the field size is shown in Fig.4.8. The turnaround time of each method increases as the field size increases. The difference of the turnaround times between the proposed method and the others increases as the field size increases. In the proposed method, devices mutate from normal to heretic and strongly repulse their neighbors, go far from the swarm, and form swarms to events. The formed multiple swarms disperse widely and rapidly to the field, and the proposed method effectively works for widely and scattered events.

4.6 Conclusions of mutation MSC

In real environments, most physical information emitted by a natural phenomenon is unstructuralized without any information that can distinguish events. Therefore, events are individually indistinguishable based on physical information. In chapter 4, assuming that events are individually indistinguishable by the physical information emitted by them, we proposed a mutation MSC that searches for and actuates a larger number of such events within a limited time. The mutation MSC introduces the following two mechanisms to search for and actuate indistinguishable multiple events in parallel by forming multiple swarms.

- Swarm local division mechanism
- Swarm global division mechanism

The swarm local division mechanism divides a swarm into multiple swarms by mutating nodes from a follower to a leader based on the swarm's density and searches for and actuates indistinguishable multiple events in local areas. The swarm global division mechanism spreads devices by mutating nodes from normal to heretic devices, forms swarms in wide areas, and searches for and actuates indistinguishable multiple events in wide areas.

In simulation evaluations, the mutation MCS's effectiveness was shown:

- the mutation MSC decreased the turnaround times better than the MSC and a random search.
- the difference of the turnaround times increased as the field size increases.

- the performance of the search and actuation in the mutation MSC, which is dependent on the upper limit for the number of heretic devices, can be controlled by using it.

Chapter 5

Dynamic swarm-scale MSC

MSC assumes that the strength of the physical information sensed by a device monotonically increases as a device approaches an event which is a source of physical information. However, in a real environment, physical information includes random errors caused by an obstacle or interference and its strength does not necessarily monotonically increase as a device approaches an event. In such environments, the swarms in an MSC is required long time to search for and actuate multiple unknown events for long period of time because they are moving in the incorrect direction to an event.

In this chapter, a dynamic swarm-scale MSC is proposed to search for and actuate multiple unknown events in noisy environments where the strength of the physical information includes random errors. The dynamic swarm-scale MSC is an MSC with a dynamic scale mechanism. This dynamic swarm-scale mechanism adapts the spatial scale in the swarm to the strength of the physical information emitted by events and provides a swarm with tolerance to random error.

This chapter first outlines the basic MSC model and then describes a swarm's behavior in noisy environments and a dynamic swarm-scale MSC. Finally, the effectiveness of the dynamic swarm-scale MSC is shown with simulation evaluations.

5.1 Basic MSC model

In this section, the basic MSC model is described.

5.1.1 Search and actuation mechanism based on PSO

The search and actuation mechanism based on PSO consists of the following mechanisms:

- a mechanism that forms a swarm based on PSO.
- a mechanism that dynamically selects the nearest device to an event as a swarm's leader.
- a mechanism that avoids collisions between devices.
- a mechanism that continuously searches for and actuates multiple events.

See chapter 3 for the details of the above mechanism.

5.1.2 Dynamic multiple-swarming mechanism

The dynamic multiple-swarming mechanism consists of the following mechanisms:

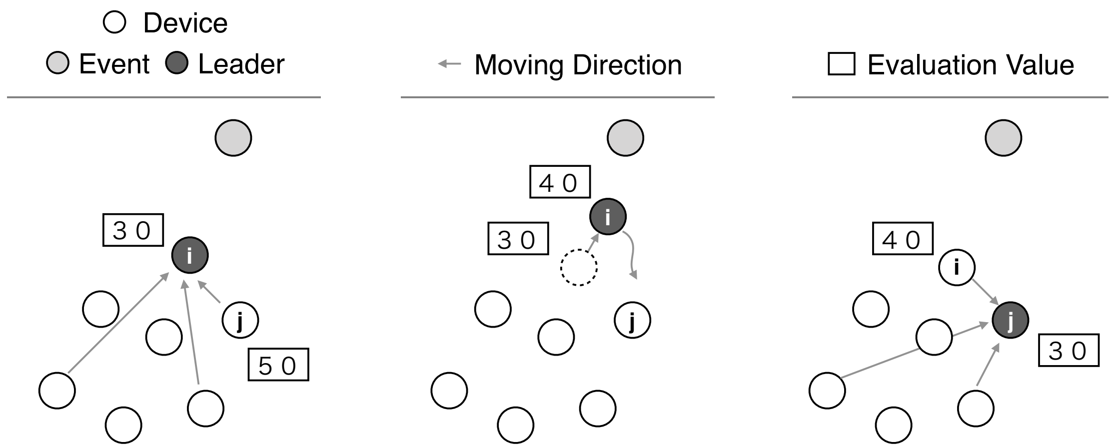
- a mechanism that divides a swarm into multiple swarms based on multiple events with the event-crowd degree.
- a mechanism that makes the number of followers uniform among multiple swarms with the neighbor-crowd degree.

See chapter 3 for details of the above mechanisms.

5.2 MSC swarm behavior in noisy environments

In an environment whose physical information does not include random errors, the strength of the physical information emitted by events monotonically increases when approaching the event. The increase in the strength of the physical information sensed by the device corresponds to a decrease in the distance between the device and the event. In MSC, the device nearest the event (Fig.5.1a) is selected as a leader, which the other devices follow.

A noisy environment, where the physical information is sensed by a device, includes random errors, and two incorrect behaviors emerge in an MSC swarm. One is that the leader moves in an incorrect direction to an event: in a direction where the personal best evaluation value decreases, that



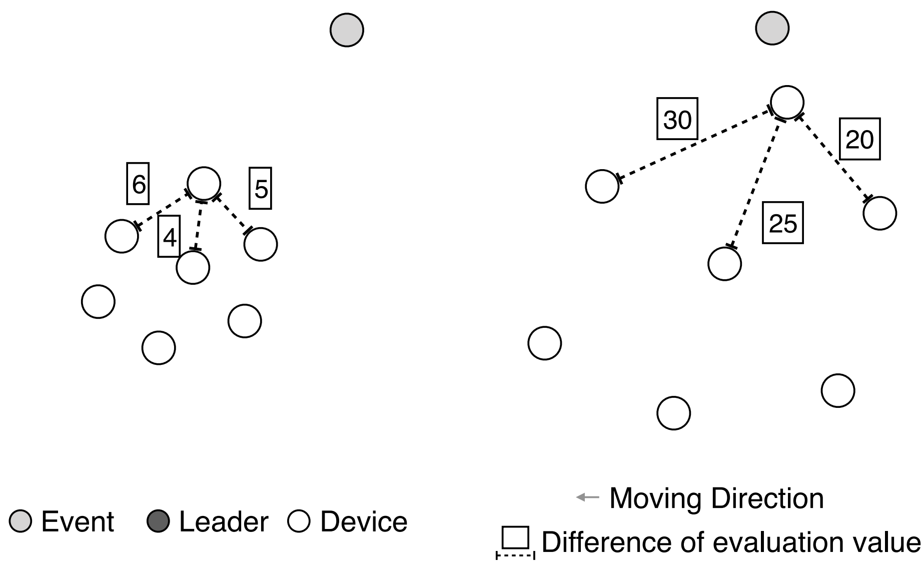
(a) Behavior of devices in no noisy environment (b) Pbest error of a leader in noisy environment (c) Leader selection error in noisy environment

Figure 5.1: Value, Direction, and Environment for internal consistency.

is, the sensing strength of the physical information increases, and turns its direction to the opposite direction when the personal best evaluation value increases. But as shown in Fig.5.1b, the leader moves in an incorrect direction because the strength of the physical information from events oscillates by random error, and a decrease in the personal best evaluation value does not necessarily correspond to the approach to an event. Consequently, MSC requires more time to search for events. The incorrect behavior is called Pbest error.

The other is the leader-selection error shown on Fig.5.1c. In a noisy environment, the strength of the physical information from events oscillates by random error, and the decrease in the personal best evaluation value does not necessarily correspond to the approach to an event. Therefore, a device with the smallest personal best evaluation value is not necessarily nearest the event (device j on Fig.5.1c). An incorrect device was selected as a leader in the swarm. Consequently, the swarm spent more time searching for and actuating events.

The Pbest error is incorrect behavior in an individual leader, who retains the nearest device to an event without any occurrence in the leader-selection error. On the other hand, the leader-selection error is incorrect behavior in the swarm; all the devices behave incorrectly. Therefore, the leader-selction error more strongly impacts the searching actuation performance than the Pbest error.



(a) Behavior of devices in no noisy environment

(b) Pbest error of a leader in noisy environment

Figure 5.2: Direction in evaluation value by dynamic swarm-scale control.

5.3 Dynamic swarm-scale mechanism

The dynamic swarm-scale MSC avoids the leader-selection error in a noisy environment by using a dynamic swarm-scale mechanism. If the distance is small between the devices in a swarm, that is, the swarm's scale is spatially small, the devices in it are located near each other, and their evaluation values that are near the event are also close to each other. Therefore, the relation between their evaluation values on spatially small-scale swarms is easily disordered by random error in the physical information, and leader-selection error occurs easily (Fig.5.2a). In other words, a spatially small-scale swarm sensitively reacts to random error in the physical information. On the other hand, if the distance between devices in a swarm is large, that is, if the swarm scale is spatially large, the devices in it are located far from each other, and their evaluation values are clearly different. Therefore, the relation between their evaluation value on a spatially large-scale swarm is difficult to disrupt by random error in the physical information, and leader-selection error rarely occurs (Fig.5.2b). In other words, a spatially large-scale swarm can absorb the random error in its physical information.

As mentioned above, a dynamic swarm-scale mechanism spatially inflates the swarm to eliminate leader-selection error. However, if the swarm scale remains large, the number of actuating devices

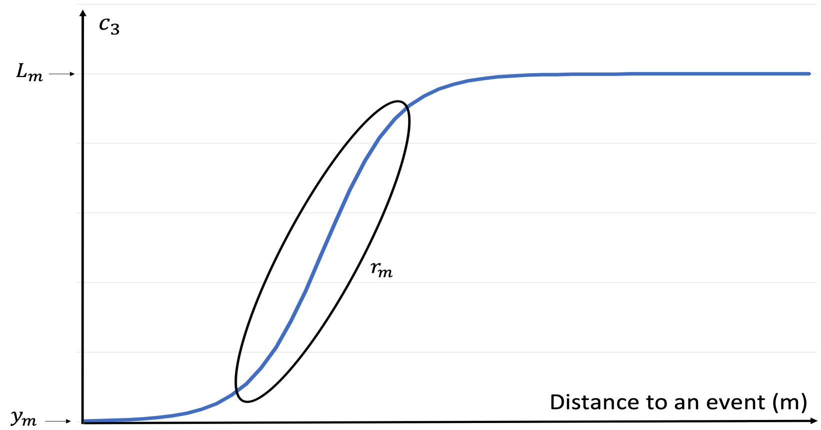


Figure 5.3: Transition of c_3 with dynamic swarm-scale mechanism.

around an event decreases. On the other hand, the random errors in the physical information will probably decrease when approaching an event. Therefore, a dynamic swarm-scale mechanism spatially shrinks the swarm as it approaches events and obtains a sufficient number of devices in the actuation phase. At a point far from the event, the mechanism inflates the swarm's scale to absorb random errors and shrinks its scale for more devices to actuate when approaching an event. For the above behavior to emerge in a swarm, the mechanism utilizes the avoidance weight in Eq.(3.12), which represents the repulsive force among the devices:

$$c_3^i(t) = \frac{y_c L_c}{y_c + (L_c - y_c)e^{-r_c E_i(t)}}, \tag{5.1}$$

where y_c is a lower limit of the avoidance weight, L_c is its upper limit, and r_c is its slope.

In Fig.5.3, the x-axis shows the distance to an event, and the y-axis shows the c_3 in Eq.(5.1). Based on Fig.5.3, c_3 inflates the scale of a swarm at a point far from an event and shrinks it when approaching an event by controlling the avoidance weight based on a device's evaluation value.

5.4 Simulation evaluation

This simulation validates the effectiveness of dynamic swarm-scale MSC.

5.4.1 Simulation specifications

The simulation parameters are listed in Table 5.1. The devices and events are defined as follows:

- a device is equipped with an IEEE802.11b interface and periodically advertises its information (section 3.2.8).
- an event is equipped with an IEEE802.11b interface and periodically advertises a beacon that includes event identities as an MAC address.

Each device receives information from its neighboring devices and beacons from the events. The information's RSSI from the neighboring devices and the beacons from the events are treated as a dBm value. Each device also derives the following three evaluation values:

Table 5.1: Simulation parameters of evaluation for dynamic swarm-scale MSC.

Parameters	Values
Simulator	ns3
Simulation time (sec)	5000
Number of trials for each simulation scenario	10
Number of devices	10~30
Number of events	1,10~30
Initial location of devices (m×m)	(30,30), Uniform distribution
Initial location of events (m×m)	(100,100), Uniform distribution
Update cycle of velocity vector (sec)	0.1
Inertia weight w	0.5
Avoidance degree k	2
Coefficient of event crowd degree	-10
Random number space for β in Eq.(3.5)	$[-30, 30]$
M^{upper} in search phase (m/sec)	1
M^{upper} in actuation phase (m/sec)	0.3
Actuation capacity of event	300
Wireless communications	IEEE802.11b
Transmission power (dBm)	17.0206

Table 5.1: (cont'd) Simulation parameters of evaluation for dynamic swarm-scale MSC.

Parameters	Values
Fading model	Rician fading
K-factor (dB)	1
Transition threshold to actuation phase (dBm)	-50.6262
Distance to collision D_c (m)	1
y_c in Eq.(5.1)	5
L_c in Eq.(5.1)	1000
r_c in Eq.(5.1)	0.1 and 0.3

- Personal best evaluation value (E_i^{Pbest})

  Based on Eqs.(3.19) and (3.20), a personal best evaluation value is defined:

$$E_i^{Pbest(K)}(t) = \min_{k \in discovery_i(t)} \{|RSSI_i^k(t)| + c_4|D_i^K(t)|\}, \tag{5.2}$$

  where $RSSI_i^k(t)$ is the RSSI of a beacon that device i receives from K at t and $discovery_i(t)$ is a set of events from which device i receives beacons at t. If a device cannot receive a beacon from any event, let the personal best evaluation value be a positive infinity.

- Local best evaluation value (E_i^{Lbest})

  Based on Eqs.(4.1) and (4.2), a local best evaluation value is defined:

$$E_i^{Lbest}(t) = \min_{j \in neighbor} \{E_j(t) + c_4|N_i^j(t)|\}. \tag{5.3}$$

- Self-evaluation value (E_i)

Based on Eq.(3.9), a self-evaluation value is defined:

$$E_i(t) = \begin{cases} E_i^{Pbest(K)}(t) \\ \quad if \;\; E_i^{Pbest(K)}(t) < \min\limits_{j \in neighbor_i(t)} \{E_j^{Pbest(K)}(t)\} \\ E_i^{Lbest} + |RSSI_i^{Lbest}(t)| \\ \quad otherwise, \end{cases} \tag{5.4}$$

where $RSSI_i^{Lbest}(t)$ is the RSSI of the information that device i received from a device treated as a local best device at t.

Each device derives the above evaluation values based on a beacon from the event and interactive information between neighbors, moves to the updated location with the evaluation values, and advertises its information with updated location and evaluation values. Each device performs the sequence in 1-sec periods. If the distance between devices becomes lower than a threshold (D_c), it judges that they have collided. Assuming that the devices fail due to a collision, the colliding devices stop and quit advertising their information.

An event has an actuation capacity, which is the room necessary to complete an event's capture. The device in the actuating phase decreases 1 actuation capacity per sec. When the actuation capacity of an event becomes 0, it disappears from the simulation field.

In this simulation, we assumed that the physical information from an event is radio waves. As the radio propagation model, we applied the rice-fading model [58] to simulate the random error of physical information as one example to the simulation evaluation, and the RSSI includes random errors based on it.

5.4.1.1 Comparative evaluation

In this simulation, we compared the four methods listed in Table 5.2. Methods 1 and 2 are mechanisms based on MSC, as described in chapter 3. The c_3 of the previous method 1 is 25, and the c_3 of the previous method 2 is 1000. Therefore, the previous method 1 searches for an event by a constantly small-scale swarm, and the previous method 2 searches for an event by a constantly large-scale swarm. Our proposed methods 1 and 2 are mechanisms with a dynamic swarm-scale mechanism. The r_c in Eq.(5.1) of proposed method 1 is 0.3, and that of proposed method 2 is 0.1. That is, proposed

Table 5.2: Comparison methods of evaluation for dynamic swarm-scale MSC.

	c_3 in search phase	c_3 in search phase
Previous method 1	25	5
Previous method 2	1000	5
Proposed methods 1	c_3 is controlled dynamically	
Proposed method 2	c_3 is controlled dynamically	

method 1 shrinks the swarm scale more rapidly than proposed method 2 when approaching an event. The four methods were evaluated by turnaround time, which is the time to finish searching for and actuating all events. If the devices cannot complete searching for and actuating at all events, let the turnaround time be the simulation time.

5.5 Simulation Results

The simulation results, which are the turnaround times, the dependence of turnaround times on actuating times and the dependence of turnaround times on searching times, are shown below.

5.5.1 Turnaround times

The simulation results are shown in Fig.5.4. Proposed method 1 outperformed previous methods 1 and 2 regardless of the number of devices and events. Proposed method 1 was almost equivalent to proposed method 2 when the number of events was 1, but it was superior in other cases. Proposed method 1 maintained a large swarm scale and shrunk it when an event was rapidly approaching. To absorb random errors, maintaining a large swarm scale until approaching an event is effective. However, when approaching an event, shrinking the swarm scale is effective because of the significant decrease in the random errors from an event.

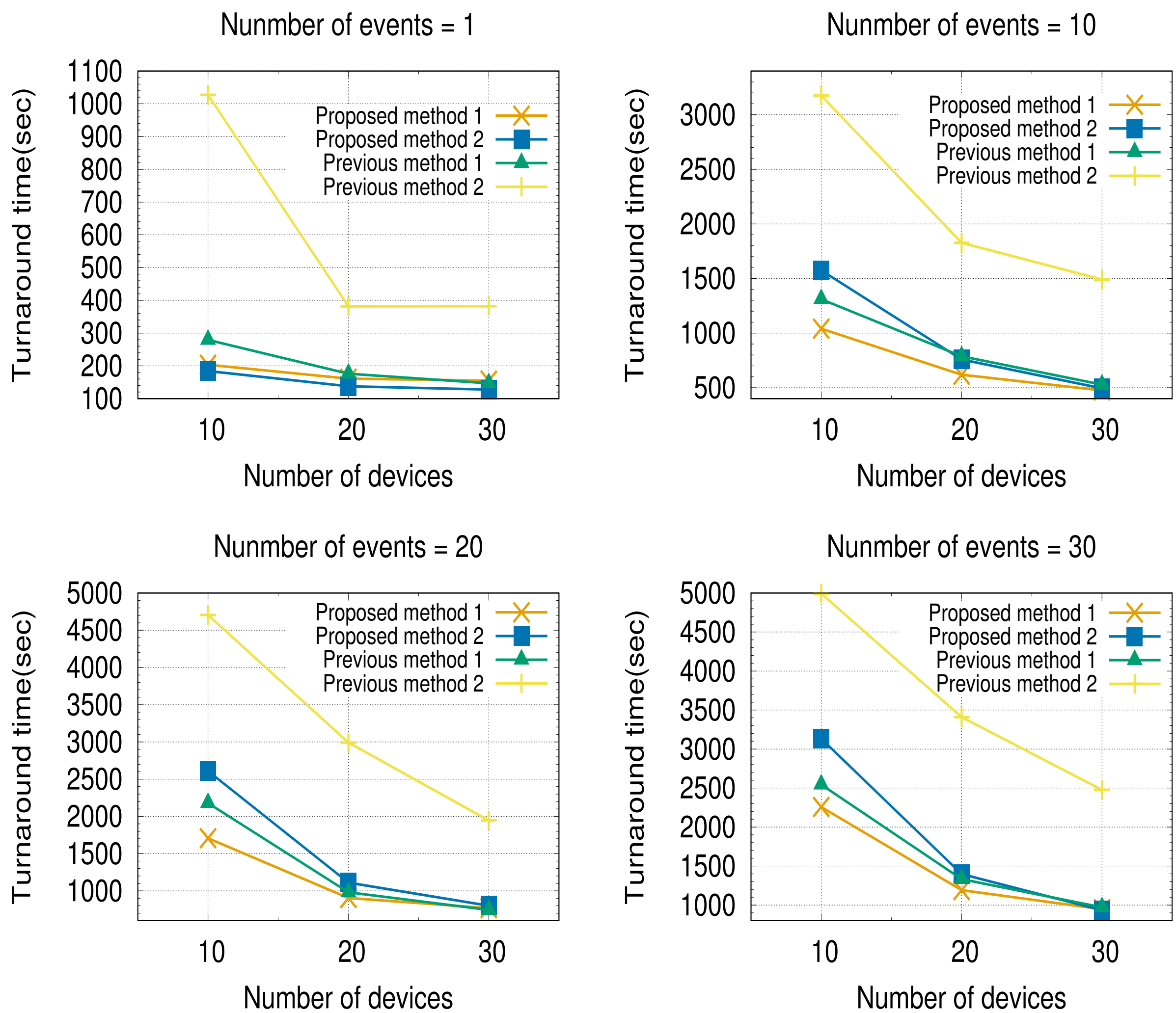


Figure 5.4: Turnaround times when actuation capacity is 300.

5.5.2 Dependence of turnaround times on actuating times

The dependence of the turnaround times on the actuating times when the actuation capacity is varied from 100 to 500 is shown in Fig.5.5. Proposed methods 1 and 2 and previous method 1 were almost equivalent. Previous method 2 was significantly inferior to the others. Proposed methods 1 and 2 shrank the swarm scale near an event, and previous method 1 kept it small; therefore, a large number of devices could actuate an event, which reduced the time. Previous method 2 kept the swarm scale large, and the number of devices near an event was small; the number of actuating devices to the event was also small. Therefore, previous method 2 required more time to actuate an event.

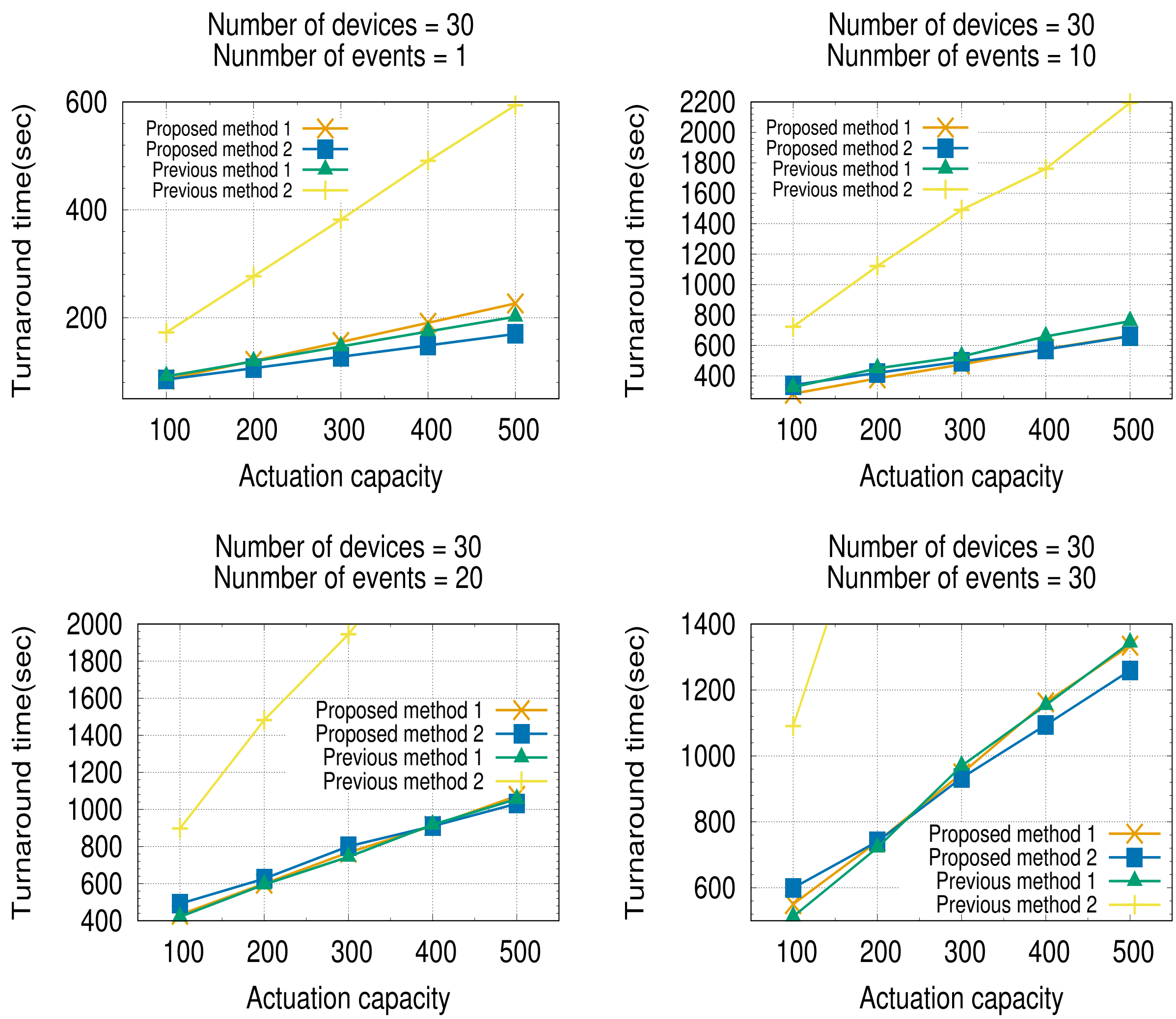


Figure 5.5: Dependence of turnaround times on actuation capacity.

5.5.3 Dependence of turnaround times on searching times

Figure 5.6 shows the turnaround time when the actuation capacity was 1. Proposed method 1 outperformed the others regardless of the number of devices and events. The swarm-scale mechanism (section 5.5.1) increased the search capability and absorbed random errors.

Table 5.3 lists the ratios of the leader-selection error and the pbest error of a leader. As mentioned above, the leader-selection error is incorrect behavior when the device, which is not the nearest to an event, is selected as a leader. The pbest error is incorrect behavior when the leader moves away from an event due to pbest with random error.

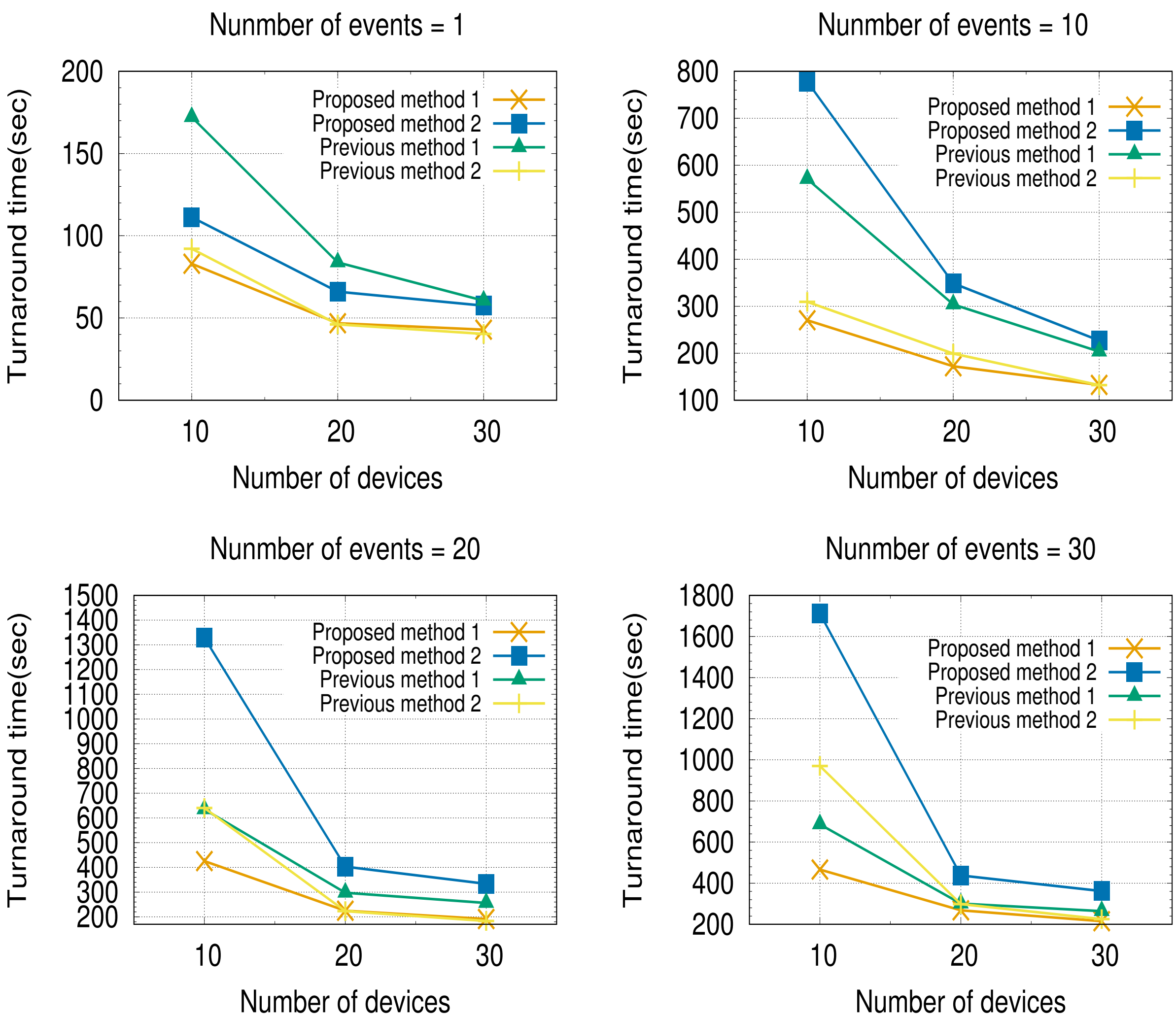


Figure 5.6: Turnaround times when actuation capacity is 1.

Table 5.3 shows that the pbest error of a leader for each method ranged from 28 to 37%, and the maximum difference among the methods was about 10%. The leader-selection error for each method ranged from 48 to 74%, and the difference among previous method 1, which kept the swarm scale small, and the other three methods, which kept the swarm scale large at a point far from an event, was about 25%. The leader was correctly selected by inflating the swarm scale at a point far from an event. By inflating the swarm scale, the difference in the evaluation value between devices increased and became so large that it was not affected by the random error. As a result, the disorder in the relations between the personal best evaluation values derived by each device decreased, and the

Table 5.3: Ratio of two incorrect behaviors in noisy environment.

	Leader-selection error	Pbest error of leader
Previous method 1	74.03%	37.69%
Previous method 2	48.86%	34.52%
Proposed method 1	51.38%	28.77%
Proposed method 2	57.42%	34.65%

leader was correctly selected.

Based on the above simulation results, our proposed method, which dynamically controls a swarm's scale, reduces the turnaround time even in a noisy environment.

5.6 Conclusions of dynamic swarm-scale MSC

In this chapter, a dynamic swarm-scale MSC was proposed to search for and actuate a large number of events in noisy environments where the strength of the physical information from an event oscillates due to random error. The proposed method spatially inflated a swarm's scale in the search phase and causes the swarm to absorb the oscillation in the strength of the physical information by random error and to accept such error. The proposed method also shrinks the scale of the swarm as it approaches events, and as the random error in the physical information from events decreases, the swarm rapidly begins to focus on events.

Our simulation evaluations showed that the proposed method decreased the time to search for and actuate multiple events in a noisy environment.

Chapter 6

Conclusions

As alternatives to humans in the near future, such autonomous mobile devices as robots and unmanned aerial vehicles are expected to search for and actuate diverse emergent events whose locations and numbers are unknown. When an autonomous mobile device searches for and actuates unknown events, it searches based on sensing the physical information emitted by an event, such as smell and temperature, and when it reaches an event, it processes the event based on its actuating function. In this thesis, we proposed a Mobile Sensing Cluster (MSC) to search for and actuate a large number of events whose locations and numbers are unknown in a limited time. MSC expands Particle Swarm Optimization (PSO) to achieve appropriate balancing of parallelism and collaboration in searching for and actuation of multiple unknown events and can dynamically form multiple swarms. In MSC with wireless communications, each devise shares searching and actuating information among its neighbors and forms multiple swarms with them to search for and actuate multiple unknown events. In this thesis, we proposed three MSC schemes and applied them to the following assumptions for events.

- The strength of the physical information emitted by an event increases accordingly as approaching event, and the events are distinguishable from each other based on its emitting physical information.

- The strength of the physical information emitted by an event monotonically increases accordingly as approaching event, and the events are indistinguishable from each other based on its emitting physical information.

- The strength of the physical information emitted by an event includes random error, and the events are distinguishable from each other based on its emitting physical information.

In chapter 3, we proposed an MSC for the strength of the physical information emitted by events monotonically increasing as approaching and distinguishable events, and described the following mechanisms that constitute MCS:

- a mechanism that forms a swarm based on PSO.
- a mechanism that dynamically selects the nearest device to an event as a swarm's leader.
- a mechanism that avoids collisions between devices.
- a mechanism that continuously searches for and actuates multiple events.
- a mechanism that dynamically divides a swarm into multiple swarms and balances the swarm size among multiple swarms.

Furthermore, MSC effectiveness was shown by the simulation evaluation as follows:

- MSC can search for and actuate multiple unknown events in a short time.
- In both searching and actuating performances, MSC is overwhelmingly superior to the independent parallel and collective methods.

In chapter 4, we proposed a mutation MSC for the strength of physical information emitted by events that monotonically increase as approaching indistinguishable events and described the following two mechanisms introduced to it:

- The swarm local division mechanism searches for and actuates multiple indistinguishable events in local areas by a mutation from follower to leader.
- Swarm global division mechanism searches for and actuates multiple indistinguishable events in a wide area by mutation from normal to heretic devices.

In the simulation evaluation, the effectiveness of a mutation MCS was shown:

- The mutation MSC largely decreased the turnaround time more than the MSC and a random search.

- The difference of the turnaround times increased as the field size increased.
- The search and actuation performance in the mutation MSC is dependent on the upper limit for the number of heretic devices and can be controlled by using it.

In chapter 5, we proposed a dynamic swarm-scale MSC for the strength of the physical information from distinguishable events with random error and described the mechanism in a dynamic swarm-scale MSC:

- In the search phase, the mechanism inflates the swarm and provides it with a tolerance capability to random error in the physical information from events.
- In the actuation phase and where the swarm is near events, the mechanism shrinks the swarm and provides it with high actuation capability to events.

The simulation evaluation showed that the mechanism decreases the time to search for and actuate multiple events in a noisy environment.

In all of the above assumptions for events, the proposed mechanisms for MSC that forms multiple swarms by using autonomous mobile devices can quickly search for and actuate a large number of multiple unknown events, and their effectiveness is very high.

Finally, we describe the future works. The future works are as follows:

- Fusion with mutation MSC and dynamic swarm-scale MSC.
- Implementation and experiment for the above fusion MSC.

The first work will adapt MSC to the indistinguishable event in nosy environment. The mutation MSC is adapted to the indistinguishable event whose strength overlaps with others and the dynamic swarm-scale MSC is adapted in noisy environment with random errors. Therefore, The adaptation of MSC to indistinguishable event in nosy environment will be realized by the fusion with the two schemes for MSC.

The second work will verify the effectiveness of the fusion MSC in a real environment where the event is indistinguishable and where the strength of physical information from event includes random errors. For the verification, the fusion MSC will be implemented and experimented, by using iRobot Roomba or E-puck2 as autonomous mobile robot, and by using Bluetooth or WiFi as wireless communications.

Bibliography

[1] Sugisaka, M.: Working Robots for Nuclear Power Plant Desasters, *IEEE DEST 2011*, pp-358-361(2011).

[2] Sarkar, S., Patil, Akash., Hartalkar, A and Wasekar, Adiya.: Earthquake rescue robot: A purview to life, *2017 Second ICECCT*, pp.1-7(2017).

[3] Allan, C., Sibonelo, M. and Riaan, S.: Survey and requirements for search and rescue ground and air vehicles for mining applications, *M2VIP*, pp.105-109(2012).

[4] Qianying, P. and Hongtao Y: Survey of particle swarm optimization algorithm and its application in antenna circuit, *2015 IEEE ICCP*, pp.492-495(2015).

[5] Dayal, R.P., Jayanta, K. and Mukesh, K.S.: Navigation of Multiple Mobile Robots using Swarm Intelligence, *2009 NaBIC*, pp.1145-1149(2009).

[6] Hua, B. and Bo, Z.: A Survey on Application of Swarm Intelligence Computation to Electric Power System, *2006 6th World Congress on Intelligence Control and Automation*, pp.7587-7591(2006).

[7] Kassabalidis, I., EI-Shakawi, M.A., Marks, R.J., Arabshahi, P. and Gray, A.A.: Swarm Intelligence for Routing in Communication Networks, *GLOBECOM'01*, Vol.6, pp.3613-3617(2001).

[8] Shiqiang, W., Jianchun, X., Xiyan, J. and Yunchuang, D.: A Decentralized Swarm Intelligence Algorithm for Global Optimization of HVAC System, *IEEE Access*, Vol.7, pp.64744-64757(2019).

[9] Shahana, K., Subhajit, G. and C.J.: A Survey of Particle Swarm Optimization and Random Forest based Land Cover Classification, *ICCCA2016*, pp.241-245(2016).

[10] Feng, Y., Pengxiang, W., Yizhai, Z., Litao, Z and Jianchun, L.: Survey of Swarm Intelligence Optimization Algorithms, *2017 ICUS*, pp.544-549(2017).

[11] Cameron, J., Ganesh, K.V. and Parviz, P.: Hardware Implementations of Swarming Intelligence - A Survey, *2008 IEEE Swarm Intelligence Symposum*, pp.1-9(2008).

[12] James, K. and Russell, E.: Particle Swarm Optimization, *in Proc. the 1995 IEEE International Conference on Neural Networks*, pp.1942-1948(1995).

[13] Kangtai, W. and Fupeng, L.: A dynamic chaotic mutation based particle swarm optimization for dynamic optimization of biochemical process, *ICISCE*, pp.788-791(2017).

[14] Yuanbin, M., Hetong, L. and Qin, W.: Conjugate direction particle swarm optimization solving systems of nonlinear equations, *COMPUT MATH APPL*, Vol.57, Issues.11-12, pp.1877-1882(2009).

[15] Le, Y., Dakuo, H., Qingkai, W., Jiahuan, L., Yingjie, H and Zipeng, Z.: Particle Swarm Optimization Algorithm Based on Robust Control of Random Discrete Systems, *ICISCE*, pp.1089-1093(2017).

[16] Soririos, S.: Particle Swarm Optimization and Application in Robotics: A Survey, *0218 9th IISA*, pp.1-7(2018).

[17] Reynolds, W.C.: Flocks herds and schools:A distributed behavioral model, *SIGGRAPH Comput. Graph.*, Vol.21, No.4, pp.25-34(1987).

[18] Eversham, J., Ruiz, F.V.: Parameter analysis of Reynolds flocking model, *2010 IEEE 9th International Conference on Cyberntic Intelligent Systems*, pp.1-7(2010).

[19] Hauert, S., Leven, S., Varga, Maja. and Ruini, F.: Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.5015-5020(2011).

[20] Zhiwei, Y., Wei, L., Hongwei, C. and Enbo, Z.: A Novel Feature Selection Approach Based on Swarm Intelligence, *2009 International Workshop on Intelligent Systems and Applications*, pp.1-4(2009).

[21] Yan-Fei, Z. and Xiong-min, T.: Overview of Swarm Intelligence, *ICCASM 2010*, pp.V9-400 - V9-403(2010).

[22] Kritele, L., Benhala, B. and Izeddine, Z.: Optimal digital IIR filter design using ant colony optimization, *2018 4th ICOA*, pp.1-5(2018).

[23] Gu, P., Xiu, C. Cheng, Y., Luo, J. and Li, Y.: Adaptive ant colony optimization algorithm, *2014 ICMC*, pp.95-98(2014).

[24] Siyuan, Z.: Cloud Computing Analysis and Optimization Based on Map-Reduce and Improved Ant Colony Optimization Algorithm, *2018 IEEE 9th ICSESS*, pp.391-394(2018).

[25] Ramlakhan, S. and Unmukh, Datta.: Modified Ant Colony Optimization Algorithm with Uniform Mutation using Self-Adaptive Approach for Travelling Salesman Problem, *2013 ICCCNT*, pp.1-4(2013).

[26] Ruey-Maw, C., Yin-Mou, S and Ching-Te, W.: Ant Colony Optimization Inspired Swarm Optimization for Grid Task Scheduling, *IS3C, International Symposium on*, pp461-464(2016).

[27] Victor-Emil, N. and Catalina-Elena, N.: An Artificial Bee Colony Approach for Classification of Remote Sensing Imagery, *2018 10th ECAI*, pp.1-4(2018).

[28] Lei, W., Xin, Z. and Xiu, Z.: Antenna Array Design by Artificial Bee Colony Algorithm With Similarity Induced Search Method, *IEEE Transactions on Magnetics*, Vol.55, Issue.6(2019).

[29] Dervis, K. and Bahriye, A.: A Comparative Study of Artificial Bee Colony Algorithm, *Applied Mathematics and Computation*, Vol.214, Issue.1, pp.108-132(2009).

[30] Dervis, K.: An Idea based on Honey Bee Swarm for Numerical Optimization, *Tech. Rep. TR06, Erciyes University, Engineering Faculty, Computer Engineering Department*(2005).

[31] Saber, O.R., Fax, A.J. and Murray, M.Richard.: Consensus and Cooperation in Networked Multi-Agent Systems, *Proceeding of the IEEE*, Vol.95, pp.215-233(2007).

[32] Lingyu, L. and Weili, N.: Consensus Problems on Multi-agent Networks with Directed Dynamic Interactions, *2017 6th DDCLS*, pp.421-426(2017).

[33] Qiang, J. and Wallce, K.S.T.: Consensus of Multi-Agents with Event-Based Nonlinear Coupling over Time-Varying Digraphs, *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol.65, Issues.12(2018).

[34] Schaffer,J.D., Whitley, D. and Eshelman, L.J.: Combinations of genetic algorithms and neural networks: a survey of the state of the art, Proc.*COGANN-92*,pp.1-37(1992).

[35] Koza, J.R.:Survey of genetic algorithms and genetic programming, Proc.*WESCON'95*,pp.589(1995).

[36] Sheikh, R.H., Raghuwanshi, M.M. and Jaiswal, A.N.: Genetic Algorithm Based Clustering: A Survey, Proc.*2008 First International Conference on Emerging Trends in Engineering and Technology*, pp.314-319(2008).

[37] Guo, P., Wang, X. and Han, Yingshi.: The enhanced Genetic Algorithms for the Optimization Design, Proc.*BMEI 2010*, Vol.7, pp.2990-2994(2010).

[38] Gnanaprasanambilkai, L. and Munusamy, N.: Survey of genetic algorithm effectiveness in intrusion detection, Proc.*I2C2*,pp1-5(2017).

[39] Adham, A. and David, M.W.P.: The Use of Area Extended Particle Swarm Optimization (AEPSO) in Swarm Robotics, *2010 11th International Conference on Control Automation Robotics & Vision*, pp.591-596(2010).

[40] Siddarth, J., Manish, S. and Vijay, K.C.: Ad-hoc swarm robotics optimization in grid based navigation, *2010 11th International Conference on Control Automation Robotics & Vision*, pp.1553-1558(2010).

[41] Benjamin, T.C. and Matthew, A, J.: Underwater swarm robotics review, *2015 10th SoSE*(2015).

[42] Tomas, D. and Zdenek, S.: Control system based on FreeRTOS for data acquisition and distribution on swarm robotics platform, *2017 18th ICCC*(2017).

[43] Edward, B., Teddy, M., Timothy, A., Dhvani, R., Ian, B. and Yongqiang, W.: An undergraduate research platform for cooperative control and swarm robotics, *2016 IEEE ICIEA*, pp.18767-1879(2016).

[44] Jan, D., Matej, G. and Zdenek, S.: Sensoric system for navigation of swarm robotics platform, *2017 18th ICCC*, pp.429-433(2017).

[45] Tan, Y. and Zhong-yang, Z.: Research Advance in Swarm Robotics, *Defence Technoloty*, Vol.9, Issues.1, pp.18-39(2013).

[46] Francesco, M., André, G., Michael, B., D, B., Michel, L. and Dario, F.: SWARM-BOT: from concept to implementation, *Proceedings 2003 IEEE/RSJ IROS 2003*, Vol.2, pp.1626-1631(2003).

[47] Francesco, M., Luca, M.G., Dario, F., Stefano, N., J.-L.D. and Marco, D.: The cooperation of swarm-bots: physical interactions in collective robotics, *IEEE Robotics & Automation Magazine*, Vol.12, Issue.2, pp.21-28(2005).

[48] Gautam, W., Alex, N.J.R., Zhun F.: Swarm bots: System design for ECHOLOCATION, *2017 3rd ICCAR*, pp.229-232(2017).

[49] Roderich, G., Michael, B., Francesco, M. and Marco, D.: Autonomous Self-Assembly in Swarm-Bots, *IEEE Transactions on Robotics*, Vol.22, Issue.6, pp.1115-1130(2006).

[50] Marco, D., Dari, F., Luca, M. G., Francesco, M., Stefano, M., Tarek, B., Mauro, B., Michael, B., Manuele, B., Arne, V., Daniel, B., Alexandre, C,. Anders, L.C., Antal, D., Gianni, D.C., Frederick, D., Eliseo, F., Alexander, F., Javier, M. G., Jerome, G., Valentin, L., Stephane, M., Nithin, M., Marco, M. O., Rehan, O., Carlo, P., Giovanni, P., Philippe, R., James, R., Valerio, S.: Swarmanoid: A Novel Concept for the Study of Heterogeneous Robotic Swarms, *IEEE Robotics & Automation Magazine*, Vol.20, Issue, 4, pp.60-71(2013).

[51] Van, T.L. and Trung, D.N, Virtual pheromone based information foraging in modular robotics, *2014 11th URAI*, pp.379-384(2014).

[52] Farshad, A., Ali, E.T., Tomáš, K., Salar, R., ⊠lkin, E.O., Shigang, T., Simon, W., Barry, L.: Clust: Pheromone-Based Aggregation for Robotic Swarms, *2018 IEEE/RSJ IROS*, pp.4288-4294(2018).

[53] McLurkin, J. and Smith, J.: Distributed algorithms for dispersion in indoor environment using a swarm of autonomous mobile robots, *Distributed autonomous robotic system*, pp.399-408(2007).

[54] Lenka, P., Alan, W. and Richard, C.: Recruitment Near Worksites Facilitates Robustness of Foraging E-Puck Swarms to Global Positioning Noise, *2018 IEEE/RSJ IROS*, pp.4276-4281(2018).

[55] Dip, S.B, Shashank, S. and Kishore, K.: Work efficient parallel algorithms for large graph exploration, *20th Annual International Conference on High Performance Computing*, pp.433-442(2013).

[56] Mustafa, S., William, A.P., Masoud, F.: SPIHT algorithms using Depth First Search Algorithm with minimum memory usage, *2006 20th Annual Conference on Information Sciences and Systems*, pp1158-1163(2006).

[57] Sakurama, K., Yasuki, H. and Nishida, S.: Multiple Extrema Seeking by Swarm Robots Based on Distributed Optimization, *Journal of The Institute of Systems, Control and Information Engineers*, Vol.29, No.4, pp.155-161(2016).

[58] Amirhossein, A., Saeed, F.F., Bruce, F.C. and Christian, S.: Compact Rayleigh and Rician fading simulator based on random walk processes, *IET*, Vol.3, Issue.8, pp.1333-1342(2009).

Acknowledgment

I would like to thank Dr. Yasuhisa Takizawa who supervised my doctoral research. He gives insightful comments and suggestions and I am deeply grateful to him for useful discussions. I would also like to thank the referees for this thesis, Dr. Yeboon Yun and Dr. Hiroyuki Yomo, for detailed review of this thesis and for providing valuable comments to improve this thesis.

I would like to thank everyone in network engineering laboratory of Kansai University for their kind support. I am grateful to my co-authors in previous articles, Wataru Hirose, Shizuka Wasiyama, Shoma Nishigami, Dr. Takamasa Kitanouma, Dr. Naoya Adachi and Dr. Hiroyuki Yomo, for their help.

List of Previous Publications and Researches

Publications(peer reviewed)

First author

1. Nii, E., Kitanouma, T., Hirose, W., Yomo, H. and Takizawa, Y.: Mobile Sensing Cluster based on Swarm Intelligence with Multiple Autonomous Mobile Devices, *IPSJ journal*,(in Japanese), Vol.59, No.12, pp.2201-2212(2018.12).

2. Nii, E., Kitanouma, T., Yomo, H. and Takizawa, Y.: Proposal of Mutation Mobile Sensing Cluster for Indistinguishable Multiple Events(to be published in IPSJ Journal) (in Japanese).

Co-author

1. Kitanouma, T., Nii, E., Adachi, N. and Takizawa, Y.:Cloud-based Self-organizing Localization with Virtual Network Topology for Wireless Sensor Networks and Its Verification Experiments, *IPSJ journal*, (in Japanese), Vol.58, No.2, pp.448-460(2017.2).

2. Kitanouma, T., Nii, E., Adachi, N. and Takizawa, Y.:SmartFinder: Cloud-based Self Organizing Localization for Mobile Smart Devices in Large-scale Indoor Facility and Its Evaluation, *IPSJ journal*, (in Japanese), Vol.59, No.2, pp.462-472(2018.2).

3. Kitanouma, T., Nii, E., Mori, R. and Takizawa, Y.:Real Space Evaluation on Cloud-based Self Organizing Localization SmartFinder with inter-node path length, *IPSJ journal*, (to be published in IPSJ Journal), (in Japanese).

International Conference(peer reviewed)

First author

1. Nii, E., Kitanouma, T., Adachi, N. and Takizawa, Y.: Cooperative detection for falsification and isolation of malicious nodes for wireless sensor networks in open environment, 2017 IEEE Asia Pacific Microwave Conference(APMC), Malaysia(2017.11).

2. Nii, E., Shizuka, W., Kitanouma, T. and Takizawa, Y.: Dynamic Multiple Swarming for Mobile Sensing Cluster based on Swarm Intelligence, 2019 IEEE 5th World Forum on Internet of Things(WF-IoT), Ireland(2019.4).

3. Nii, E., Nishigami, T., Kitanouma, T., Yomo, H. and Takizawa, Y.:Behavior Control of Swarm Composed of Multiple Autonomous Mobile Devices to Random Error, 2019 Global Information Infrastructure and Networking Symposium (GIIS), France(2019.12).

Co-author

1. Kitanouma, T., Nii, E., Takashima, Y., Adachi, N. and Takizawa, Y.: SmartFinder: Cloud-based self organizing localization for mobile smart devices in large-scale indoor facility, 2017 Global Internet of Things Summit (GIoTS), pp.1-6 (2017.6).

2. Asada, A., Yomo, H., Nii, E. and Takizawa, Y.:Mobility Control Exploiting Swarm Intelligence for Mobile Sensing Searching for Indistinguishable Multiple Targets, Proc. of IEEE International Conference on Systems, Man and Cybernetics (SMC 2019)(2019.10).

3. Kawata, T., Kitanouma, T., Nii, E., Mori, R. and Takizawa, Y.: Self-Organizing Localization with multiplexed topology for smart devices in indoor space, 2019 Global Information Infrastructure and Networking Symposium (GIIS), France(2019.12).

4. Kimura, Y., Nii, E. and Takizawa, Y.:Cooperative Detection for Falsification and Isolation of Malicious Node with Inter-node Vote for Wireless Sensor Networks in Open Environment, 2019 Global Information Infrastructure and Networking Symposium (GIIS), France(2019.12).

Oral

First author

1. 新居英志・北乃馬貴正・滝沢泰久:群知能を用いた移動センシングクラスタ, 研究報告マルチメディア通信と分散処理 (DPS), Vol.2017-DPS-171, No.27, pp.1-8,2017, 情報処理学会第171 回マルチメディア通信と分散処理研究会, 北谷美浜メディアステーション (2017.5).

2. 新居英志・北乃馬貴正・滝沢泰久:複数移動体における集団探索メカニズムの考察, 情報処理学会第 80 回全国大会公演論文集, Vol.2018, 2018, 情報処理学会第 80 回全国大会, 発表番号 6D-07, 早稲田大学 (2018.3).

3. 新居英志・北乃馬貴正・滝沢泰久:移動センシングクラスタにおける群構成に関する考察, 研究報告マルチメディア通信と分散処理 (DPS), Vol.2018-DPS-175, 情報処理学会第 175 回マルチメディア通信と分散処理研究会, イーフ情報プラザ (2018.5).

4. 新居英志・西上翔磨・四方博之・北乃馬貴正・滝沢泰久:ノイズ環境下における移動センシングクラスタの群制御改良方式, 研究報告マルチメディア通信と分散処理 (DPS), Vol.2019-DPS-180, No.16, pp.1-8, 情報処理学会第 180 回マルチメディア通信と分散処理研究会, 石川県政記念しいのき迎賓館 (2018.5).

Co-author

1. 廣瀬航, 新居英志, 四方博之, 滝沢泰久:無線信号伝搬の影響を考慮した群移動センシング特性評価, 研究報告モバイルコンピューティングとパーベイシブシステム (MBL), Vol.2018-MBL-86, No.40, pp.1-6, モバイルコンピューティングとパーベイシブシステム研究会, 上智大学 (2018.2).

2. 森流星，滝沢泰久，北之馬貴正，新居英志:RTOF を用いた自己組織化ノード位置推 定方式の高精度化に関する考察，情報処理学会第 80 回全国大会，5S-07 (2018.3).

3. 北之馬貴正，新居英志，安達直世，滝沢泰久:SmartFinder:集約型自己組織化スマートデバイス位置推定方式のノード間メトリックを用いた拡張とその実装，研究報告マルチメディア通信と分散処理研究会, Vol. 2018-DPS-175, No.15, pp.1 - 8 (2018.5).

4. 森流星，北之馬貴正，新居英志，安達直世，滝沢泰久:IR-UWB を用いた自己組織化ノード位置推定方式のシミュレーション評価，研究報告マルチメディア通信と分散処 理研究会，Vol. 2018-DPS-175, No.14, pp.1-7 (2018.5).

5. 川田 千尋, 北之馬 貴正, 新居 英志, 森 流星, 滝沢 泰久:集約型自己組織化スマートデバイス位置推定方式における移動デバイス位置推定精度改善手法の提案, 情報処理学会第 81 回全国大会公演論文集, Vol.2019, No.1, pp.261-262, 情報処理学会第 81 回全国大会 (2019.2).

6. 木村圭希, 新居英志, 滝沢泰久:開放環境 WSN における協調的パケット改ざん検知と信頼度に基づく不正ノード孤立化手法の提案, 情報処理学会第 81 回全国大会公演論文集, Vol.2019, No.1, pp.257-258, 情報処理学会第 81 回全国大会 (2019.2).

7. 浅田晃年, 新居英志, 四方博之, 滝沢泰久:識別不可能な複数目的地の探索を実現する移動センシングクラスタ, 電子情報通信学会ネットワークシステム研究会, 2019 年 3 月

8. 木村圭希, 新居英志, 滝沢泰久:開放環境無線センサネットワークにおける協調的パケット改竄検知と多数決手法を用いた不正ノード孤立化手法の提案, 研究報告マルチメディア通信と分散処理 (DPS), Vol.2019-DPS-180, No.17, pp.1-8, 情報処理学会第 180 回マルチメディア通信と分散処理研究会, 石川県政記念しいのき迎賓館 (2019.9).

9. 川田 千尋, 北之馬貴正, 新居英志, 森流星, 滝沢泰久:開放環境無線センサネットワークにおける協調的パケット改竄検知と多数決手法を用いた不正ノード孤立化手法の提案, 研究報告マルチメディア通信と分散処理 (DPS), Vol.2019-DPS-180, No.8, pp.1-7, 情報処理学会第 180 回マルチメディア通信と分散処理研究会, 石川県政記念しいのき迎賓館 (2019.9).

Patent

1. 第 6467334 号，無線通信システム，新居英志，北之馬貴正，滝沢泰久

Awards

1. Recommended Paper of IPSJ, Nii, E., Kitanouma, T. and Takizawa, Y.:群知能を用いた移動センシングクラスタ (2017.5).

2. Best Paper Award of IPSJ National Conventions, Nii, E., Kitanouma, T., Adachi, N. and Takizawa, Y.:Cooperative Detection for Falsification and Isolation of Malicious Node for Wireless Sensor Networks in Open Environment(2017.3).

3. Student Encouragement Award of IPSJ National Convention, Nii, E., Kitanouma, T., Adachi, N. and Takizawa, Y.:Cooperative Detection for Falsification and Isolation of Malicious Node for Wireless Sensor Networks in Open Environment(2017.3).