Designing an Algorithm for Swarm Behavior Using the Concept of *Umwelt*

Ryusuke Fujisawa[1] **Takashi Hashimoto**[2]

1:The University of Electro-Communications
2:Japan Advanced Institute of Science and Technology

Abstract: In this study, we propose a methodology for designing a swarm behavior. The difficulty in designing the swarm behavior is a gap between the object of evaluation and that of design. The former is the performance of a group, but the latter is the action of each individual. We utilize the concept "*Umwelt*" in ethology for bridging the gap. The advantage of this concept is that all actions necessary for the swarm behavior can be derived from the purpose of each individual. Using this concept, the swarm behavior can be built into the action algorithm of the individuals. In order to evaluate the proposed method, we construct the swarm algorithm for search and collection task. Using a computer simulation, we confirmed that the swarm successfully achieved the task with flexibility and parallelism, and also robustness in part. These results support the effectiveness of the proposed methodology.
Keywords: *Umwelt*, Swarm behavior, Pheromone communication

1 Introduction

1.1 Features of a Swarm

A swarm is a kind of distributed autonomous system. Many agents act distributedly without a central controller. There are three remarkable features of a swarm: flexibility, robustness and parallelism.

Flexibility is a characteristic that enables adaption to changing environments. The swarm is constructed by a number of individuals. The swarm of social insects has flexibility with respect to its environment. Social insects correspond with their changing environment as a result of interactions among individuals [1]. For example, army ants grasp each others' bodies and construct a structure like a "ladder" or a "bridge" in certain situations [2].

Robustness is a characteristic that the system can keep its function even if some of its agents cannot work. Since the swarm consists of many agents, the function of the system does not strongly rely on any individual agent. Thus, the swarm can tolerate the wastage of agents and the failure of foraging or returning to a nest.

The swarm can process a lot of tasks in parallel. A number of individuals act autonomously, as a result the group solves multiple tasks at the same time. A typical example is the foraging of the social insects. The foraging task can be divided into search and collection tasks. The social insects solve both tasks in parallel. All individuals do not concentrate on just one object, but search and collect a number of objects in parallel. As a result, the social insects can search a vast environment for collecting the objects.

There are difficulties in utilizing such useful features of the swarm in engineering. The problem is that the entire behavior of the swarm cannot be designed. However, the work performance of the swarm is decided by its entire behavior, which appears as a result of the autonomous actions of individual agents. The object of design is the individual agent, but the object of evaluation remains the swarm behavior. This gap makes the utilization difficult [3]. Basically, a research focusing on "how the swarm behavior is built into individual agents" is needed.

1.2 Related Works

There are some researches that apply swarm behavior to engineering, such as SWARM-BOTS [4] and ant colony optimization (ACO) [5].

The SWARM-BOTS project, in which sociobiologists participate as well as engineers, advances with both real machines and computer simulations. In research involving real machines, multiple robots demonstrate performances higher than individuals by cooperating with each other [6]. Cooperatively, they can pass a gap and a step that an individual cannot pass [7]. In the computer simulation research, it has been shown that a swarm responds to different environments by generating various patterns through interactions among individuals [8]. Although the project deals with various problems, they are solved individually, and general applicability is not discussed in depth.

The ACO is a kind of optimization algorithm in which an ant society's behavior to find the shortest paths between a nest and food sites is applied. The point of the algorithm is to utilize pheromone trails that agents generate. The ACO is applied to some combinatorial optimization problems such as the traveling salesman problem and network routing. However, the method to design the group and to use the pheromone efficiently for various problems remains to be established. To effectively utilize the advantages of swarm behavior, a design method for the swarm behavior needs to be debated.

1.3 Basic Idea for the Swarm

The swarm can acquire only local information about the local and physical environment, nearby agents and their own states as sensed by themselves. In computer simulation of agent behavior, a designer designs an environment of agents and information given to the agents, and decides behavioral rules of the agent naturally based only on the information designed. However, as for robotic systems, while a designer equips robots with particular sensors, the environment of the robots is not fully specified. Thus, the information the robots sense is considerably different from the designer, namely, they live in a different world from us. In spite of this fact, the designer, who has a global view, may often design the behavioral rules from the designer's viewpoint. When we design robotic agents, we should take view-from-within seriously and we think the design methodology based on this consideration is demanded.

Designing the behavior of the autonomous agents should be grounded on the subjective information of the agents. The concept of "*Umwelt*" is of help to design an agent's behavior based on their subjective world. The concept was propounded by Uexküll [9] so as to understand an organism's behavior. An organism is though to have a phylo-specific world, which is called *Umwelt*. The organism detects only needful information for its actions, and acts in the physical world. The *Umwelt* is constituted by needful information and actions. But, Uexküll referred only to an individual organism's behavior. To utilize this concept for the design of the swarm behavior, we need to consider interactions among the *Umwelts* of multiple agents.

The purpose of this study is to propose and evaluate a design method for algorithms related to the swarm behavior. The following three objectives will be focused on:

- Propose a design method for the swarm behavior algorithm using the concept of the *Umwelt*.
- Apply the design method to construct an algorithm for an object transportation task by the swarm. This is a typical task that is expected to be effectively solved by the swarm.
- Evaluate the swarm behavior in the task by developing a multi-agent system (MAS) and its simulator by implementing the algorithm designed.

2 Task

Tasks that make the most of the swarm's merits are "search" and "collection". Since the swarm is composed of many agents, it can search a vast space and collect many objects in parallel. Furthermore, when the swarm progressively resolves the collecting task, the environment inevitably changes. Thanks to its flexibility, the swarm can deal with the changing environment without modifying the individuals' behavioral algorithms.

The search and collection task charges the swarm with challenging issues when the task includes problems that exceed the ability of each individual. Examples are transporting an object too heavy for one agent to move, crossing a ditch larger than one agent's size, and climbing a high level. To resolve such issues, the ability more than "mere a set of individuals" is necessary. Namely, the agents in a swarm must cooperate with each other.

The swarm has a lot of merit, albeit we cannot use the swarm for all-round tasks. Highly accurate positioning is a weak point for the swarm. The agents in the swarm have only local information and usually are not able to know their accurate position in an environment.

In this study, we address the task of search and collection to test the effectiveness of the swarm behavior algorithm. The collected object is a large mass object that cannot be transported by an individual. To resolve this challenging task, agents have to closely cooperate with each other. The concrete problems are physical connection and attraction.

For mutual connection between the agents, we use self-assembly (SA) [2][10], which is found in social insects and is utilized to overcome related issues [4]. Social insects such as ants, bees and termites show very advanced behaviors in the swarms, although individually they have only very simple intelligence. Especially, the SA as collective social insect behavior is observed in ants and bees. The SA constructs a physical structure by gathering together with a group of individuals. The following functions are achieved by the SA:

- Defense
- Pulling structures
- Thermoregulation
- Colony survival under inclement conditions
- Ease of passage when crossing an obstacle

In this study, we use the "pulling structure" to transport large mass objects. When an agent finds an object too heavy to move, if the agents can connect together and pull the object in one direction, they exert enough force to move the object.

A sufficient number of agents must gather around the large mass object to realize the pulling structure. If the agent that finds the object can attract other agents, the gathering is achievable. A pheromone trail can attract other agents. Pheromones are inducers that are secreted by a body of an organism and induce specific actions in other agents [11]. The ants generate pheromone trails between preys and a nest. The ants that perceive the pheromone are attracted in the direction of higher concentration [12]. Using a pheromone trail in swarm behavior has the following advantages:

- The agents need not memorize the precise position of objects.
- The agents can transmit information to close agents indirectly.
- When the agents communicate with each other, they have no interference due to electromagnetic wave problems.
- The swarm can continue and accelerate work by communications between individuals (this is called a group effect).

3 Design Method for Swarm Behavior

3.1 *Umwelt*

Uexküll [9] explains the behavior of an organism using the concept of *Umwelt*. The *Umwelt* is the world of an organism that is composed of a perceptual world and an effector world (see Fig. 1). These worlds interact with each other. Everything perceived by the organism forms the perceptual world. All actions of the organism compose the effector world. Each of these worlds consists of perceptual and effector cues, respectively. These signs are defined as follows.

Perceptual cue
A significant sign for the organism itself

Effector cue

A significant sign that the organism produces for others to perceive

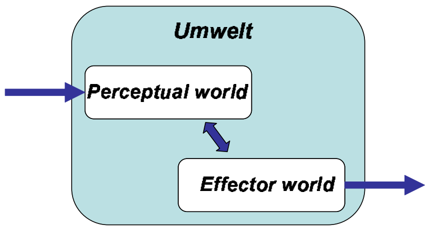

Fig. 1: *Umwelt*

Uexküll concretely explains the *Umwelt* using the example of the postnuptial actions of a female tick. The tick scrambles to the top branch of a suitable shrub using its epidermo-photoreceptor. Then, it receives the "butyric acid" stimulation that is secreted from a mammal's cutaneous glands, and "falls" from the top branch. Then, perceiving to have fallen on "something warm" because of a sharp temperature sensation, it "moves to a place without hair". When it perceives "skin" by tactile perception, it begins to "suck warm blood". In this example, the perceptual and effector cues are set out below.

Perceptual cue

P_1 Butyric acid of an animal
P_2 Warmth
P_3 Skin

Effector cue

E_1 Fall
E_2 Move to the place without body hair
E_3 Suck blood (The action of the tick continues after above actions.)

The symbols P_i $(i = 1, 2, 3)$ and E_i $(i = 1, 2, 3)$ mean a perceptual cue and an effector cue, respectively.

The remarkable point here is that the perceptual cues and the effector cues in the distinctions of the tick are just three, and they are catenated. The tick is secure in the certainty of the action by few signs. All perceptual and effector cues can be identified by tracing from the final effector cue (the purpose of the organism). We use this feature to design the algorithm.

This feature resembles a deterministic finite automaton (DFA) closely. The DFA is constructed by event, state transition, and action. The tick's DFA is shown in Fig. 2. Each state is defined as:

State

S_1 Waiting animals
S_2 Searching skin
S_3 Blood-sucking

The symbol P_i $(i = 1, 2, 3)$ is the perceptual cue. The symbol S_i $(i = 1, 2, 3)$ is the state of the tick, and E_i $(i = 1, 2, 3)$ is the effector cue in the tick's state. The tick detects the perceptual cue (P_i), and changes its internal state (S_i), and acts (E_i).

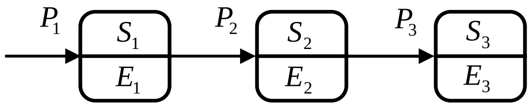

Fig. 2: State transition of tick

3.2 Design Method

From a DFA of a tick (see Fig. 2), the inner processing of the agent is set out below:

1. The agent obtains the perceptual cue from the external world.
2. The perceptual cue changes the agent's internal-state (state).
3. Until obtaining a new perceptual cue, the effector cue fixed in a state is executed.

The DFA for the swarm behavior algorithm is constructed according to the following procedures:

1. Identify all perceptual and effector cues by backtracking from the final effector cue that forms the purpose of each individual.
2. Decide internal states of the individuals from the effector cues (actions).
3. Decide the perceptual cues (stimuli) of the individuals for each state.

Moreover, the following two are thought to be indispensable features for the swarm behavior algorithm.

- Other agents exist in an agent's Umwelt
- Action depends on the other agents

As a result of these features, a very important idea for the swarm behavior algorithms is that "other agents' effector cues become one's own perceptual cues" (see Fig. 3). The algorithm for the swarm behavior has to be designed based on this idea.

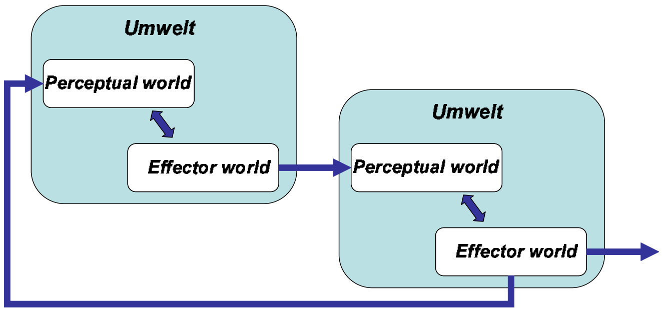

Fig. 3: Interaction of the *Umwelt* between different agents

3.3 *Umwelt* Including Self-assembly and Pheromone Trail

Using the above method, we identify all perceptual and effector cues for an agent who transports the objects in cooperation with other agents, using SA and pheromone trail. Hereafter, the object of the transportation is called the prey, and the goal of the transportation the nest.

Perceptual cue

P_1 Prey contact (0/1)
P_2 Prey movement (0/1)
P_3 Pheromone (0/1)
P_4 Nest arrival (0/1)
P_5 Contact agent state (1-9)
P_6 Passive SA (0/1)[1]
P_7 Elapse (0/1)

Effector cue

E_1 Walk randomly
E_2 Go to the nest
E_3 Grasp the prey
E_4 Make SA actively[2]
E_5 Secrete the pheromone
E_6 Trace the pheromone

In itemizing the perception cues, the number of parenthetical references is the input. In P_i $(i = 1, 2, 3, 4, 6, 7)$, 0 means non-detection, and 1 means detection. In P_5, the reference indicates the state of the contacted agents.

All agents detect only one global information, that is, the position of the nest. We think that it is reasonable assumption, since, from the viewpoint of biological plausibility, the ants (*S. invicta*) can detect the direction of the nest according to the direction of the sun [12], and, from the viewpoint of engineering feasibility, detecting the nest direction can be easily implemented using a light and light sensors [13].

To perceive the passage of time, that is, the perception sign P_7, the agent has the following three timers:

- Individual transportation timer (IT timer)
- Chemoattraction timer (CA timer)
- Self-Assembly timer (SA timer)

The IT timer counts the time elapsed since the beginning of an individual transportation. When this time reaches a certain amount without moving the prey, the agent gives up the individual transportation. The CA timer measures the time in the attracted state. This is used to avoid meaningless attraction to "old" pheromone trails. The pheromone trails remain in the field after accomplishment of the transportation. The SA timer counts the time while an agent forms a SA. The agent ceases SA, if the timer comes to certain amount. When each timer reaches a certain value, P_7 becomes 1.

3.4 State Transition Rule

The swarm behavior algorithm for the transportation of large mass objects using SA and pheromone trail is established by above 7 perceptual and 6 effector cues. We design the DFA using proposed algorithm. The algorithm by which the perceptual and the effector cues are appropriately tied is shown in Fig. 4 in a form of state transition rule.

The state transition rule is constructed using DFA. Generally, DFA is used for the design of the swarm behavior, but a design method with DFA for a swarm has not yet been established. Thus, we construct state transition rules using the above detailed method. The agent obtains the perceptual cues from the environment, the other agents and its own internal timer, and changes its state according to the algorithm. The role of each state and the effector and perceptual cues belonging to the state are set out below. P_i and E_j indicate the perceptual cue (perception) and the effector cue (action), respectively. The expression $E_i \wedge E_j$ means that both effector cues are executed at once, and $E_i \vee E_j$ either one of the two. When a value of P_i is x, the internal state change to S_k. The agent changes its own state according to the state transition rule (see Fig. 4). E_i, $P_i(x)$, and the state transition at each S_i are shown in Table 1. The function of the each state is as follows:

State	Effctor cue	Perceptual cue	Next state
S_1	E_1	$P_1(1)$	S_2
		$P_3(1)$	S_4
		$P_5(6 \vee 7)$	S_6
S_2	$E_2 \wedge E_3$	$P_2(0) \wedge P_7(1)$	S_3
		$P_4(1)$	S_1
S_3	$E_2 \wedge E_5$	$P_4(1)$	S_4
S_4	E_6	$P_1(1)$	S_5
		$P_1(0) \wedge P_7(1)$	S_1
		$P_5(6 \vee 7)$	S_6
S_5	$E_2 \wedge E_3$	$P_2(0) \wedge P_7(1)$	S_6
		$P_4(1)$	S_1
S_6	$E_2 \wedge (E_3 \vee E_4)$	$P_1(1) \wedge P_2(1)$	S_9
		$P_5(8 \vee 9)$	S_8
		$P_6(1)$	S_7
		$P_7(1)$	S_3
S_7	$E_2 \wedge (E_3 \vee E_4)$	$P_1(1) \wedge P_2(1)$	S_9
		$P_5(8 \vee 9)$	S_8
		$P_6(0)$	S_6
S_8	$E_2 \wedge E_4$	$P_5(1)$	S_1
S_9	$E_2 \wedge E_3$	$P_4(1)$	S_1

Table 1: State transition rule of each state

State

S_1 Searching (initial state)
- Walking randomly
The agent changes its state to S_2, S_4 or S_6, when it perceives $P_1(1)$, $P_3(1)$ or $P_5(6 \vee 7)$, respectively.

S_2 Individual transportation
- Carrying a prey alone[3]
The agent changes its state to S_3 or S_1, when it perceives $P_2(0) \wedge P_7(1)$ or $P_4(1)$, respectively.

S_3 Pheromone secretion
- Laying down pheromone trail between the nest and a prey
The agent changes its state to S_4 when it perceives $P_4(1)$.

S_4 Pheromone attracted
- Attracted by and tracing the pheromone
The agent changes its state to S_5, S_1 or S_6, when it perceives $P_1(1)$, $P_1(0) \wedge P_7(1)$ or $P_5(6 \vee 7)$, respectively.

S_5 Non-SA transportation
- Carrying a prey alone after attracted[4]
The agent changes its state to S_6 or S_1, when it

[1] Held by another agent
[2] Hold another agent with own arm.
[3] S_2 is the state in which the agent is not attracted and transports a prey alone.
[4] S_5 is the state in which the agent transports without SA after attracted.

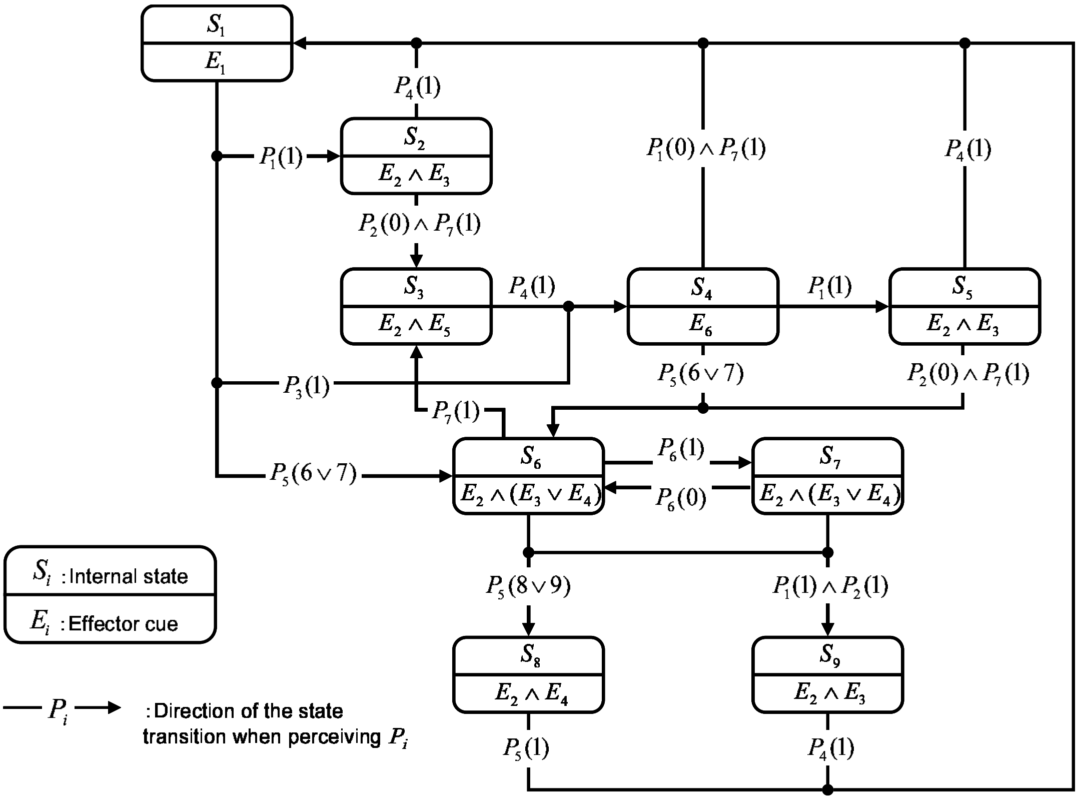

Fig. 4: State transition rule for the swarm behavior

perceives $P_2(0) \wedge P_7(1)$ or $P_4(1)$, respectively.

S_6 SA standby [End-SA]
- Waiting for SA and linking to another agent at rearmost SA chain

The agent changes its state to S_9, S_8, S_7 or S_3, when it perceives $P_1(1) \wedge P_2(1)$, $P_5(8 \vee 9)$, $P_6(1)$ or $P_7(1)$, respectively.

S_7 SA standby
- Waiting for SA and linking another agent

The agent changes its state to S_9, S_8 or S_6, when it perceives $P_1(1) \wedge P_2(1)$, $P_5(8 \vee 9)$, or $P_6(0)$, respectively.

S_8 SA transportation
- Transporting by SA chain

The agent changes its state to S_1 when it perceives $P_5(1)$.

S_9 SA transportation [Root-SA][5]
- Transporting at the root of SA chain

The agent changes its state to S_1 when it perceives $P_4(1)$.

4 Simulation of Swarm Behavior

In order to test the algorithm for swarm behavior, we conducted a simulation. In this section, we explain the details of the simulator and the task.

[5]Root-SA means that an agent forms a SA chain and grasps a prey directly.

4.1 Agent

Basic Structure and Function

The agent's body is a cylindrical shape and has upper and lower parts. The upper part is equipped with an arm to hold a prey (for E_3) or an agent (for E_4). The agent can detect the direction to the nest (for P_4). The agent has contact sensors around its body (for P_1, P_6) and has a contact communication function (for P_5). Additionally, the agent can detect whether it is moving or not (for P_2), and how much time passes (for P_7). The lower part has wheels to move (for E_1, E_2, E_6), ethanol sensors to detect pheromone (for P_3), and an ethanol dripper to leave a pheromone trail (for E_5). In E_1 (random walk), the agent normally goes straight, and the probability of change in its direction is 20%. The new direction is randomly selected between $-\pi/4$ to $\pi/4$ from the direction of movement. These two parts can rotate separately[6]. The advantage of this mechanism is that the direction of travel is not limited by the direction of the prey held. The agent's diameter is 200[mm]. The maximum speed and force are 100[mm/s] and 15[N]. The maximum action time is 3600[s]. We suppose realistic dimensions and parameters for the agent in order to be able to implement physical agents straightforwardly.

Communication

An agent directly and indirectly communicates with the other agents. The direct communication is achieved by sending the current state of oneself to another agent, when the agent contacts with the other agents. The pheromone trails serve for indirect communication.

[6]The same mechanism is adopted in SWARM-BOTS[4]

In an actual robotic system, ethanol (C_2H_5OH) is used as a substitution for pheromone. The agent lets down the ethanol tank located at the center of the bottom of the body, when the agent is in the pheromone secretion state (S_3). The ethanol is perceived as the perceptual cue of the pheromone by the other agents (S_1). The ethanol sensors are used to trace the pheromone trails. The tracing mechanism imitates that of the ants. The ants detect pheromone trails using two right-and-left antennas [14]. When the right antenna detects pheromone, the ant advances to the right, and vice versa as shown in Fig. 5. To imitate this action, two ethanol sensors are installed on the bottom of the body at $\pi/4$ from the direction of movement.

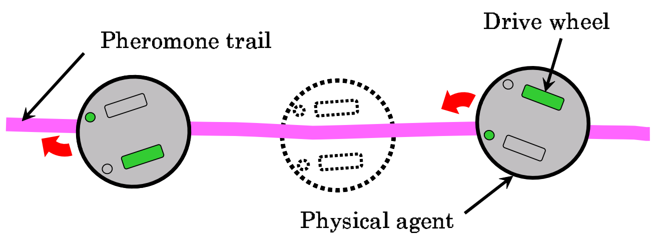

Fig. 5: Behavior on the pheromone trail

4.2 Environment

The agents move around a square field of 10000 $\times$ 10000[mm] in size. The coefficient of friction between the field and a prey is 0.5. The field is discretized with 20[mm] $\times$ 20[mm] grids to calculate the evaporation and the diffusion of pheromone. The size of the field is 10000 $\times$ 10000[mm]. There are 25000 computational grids on the field.

Pheromone on the field

The pheromone drip on the surface of the field evaporates into the atmosphere and then diffuses. The evaporation is calculated according to the following equation:

$$F_p(x,y,t) = \gamma_{vap} F_p(x,y,t-1) + \Delta F_p(x,y,t) \quad , \tag{1}$$

where $F_p(x,y,t)$ is an amount of pheromone at time t and at grid (x,y), x and y are the X-Y coordinates in the field, and γ_{vap} is the evaporation coefficient (0.99). The second term, $\Delta F_p(x,y,t)$, is the amount of the pheromone drip:

$$\Delta F_p(x,y,t) = \begin{cases} Q_p & \text{if an } S_3 \text{ agent is on the grid } (x,y), \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

Where Q_p is actual amount of pheromone drip.

Pheromone in the atmosphere

The diffusion of the pheromone is calculated by

$$\begin{aligned} A_p(x,y,t) &= A_p(x,y,t-1) \\ &\quad +\gamma_{dif}\{A_p(x+1,y,t-1) \\ &\quad +A_p(x-1,y,t-1) \\ &\quad +A_p(x,y+1,t-1) \\ &\quad +A_p(x,y-1,t-1) \\ &\quad -5A_p(x,y,t-1)\} \\ &\quad +(1-\gamma_{vap})F_p(x,y,t) \quad , \end{aligned} \tag{3}$$

where $A_p(x,y,t)$ is the amount of pheromone in the atmosphere and above grid (x,y) at time t and γ_{dif} is the diffusion coefficient (0.01). $A_p(x+1,y,t-1)$, $A_p(x-1,y,t-1)$, $A_p(x,y+1,t-1)$ and $A_p(x,y-1,t-1)$ mean inflow of the pheromone from the adjacent 4 grids. $-5A_p(x,y,t-1)$ means diffusion to the adjacent 4 grids and disappearing to the atmosphere.

4.3 Task

The task to be solved by the swarm is "search and collection of the preys". The agents transport preys that have the same size and mass and are scattered in the field, to a nest positioned at the center of the field (see Fig. 6). In this figure, the checker board lattice is the field. The small circle at the center of the field is the nest, and the larger circle is the transportation goal. The cylinders surrounding the goal circle are the preys.

The agents begin search from the nest. If they find a prey, then they try to drag it to the nest. When all preys have been transported to the nest, the task is completed.

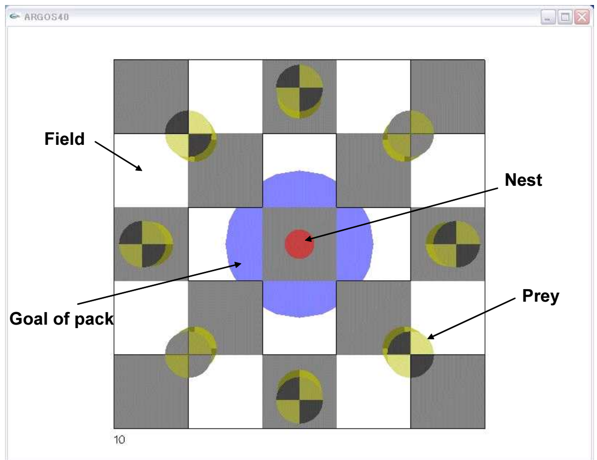

Fig. 6: Field configuration

5 Simulation Results

We undertook computational simulations of the swarm behavior algorithms drawn in Fig. 4 using the MAS simulator detailed in the previous section. In this section, we analyzed the simulation results paying attention to the following three points:

1. does the swarm solve the task or not?
2. does the parallel processing work effectively?
3. does the task allocation work effectively?

1) and 2) are proper behavior of the swarm, but 3) is adaptive behavior.

In the initial state, the preys are located on a circle around the nest with a radius of 4000[mm] as shown in Fig. 6. The agents start to move from the center of the nest.

5.1 Task Solution

Typical parameter settings showing the task solution are Table 2.

Table 2: Parameter settings for task solution

The number of agents	30
The number of preys	8
The mass of a prey	15[kg]

Actual simulations look like Fig. 7. The short lines expanded from the agents designate the directions of movement. The number at the left bottom corner of field is elapsed time from the beginning of the simulation.

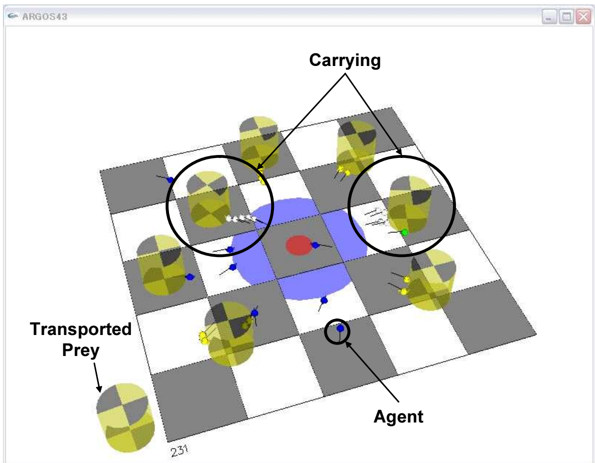

Fig. 7: Task solution

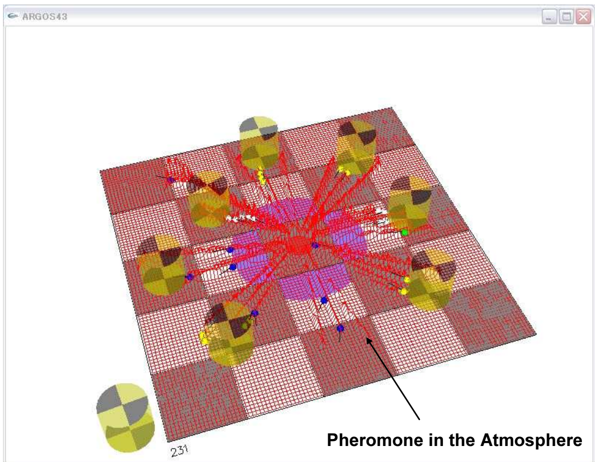

Fig. 8: Pheromone in the atmosphere

The agents generate pheromone trails (see Fig. 8), attract other agents, and repeat SA (see Fig. 7). The agents lay down the pheromone trails between the preys and the nest. The pheromone trails form straight lines since the agents know the direction of the nest.

5.2 Parallel Processing

In Fig. 7, the swarm simultaneously carried two preys (indicated by two circles), that is, the swarm demonstrated parallel processing. To estimate how the swarm solves the task in parallel, we observed the time it took to solve the task for different numbers of preys. The parameter settings are shown in Table 3.

Table 3: Parameter settings for parallel processing

The number of agents	40
The number of preys	1,2,4,6,8
The mass of a prey	20[kg]

The task solution time is shown in Fig. 9. The vertical axis is the task solution time, and the horizontal axis is the number of preys. In this figure, a linear line is plotted for comparison, which is an extrapolation of a line between 0 and the result for 1 prey. This line indicates estimated time to solve the task serially. The actual task solution time, shown by the dashed line, is always lower than the linear line. This means that the present algorithm for the swarm behavior can effectively perform parallel processing. And it means that the task can be parallelized.

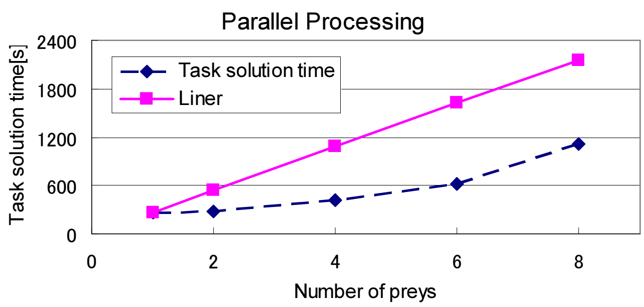

Fig. 9: Time to solve the tasks

5.3 Task Allocation

5.3.1 Task Solved

Figure 10 shows a transition of the ratio of the states in the course of time in an experiment of a solvable task. The vertical axis is the state-ratio, and the horizontal axis is time. The lines in the same region indicate the difference of the agent's state. The parameters in this experiment are summarized in Table 4. Under the present settings, one agent has the transporting capacity of 1.5[kg] at most. Coefficient of friction between the field and the prey is 0.5. Thus, the prey of 15[kg] can be carried by 5 agents.

Phase 1

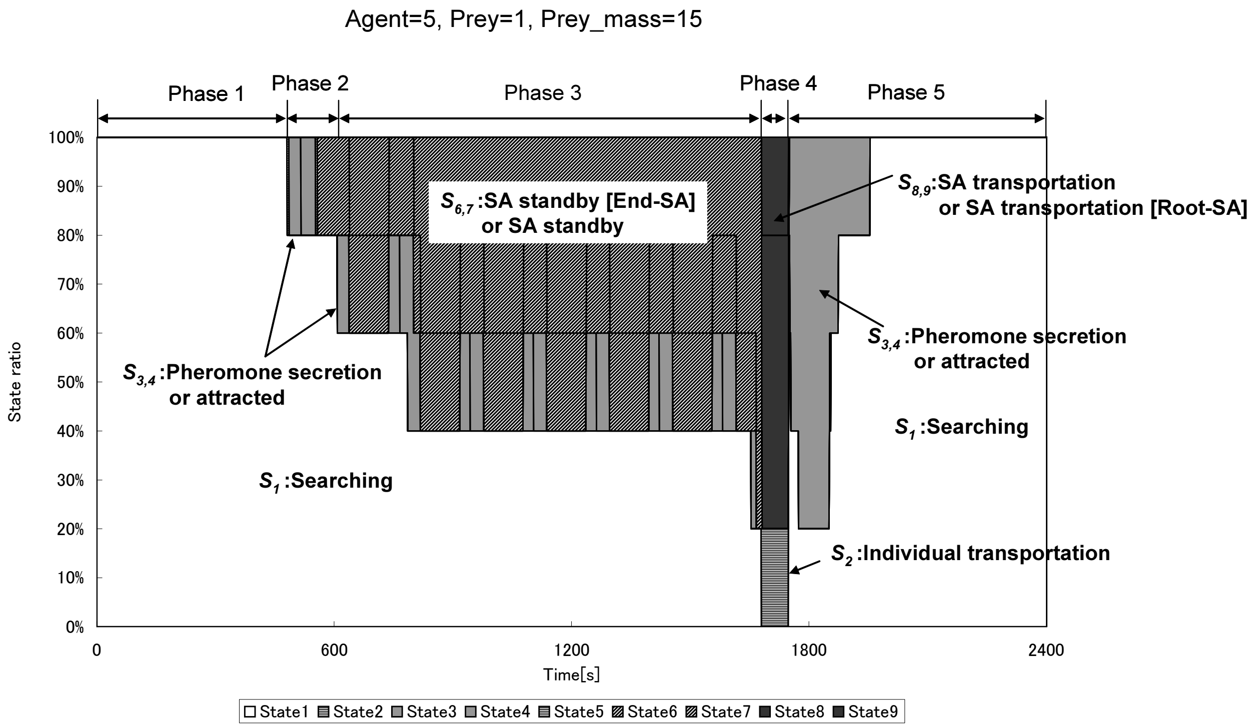

Fig. 10: The transition of the ratio of states in the process of task solving

Table 4: Parameter setting of a solvable task

The number of agents	5
The number of preys	1
The mass of a prey	15[kg]

At first, all agents searched in parallel for a prey (S_1 : Searching).

Phase 2
At 481[s] an agent found a prey, that is, received the perceptual cue $P_1(1)$ and tried to transport the prey by changing its state into S_2 (Individual transportation)[7]. But the prey could not be moved by one agent, thus, the state became S_3 (pheromone secretion), and 20% of the area in Fig. 10, that is one agent out of five, became "$S_3, 4$: Pheromone secretion or attracted". The agent shuttled between the prey and the nest for laying down a pheromone trail to attract other agents, and then turned to the S_6 (SA standby [End-SA]). At 608[s], an agent found the pheromone trail, that is, received a perceptual cue $P_3(1)$.

Phase 3
At 639[s], two agents constructed SA (S_6: SA-standby [End-SA] and S_7: SA-standby). But two agents are not enough to move the prey, the second agent changed the pheromone secretion state and began to strengthen the trail (repeated twice). At 786[s], a third agent, and at 1667[s] a fourth agent were attracted through the pheromone trail. The four agents make a SA train.

Phase 4
At 1680[s], the fifth agent contacted the prey and changed its state (S_1 to S_2). Finally, the prey was transported by five agents cooperatively, four agents forming SA and one agent non-SA. This is indicated by "$S_8, 9$: SA transportation or SA transportation [Root-SA]" and "S_2: Individual transportation" in the right side of Fig. 10.

Phase 5
After finishing the transportation, four agents were attracted by the meaningless pheromone trail. At 1955[s], the pheromone trial evaporated completely, and all agents changed their states (S_4 to S_1).

5.3.2 Task Unsolved

If a task is not solvable, the transition of the state ratio differs completely as depicted in Fig. 11. The vertical axis is the state-ratio, and the horizontal axis is time. Table 5 indicates the parameter settings for this unsolvable task experiment. The mass of a prey is now 20[kg] despite there being just the same number of agents as in the previous experiment. Therefore, this prey cannot be carried even though all five agents form SA.

Table 5: Parameter setting for an unsolvable task

The number of agents	5
The number of preys	1
The mass of a prey	20[kg]

[7]The agent's state changes from S_1 to S_2. But it doesn't appear in the graph because S_2 is only five seconds.

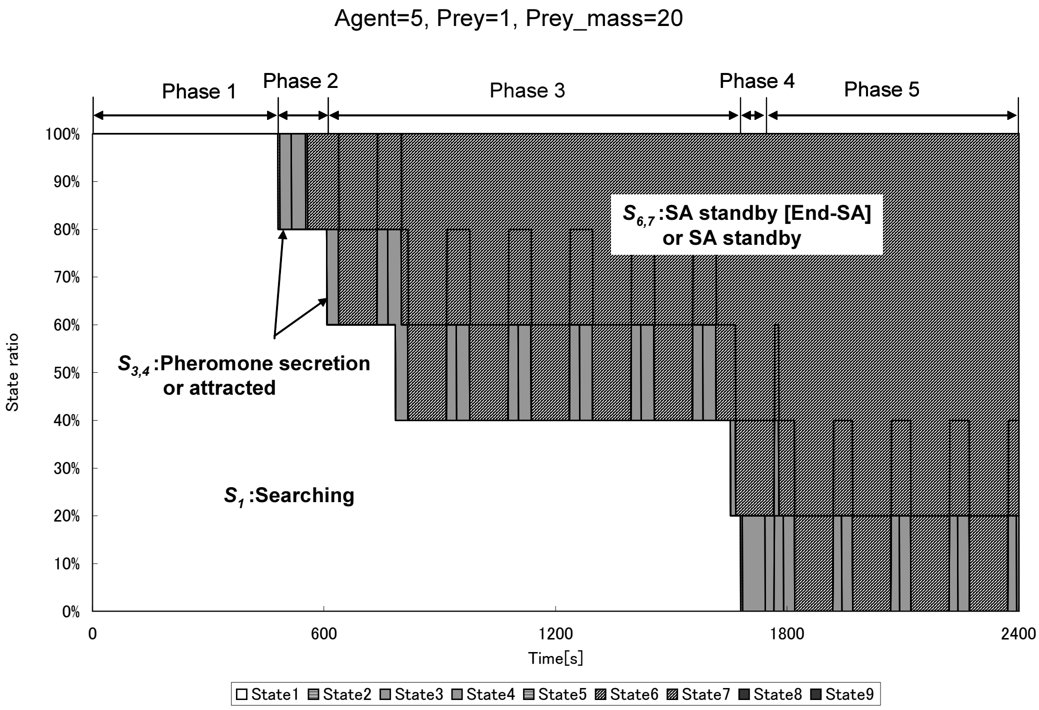

Fig. 11: Transition of the ratio of states in the process of task unsolving

Until phase 3 (1680[s]), the behavior of the swarm was the same as in Fig. 10. Namely, the swarm searched for a prey, attracted agents and formed SA. But since the prey could not be moved by the SA of 5 agents, the last agent of the SA train tried to reinforce the pheromone trail forever. Thus, the graph shows an oscillation between pheromone secretion or attracted (S_3 or S_4) and SA standby [End-SA] or SA standby (S_6 and S_7) after 1680[s].

6 Discussion

The swarm has three remarkable features: flexibility, robustness and parallelism. In this study, we showed that the proposed swarm algorithm achieves two of the three, flexibility and parallelism.

In this paper, the task is not solved only by an individual agent. Agents need to cooperate with each other, that is, make a self-assembly structure. The results shown in section 5.1 describes the overall behavior, especially about the pheromone trails and the self-assembly, which are the critical parts of our algorithm. This result validates that the agents construct the pheromone trails and the self-assembly structures appropriately. The result of parallelism shown in section 5.2 makes us confirm that parallel processing works well.

When the task is solvable, the task solving is accelerated by a lager number of agents (see Fig. 9). This observation suggests that the present swarm algorithm shows parallelism. However, the scalability of the parallelism for the large number of agents and preys is not clear. Actually, in Fig. 9, the slope of the task solution time (lower line) is nearly the same as the linear extrapolation (upper line), when there are many preys. In order to clarify the scalability, more thorough experiments with a large system ought to be conducted. If we can predict the number of agents needed to solve a task within required time, it is very useful for actual engineering problems.

The agents regulate their internal states appropriately corresponding to the surrounding circumstances (see Fig. 10). This is self-organized behavior. The circumstances consist of not only the preys but also the other agents. As the agents move and the preys are transported, the circumstances change from moment to moment. Since the agents can adapt to such dynamic situations and coordinate their roles, the task could be successfully solved. But this is not always the case. When the task is fundamentally impossible to solve, the agents cannot allocate their roles and repeat the same action in vain, over and over again (see Fig. 11).

In order to obtain the results described here, we do not need to adjust the parameters both by hand and by some adaptive algorithm such as genetic algorithm. All result are very generic. This fact suggests that the design method and the proposed swarm behavior algorithm are available and efficient.

We did not show that our algorithm has the remained feature, robustness, directly. But, it is to be expected from the results shown in Fig. 9 that the swarm can solve the task if several agents stop working, even though the solving time may be prolonged, when the overall number of agents is large enough. But, in realistic situations, the broken agents may be obstacles for the normal agents. In order to understand the effect of the broken agents, we will try to stop some agents while simulating.

7 Conclusion

The features of a swarm are well known, as it cannot be designed by a top-down approach [3]. In order to design a swarm behavior, it is necessary to build the behavior of the swarm into the design of individuals. To develop a methodology, we utilize the concept of *Umwelt* [9], which was proposed in ethology for understanding animal behavior. The *Umwelt* is an organism's own world, consisting of perceptual and effector cues, that are meaningful information and actions for the organisms.

For designing the swarm behavior, the *Umwelt* provides the following three important ideas:

- Bottom-up design: the swarm behavior is designed from the agent side.
- Back-tracking: Sequences of perceptual and effector cues can be backtracked if the individuals behavior is purposeful.
- *Umwelt* interaction: Other agents' effector cues form one's own perceptual cues.

Namely, the swarm can be defined as a group of individuals whose *Umwelts* interact with group members. The design method for swarm behavior algorithm using *Umwelt* is as follows:

1. Identify all perceptual and effector cues by back-tracking from the final effector cue that forms the purpose of each individual.
2. Decide internal states of the individuals from the effector cues (actions).
3. Decide the perceptual cues (stimuli) of the individuals for each state.

We actually designed swarm behavior achieving the search and collection of large mass objects following this procedure (see Fig. 4). The swarm uses self-assembly for mutual connection among agents, and pheromone trails to attract other agents. Using a multi-agent system simulator implementing this algorithm, we showed that the agents cooperatively transported objects from the field to the goal. In the solving processes the swarm exhibited flexibility and parallelism, which are the important features of a swarm. We discussed that it was also naturally expected for the swarm algorithm to show the other indispensable characteristic, robustness.

Thus, the effectiveness of the *Umwelt* for designing swarm behavior was confirmed to some extent. We can conclude that the design methodology of the swarm behavior based on the concept of *Umwelt* is effective. Since the adopted task, "search and collection of large mass objects" is a kind of general tasks to which swarm behavior is applied. More general applicability of the concept *Umwelt* should be ascertained. We will verify the effectiveness of *Umwelt* as a design methodology through designing actual algorithms for various tasks that the swarm should show its ability in solving. For firmly ascertaining the effectiveness, we should apply this methodology to various tasks in not only the simulated agents but also in robotic systems.

Acknowledgment

This research has been supported by Prof. Fumitoshi Matsuno and a group of researchers at the University of Electro-Communications, Tokyo. The authors have been given an important idea by Prof. Tetsuo Sawaragi at Kyoto University, Kyoto.

References

[1] C. Detrain and J-L. Deneubourg. Complexity of environment and parsimony of decision rules in insect societies. *The Biological Bulletin*, Vol. 202, No. 3, pp. 268–274, 2002.

[2] C. Anderson, G. Theraulaz, and J.-L. Deneubourg. Self-assemblages in insect societies. *Insects Sociaux*, Vol. 49, pp. 1–12, 2002.

[3] J. Oota and T. Arai. Multi-agent robot systems. *Journal of the Robotics Society of Japan*, Vol. 20, No. 5, pp. 487–490, 2002.

[4] M. Dorigo, E. Tuci, R. Groß, V. Trianni, T.H. Labella, S. Nouyan, C. Ampatzis, J.-L. Deneubourg, G. Baldassarre, S. Nolfi, F. Mondada, D. Floreano, and L.M. Gambardella. The swarm-bot project. In E. Sahin and W.M. Spears, editors, *Proc. Swarm Robotics: SAB 2004 International Workshop*, Vol. 3342, pp. 31–44. Springer Verlag, 2005.

[5] M. Drigo, G.D. Caro, and L.M. Gambardella. Ant algorithms for discrete optimization. *Artificial Life*, Vol. 5, No. 2, pp. 137–172, 1999.

[6] F. Mondada, M. Bonani, A. Guignard, S. Magnenat, C. Studer, and D. Floreano. Superlinear physical performances in a swarm-bot. In Mathieu S. Capcarrere, Alex A. Freitas, Peter J. Bentley, Colin G. Johnson, and Jon Timmis, editors, *Proc. Advances in Artificial Life: 8th European Conference*, Vol. 3630, pp. 282–291. Springer Verlag, 2005.

[7] R. O'Grady, R. Groß, F. Mondada, M. Bonani, and M. Dorigo. Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In Mathieu S. Capcarrere, Alex A. Freitas, Peter J. Bentley, Colin G. Johnson, and Jon Timmis, editors, *Proc. Advances in Artificial Life: 8th European Conference*, Vol. 3630, pp. 272–281. Springer Verlag, 2005.

[8] E. Sahin, T.H. Labella, V. Trianni, J.-L. Deneubourg, P. Rasse, D. Floreano, L.M. Gambardella, F. Mondada, S. Nolfi, and M. Dorigo. Swarm-bot: Pattern formation in a swarm of self-assembling mobile robots. In A. El Kamel, K. Mellouli, and P. Borne, editors, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. NJ: IEEE Press, 2002.

[9] Jakob von Uexküll. Streifzüge durch die umwelten von tieren und menschen. Japanese edition published 2005, 1934.

[10] B. Hölldobler and E.O. Wilson. *The ants*. The Belknap Press of Harvard University Press Cambridge, Massachusetts, 1990.

[11] P. Karlson and A. Butenandt. Pheromones (ectohormones) in insects. *Annual Review of Entomology*, Vol. 4, pp. 39–58, 1959.

[12] E. O. Wilson. *Sociobiology*. Sinsisakusya, 1999. in Japanese.

[13] R. Fujisawa, H. Imamura, T. Hashimoto, and F. Matsuno. Swarm intelligence of multi-robot using pheromone trail. In *Proc. The 2nd Internal Symposium on Mobiligence*, pp. 203–206, 2007.

[14] W. Hangartner. Spezifitat und inaktivierung des spurpheromons von lasium fulginosus latr. und orientierung der arbeiterinnen im duftfeld. *Zeitschrift für vergleichende Physiologie*, Vol. 57, pp. 103–136, 1967.