https://doi.org/10.20965/jaciii.2014.p0937



**Paper:**

# Variable Neighborhood Model for Agent Control Introducing Accessibility Relations Between Agents with Linear Temporal Logic

**Seiki Ubukata**[*1], **Tetsuya Murai**[*2], **Yasuo Kudo**[*3], **and Seiki Akama**[*4]

[*1]Department of Systems Innovation, Graduate School of Engineering Science, Osaka University
1-3 Machikaneyama-cho, Toyonaka, Osaka 560-8531, Japan
E-mail: subukata@sys.es.osaka-u.ac.jp
[*2]Graduate School of Information Science and Technologies, Hokkaido University
Kita 14, Nishi 9, Kita-Ku, Sapporo 060-0814, Japan
E-mail: murahiko@ist.hokudai.ac.jp
[*3]College of Information and Systems, Muroran Institute of Technology
27-1 Mizumoto, Muroran 050-8585, Japan
E-mail: kudo@csse.muroran-it.ac.jp
[*4]C-Republic
1-20-1, Higashi-Yurigaoka, Asao-ku, Kawasaki-shi 215-0012, Japan
E-mail: akama@jcom.home.ne.jp

[Received January 31, 2014; accepted May 26, 2014]

**In general, there are two types of agents, reflex and deliberative. The former does not have the ability for deep planning that produces higher-level actions to attain goals cooperatively, which is the ability of the latter. Can we cause reflex agents to act as though they could plan their actions? In this paper, we propose a variable neighborhood model for reflex agent control, that allows such agents to create plans in order to attain their goals. The model consists of three layers: (1) topological space, (2) agent space, and (3) linear temporal logic. Agents with their neighborhoods move in a topological space, such as a plane, and in a cellular space. Then, a binary relation between agents is generated each time from the agents' position and neighborhood. We call the pair composed of a set of agents and binary relations the agent space. In order to cause reflex agents to have the ability to attain goals superficially, we consider the local properties of the binary relation between agents. For example, if two agents have a symmetrical relation at the current time, they can struggle to maintain symmetry or they could abandon symmetry at the next time, depending on the context. Then, low-level behavior, that is, the maintenance or abandonment of the local properties of binary relations, grant reflex agents a method for selecting neighborhoods for the next time. As a result, such a sequence of low-level behavior generates seemingly higher-level actions, as though reflex agents could attain a goal with such actions. This low-level behavior is shown through simulation to generate the achievement of a given goal, such as cooperation and target pursuing.**

**Keywords:** agent control, topological spaces, neighborhood systems, linear temporal logic, local properties of binary relations

## 1. Introduction

In general, a multi-agent system aims to attain its goal through the cooperation and interactivity of its agents. According to Russell and Norvig [1], there are four types of agents:

- Simple reflex agent
- Agents that keep track of the world
- Goal-based agents
- Utility-based agents

In broad terms, we can call the first two types of agents, *reflex*, and the latter two, *deliberative*. Deliberative agents require careful, calculated planning that produces an extremely high-level action to attain goals, whereas reflex agents do not have such a powerful ability for planning. However, reflex agents have an extremely flexible ability for managing sudden environment changes, as indicated by Brooks [2]. Agents based on Brooks' subsumption architecture do not have any recollection or map, but they have been shown to respond immediately and appropriately to environmental changes.

In our previous work [3], we studied a rough-set-based agent control model. In certain simulation results of the method, we found deadlock problems that could be solved by reducing the agents' view range. From a topological perspective, such a solution suggests that agents can



© Fuji Technology Press Ltd. Creative Commons CC BY-ND: This is an Open Access article distributed under the terms of the Creative Commons Attribution-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nd/4.0/).

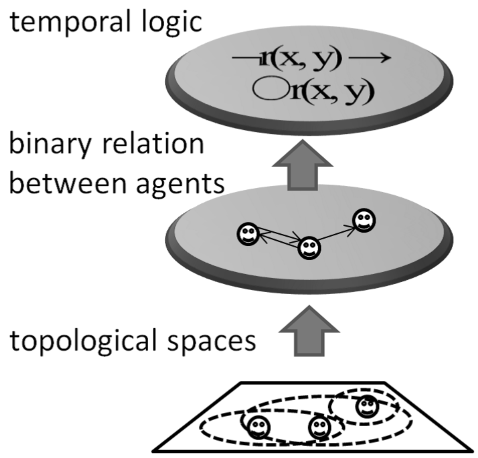


**Fig. 1.** A variable neighborhood model with three-layers.

control the size of their view range as though they could have varying neighborhoods. Thus, we proposed a variable neighborhood model in which an autonomous agent can perform an action based on the current selected neighborhood [4, 5]. However, the fact for how agents should select one neighborhood from a given neighborhood system in the topological space is not sufficiently discussed.

The agents controlled by the model are reflex agents with no ability for planning. Thus, we consider the following idea:

> Does such a method for selecting a neighborhood for the next time also give reflex agents the ability to seemingly create a plan as though they could attain their goal cooperatively?

For this purpose, we propose a *three-layered variable neighborhood model* for agent control, and the layers are as follows (**Fig. 1**):

1. Topological space
2. Agent space
3. Temporal logic

Agents with their neighborhood are in motion in a topological space, such as a plane, and in a cell space. Position and neighborhood determine the relationship between agents, and hence, we have a binary relation on the set of agents. By "agent space," we mean the pair composed of a set of agents and a binary relation on it. As agents move, their position, and thus the binary relation, changes with time. The characteristic of the model is the use of locality of the binary relation at the current time. For example, if two agents have a symmetrical relation at the current time, they struggle to maintain symmetry at the next time. Then, it is expected for the two agents to move to close positions from each other. Another example is when one agent wants to maintain symmetry, but the other agent denies symmetry; in this case, the former agent is expected to pursue the latter. Note that such properties of binary relations are not *global*, but *local*. The proposed model selects the neighborhood for agents using such local properties of relations.

This paper is organized into four sections. Section 2 provides brief mathematical preliminaries. Section 3 introduces a variable neighborhood model with three layers for reflex agent control. Section 4 shows some simulation results. Section 5 concludes the paper.

## 2. Preliminaries

### 2.1. Topological Spaces

There are many methods of introducing topology (cf. [6, 7]) into sets, such as open sets, closed sets, interior operator, closure operator, and more. Among such methods, we adopt the definition of topological space using neighborhood systems as follows:

$$\langle U, \mathscr{N} \rangle,$$

where $\mathscr{N}$ is a *neighborhood system*; that is,

$$\mathscr{N} : U \to 2^{2^U}$$

is a function that satisfies the following conditions for $x \in U$:

(N$_1$) $U \in \mathscr{N}(x)$,

(N$_2$) $X \in \mathscr{N}(x) \Rightarrow x \in X$,

(N$_3$) $X_1, X_2 \in \mathscr{N}(x) \Rightarrow X_1 \cap X_2 \in \mathscr{N}(x)$,

(N$_4$) $(X \in \mathscr{N}(x) \wedge X \subseteq Y) \Rightarrow Y \in \mathscr{N}(x)$,

(N$_5$) $X \in \mathscr{N}(x) \Rightarrow$
$\exists Y \in \mathscr{N}(x)[Y \subseteq X \wedge \forall y(y \in Y \Rightarrow X \in \mathscr{N}(y))]$.

### 2.2. Linear Temporal Logic

Temporal logic (cf. [8]) is a type of modal logic (cf. [9]). It was firstly proposed by Prior and extended by many researchers. There are many types of temporal logic and we chose a standard one for this study, that is, linear temporal logic. The language of linear temporal logic is formed by the set of atomic sentences and the propositional logical operators $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), $\rightarrow$ (material implication), and $\leftrightarrow$ (equivalence), as well as the modal operators $\bigcirc$ (NEXT), $[F]$ (GLOBALLY), $\langle F \rangle$ (IN THE FUTURE), and $U$ (UNTIL).

A model of linear temporal logic is a tuple $\langle T, \leq \rangle$, where

1. $T$ : the set of times
2. $\leq$ : a total order on $T$

In this paper, we use $\langle \mathbb{N}, \leq \rangle$, where $\mathbb{N}$ is the set of natural numbers and $\leq$ is the standard total order on $\mathbb{N}$.

$t \models p$ means that a sentence $p$ is true at a time $t$. In this paper, we mainly use the modal operator NEXT ($\bigcirc$), and its truth condition is given by

$$t \models \bigcirc p \Leftrightarrow t+1 \models p.$$

## 3. Variable Neighborhood Models with Three Layers for Reflex Agent Control

In this section, we describe a variable neighborhood model with three layers: (1) topological space, (2) agent space, and (3) linear temporal logic. Then, we explain how agents select their neighborhood at the next time using the local properties of binary relation between agents.

### 3.1. First Layer: Topological Space

Let $A = \{a_1, \ldots, a_m\}$ be a finite set of agents that move in the topological space. Typical examples of topological spaces for agent control are Euclidean and cellular spaces.

The topology of an $n$-dimensional Euclidean space $\mathbb{R}^n$ is introduced using the usual Euclidean distance defined by

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

for two points $x = (x_1, \ldots, x_n)$, $y = (y_1, \ldots, y_n) \in \mathbb{R}^n$. Every open set is a union of some $\varepsilon$-neighborhood of $x \in \mathbb{R}^n$ defined by

$$U_\varepsilon(x) = \{y \in \mathbb{R}^n \mid d(x,y) < \varepsilon\}.$$

Thus, the set of $\varepsilon$-neighborhoods is a base for a neighborhood system.

A cellular space is composed as a regular grid of cells for applications of cellular automata, life games, and so on. A two-dimensional cellular space can be identified with $\mathbb{Z}^2$, where $\mathbb{Z}$ is the set of integers with some topology generated by the Manhattan distance (for example)

$$d_{\mathrm{Mh}}(x,y) = |x_1 - y_1| + |x_2 - y_2|$$

for $x = (x_1, x_2)$, $y = (y_1, y_2) \in \mathbb{Z}^2$. In this case, 2-neighborhood is just a Neumann neighborhood as shown in **Fig. 2**. The cellular space generated from the Manhattan distance is nothing but discrete space; that is, every union of cells with the empty set is a "clopen" (open and closed) set.

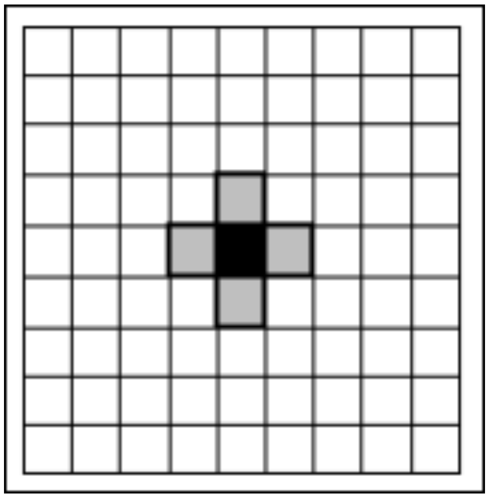

**Fig. 2.** Neumann neighborhood.

### 3.2. Second Layer: Agent Space

Every agent $a \in A$ is assumed to have its neighborhood $N(a,t)$ at each time $t \in \mathbb{N}$. We can define a binary relation between agents in terms of their position and the neighborhood they select. That is, a binary relation $R_t$ on $A$ at time $t$ is defined by

$$xR_ty \Leftrightarrow y \in N(x,t)$$

for agents $x, y \in A$. The relation describes the relationship between agents at each time. We call the pair $\langle A, R_t\rangle$ *an agent space*. This agent space is just a Kripke frame and a generalized approximation space from the perspective of modal logic and rough set theory, respectively.

### 3.3. Third Layer: Temporal Logic

In the third layer, we can describe the rules for selecting a neighborhood for an agent through temporal formulas. The binary relation in the second layer at the current time defines the truth values at the current time of atomic formulas that represent the relation between agents.

First, we formulate a language for agent control:

1. Set of individual constants: $C (= A)$ (We assume that there is a bijection between agents and their names.)
2. Predicate symbol: $r(x,y)$ ($x$ is related to $y$, which actually means that $y$ is in the neighborhood of $x$.)
3. Set of atomic formulas: $P = \{r(a_i, a_j) \mid 1 \leq i, j \leq m\}$.

Compound formulas are generated from $P$ using $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, and $\bigcirc$ with parentheses ( and ) in the usual formation rules. The language is just a sublanguage of the language of the standard propositional temporal logic language because we do not use quantifiers or other temporal operators such as $U$ (UNTIL), at least in the formulation of this paper.

Next, we explain semantics. Given a set of agents $A = \{a_1, \ldots, a_m\}$ and a family of binary relations $\{R_t\}_{t\in\mathbb{N}}$ on $A$, the truth condition for an atomic sentence $r(a_i, a_j)$ at a time $t$ is defined by

$$t \models r(a_i, a_j) \Leftrightarrow a_iR_ta_j.$$

The language allows us to represent several relationships between agents, such as

- $t \models r(x,y) \Leftrightarrow xR_ty$: $y$ is in the neighborhood of $x$ at time $t$.
- $t \models \neg r(x,y)$: $y$ is outside the neighborhood of $x$ at time $t$.
- $t \models \bigcirc r(x,y)$: $y$ is in the neighborhood of $x$ at the next time $t+1$.
- $t \models sym(x,y) = (r(x,y) \wedge r(y,x))$: $x$ and $y$ satisfy the symmetry at time $t$. ($s_{11}$)

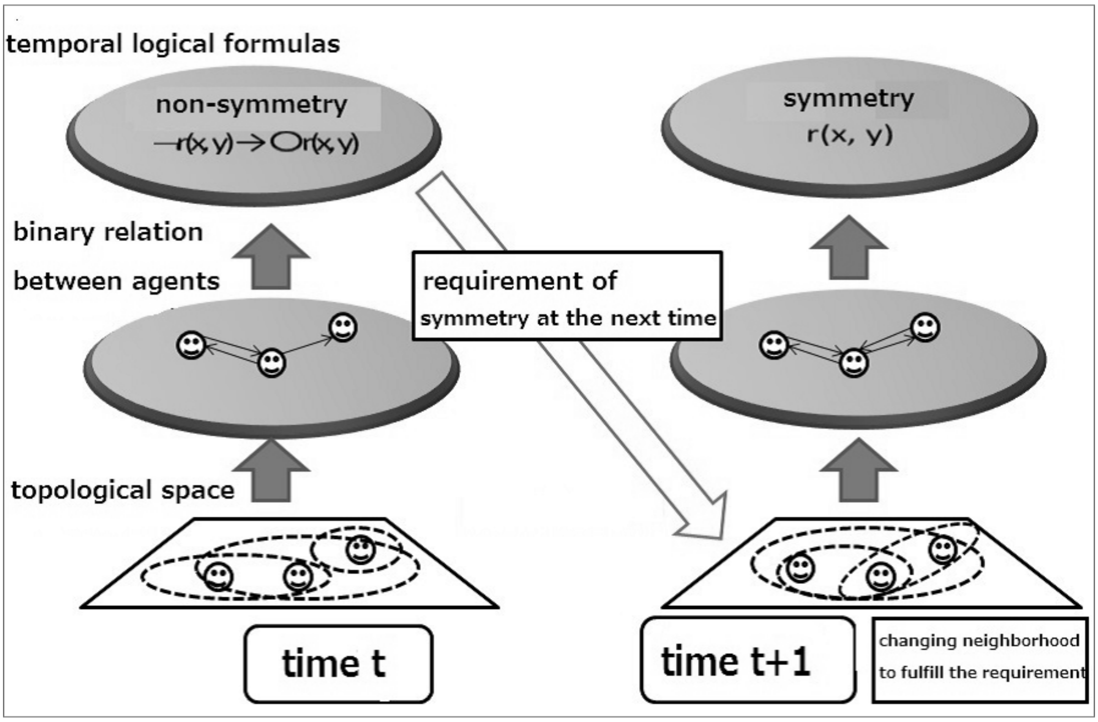


**Fig. 3.** Agent control by updating three-layered model.

- $t \models r(x,y) \wedge \neg r(y,x)$: $y$ is in the neighborhood of $x$ and $x$ is outside the neighborhood of $y$ at time $t$. ($s_{10}$)
- $t \models \neg r(x,y) \wedge r(y,x)$: $y$ is outside the neighborhood of $x$ and $x$ is in the neighborhood of $y$ at time $t$. ($s_{01}$)
- $t \models \neg r(x,y) \wedge \neg r(y,x)$: $y$ is outside the neighborhood of $x$ and vice versa at time $t$. ($s_{00}$)

Using these formulas and $\bigcirc$ (Next), we can describe several rules on the local properties of success and failure with regard to a relationship:

- $t \models r(x,y) \to \bigcirc r(x,y)$ : if $y$ is in the current neighborhood of $x$ at the current time $t$, $y$ should also be in the neighborhood of $x$ at the next time $t+1$. (positively maintaining the relation)
- $t \models \neg r(x,y) \to \bigcirc \neg r(x,y)$ : if $y$ is outside the current neighborhood of $x$ at the current time $t$, $y$ should also be outside the neighborhood of $x$ at the next time $t+1$. (negatively maintaining the relation)
- $t \models \neg r(x,y) \to \bigcirc r(x,y)$ : if $y$ is outside the current neighborhood of $x$ at the current time $t$, $y$ should be in the neighborhood of $x$ at the next time $t+1$. (positive change: obtaining a relation)
- $t \models r(x,y) \to \bigcirc \neg r(x,y)$ : if $y$ is in the current neighborhood of $x$ at the current time $t$, $y$ should be outside the neighborhood of $x$ at the next time $t+1$. (negative change: losing the relation)

Updating the proposed three-layered model using one or some of these rules along the time axis generates agent action as shown in **Fig. 3**.

### 3.4. Rules and Strategies for Selecting Neighborhoods

Next, we describe how agents select neighborhoods at the next time based on the current binary relation between agents. There are many properties of binary relations such as reflexivity, symmetry, and transitivity. By the axioms of neighborhood systems, reflexivity holds trivially for any agent. In this paper, we confine ourselves to consider local symmetry and non-symmetry.

For two agents $x$ and $y$, we have the following four states with respect to symmetry/non-symmetry:

- $s_{11}(x,y) = xRy \wedge yRx$,
- $s_{10}(x,y) = xRy \wedge \neg yRx$,
- $s_{01}(x,y) = \neg xRy \wedge yRx$,
- $s_{00}(x,y) = \neg xRy \wedge \neg yRx$

When we regard each neighborhood as a view range, we can provide the following interpretation for the four states:

- $x$ and $y$ can see each other
- $x$ can see $y$, but $y$ cannot see $x$
- $x$ cannot see $y$, but $y$ can see $x$
- $x$ cannot see $y$ and vice versa

By maintaining or abandoning such local properties of relations for the next time, we can establish the following four rules of temporal formulas with respect to local symmetry/non-symmetry in the third level of the model:

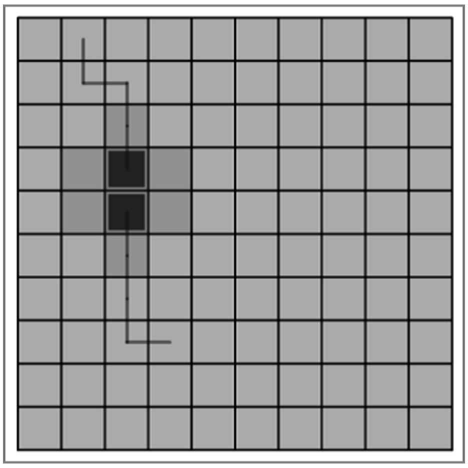

**Fig. 4.** Crowding: two agents with $(S_{11})$ and $[LN]$.

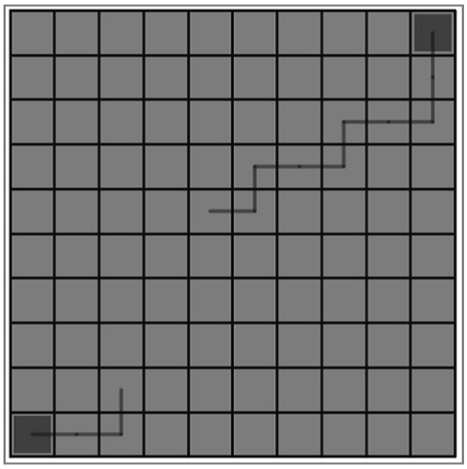

**Fig. 5.** Avoidance: two agents with $(S_{00})$ and $[GN]$.

- $(S_{11})$: $t \models \bigcirc(r(x,y) \wedge r(y,x))$
- $(S_{10})$: $t \models \bigcirc(r(x,y) \wedge \neg r(y,x))$
- $(S_{01})$: $t \models \bigcirc(\neg r(x,y) \wedge r(y,x))$
- $(S_{00})$: $t \models \bigcirc(\neg r(x,y) \wedge \neg r(y,x))$

Each rule $(S_{ij})$ satisfies the corresponding state $s_{ij}(x,y)$ at the next time. Unfortunately, the rules described above provide soft constraints only for selecting neighborhoods, but do not provide us with unique solutions. Thus, we need some strategies to select a neighborhood from the candidates generated by the rules. Such strategies depend on the type of action we want to generate. For example, we can consider the following three strategies:

- $[GN]$: select the greatest neighborhood so that an agent can satisfy the given rule.
- $[LN]$: select the least neighborhood so that an agent can satisfy the given rule.
- $[FN]$: do not change the size of the neighborhood.

## 4. Simulation Results

Low-level agent behavior, that is, maintaining or abandoning the local properties of binary relations, provides reflex agents with a method of select neighborhoods for the next time. As a result, such a sequence of low-level behavior might generate seemingly higher-level actions, as though such agent could attain a goal with said actions. In this section, we show some simulation results that focus on the local symmetry/non-symmetry of binary relations. We use ActionScript 3.0 for the simulation.

### 4.1. Cellular Spaces

First, we create some simulations of two agents. We assume a $10 \times 10$ two-dimensional cellular space. The squares on some cells are agents, and the regions around them are their neighborhood. Lines crossing the cells are the agents' movement histories.

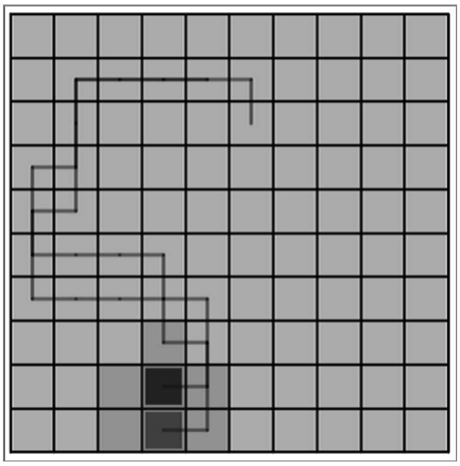

**Fig. 6.** Pursuing: one agent with $(S_{11})$ and $[LN]$ (dark gray color) and another with $(S_{00})$ and $[GN]$ (light gray color).

1. Two agents with the rule $(S_{11})$ and the strategy $[LN]$ (**Fig. 4**): we can find that the agents repeatedly move to positions close to each other, and finally they adjoin and stop. Thus, by this rule and strategy, we can realize a higher level action of crowding.
2. Two agents with the rule $(S_{00})$ and the strategy $[GN]$ (**Fig. 5**): the agents move away from each other, which means an action of avoidance.
3. One agent with the rule $(S_{11})$ and the strategy $[LN]$ and another one with the rule $(S_{00})$ and the strategy $[GN]$ (**Fig. 6**): we can find an action of pursuing.

Next, we simulate four agent cases, where three agents are with the rule $S_{11}$ and the strategy $[LN]$, and the last agent is with the rule $S_{00}$ and the strategy $[GN]$. Then, we can find that three agents cooperatively pursue the last agent, as shown in **Figs. 7** and **8**. Thus, the three agents act

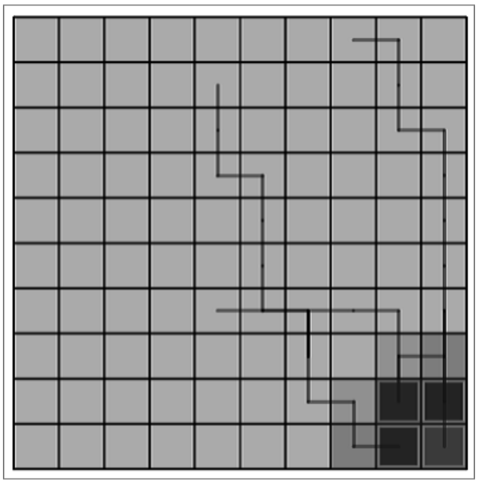

**Fig. 7.** Cooperative pursuing: three hunters and one prey (1).

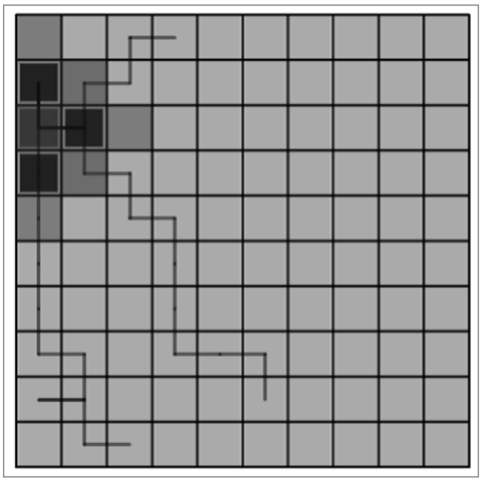

**Fig. 8.** Cooperative pursuing: three hunters and one prey (2).

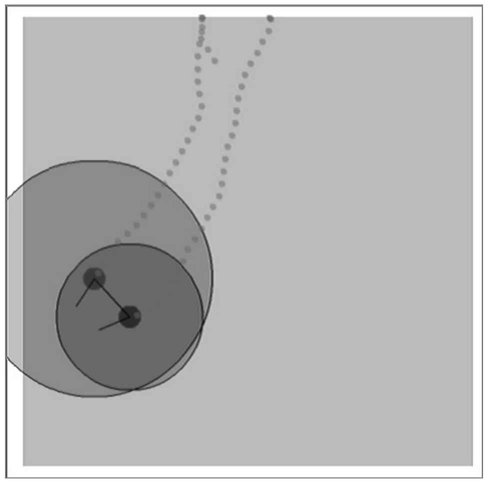

**Fig. 9.** Parallel running: two agents with $(S_{11})$ and $[LN]$.

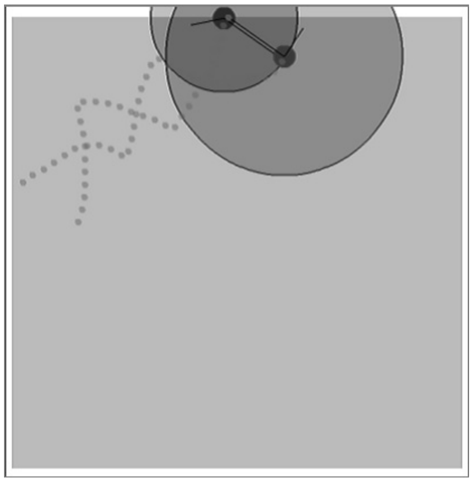

**Fig. 10.** Twisting: two agents with $(S_{11})$ and $[LN]$.

as hunters and the last agent a prey so that target pursuing cases are generated.

### 4.2. Plane: Two-Dimensional Euclidean Spaces

We created similar simulations for agents moving on a plane. Small circles with a direction line are agents, and the light gray circles around them are their neighborhood. Point sequences are the agents' movement histories.

First, we simulate two agent cases:

1. Two agents with the rule $(S_{11})$ and the strategy $[LN]$: we can find parallel running (**Fig. 9**) and twisting (**Fig. 10**).
2. Two agents with the rule $(S_{01})$ and the strategy $[LN]$ (**Fig. 11**): we can find escaping and pursuing; and their reversal.
3. Two agents with the rule $(S_{00})$ and the strategy $[GN]$ (**Fig. 12**): we can find avoidance.

Next, we show the simulation results of eight agents cases.

1. Eight agents with the rule $(S_{00})$ and the strategy $[GN]$ (**Fig. 13**): all avoid each other, and therefore, they spread uniformuly on the plane.
2. Eight agents with the rule $(S_{11})$ and the strategy $[LN]$ (**Fig. 14**): in general, they tend to crowd each other. In **Fig. 15**, we can find that the agents form some clusters.
3. Eight agents with the rule $(S_{10})$ and the strategy $[LN]$ (**Fig. 16**): we can find parallel running, and that the agents tend to crowd and not avoid each other greatly.
4. Four agents with the rule $(S_{11})$ and the strategy $[LN]$ and four other agents with the rule $(S_{00})$ and the strategy $[GN]$ (**Fig. 17**): we can find that agents with the rule $(S_{11})$ pursue agents with the rule $(S_{00})$.
5. Two agents with the rule $(S_{11})$ and the strategy $[LN]$; two with $(S_{10})$ and $[LN]$; two with $(S_{01})$ and $[LN]$; and two with $(S_{00})$ and $[GN]$ (**Fig. 18**): we can find

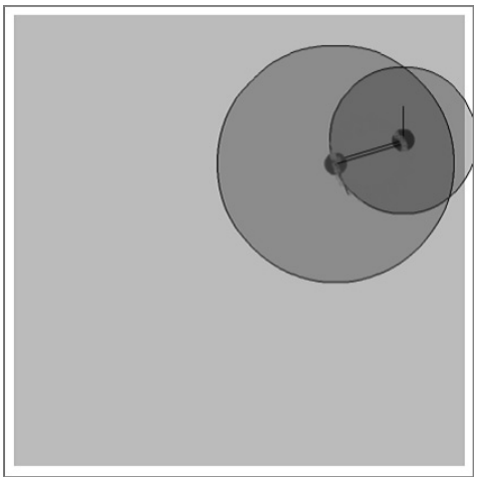

**Fig. 11.** Escaping and pursuing; and their reversal: two agents with $(S_{10})$ and $[LN]$.

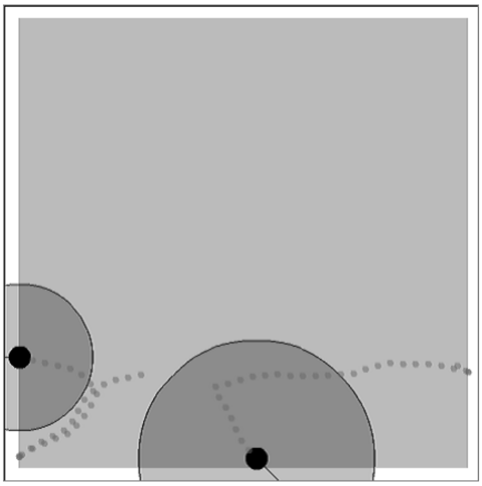

**Fig. 12.** Avoidance: two agents with $(S_{00})$ and $[GN]$.

all the actions described above appear in this simulation.

## 5. Concluding Remarks

In this paper, we proposed a three-layered variable neighborhood model for reflex agent control. The model consists of three layers: (1) topological space, (2) agent space, and (3) linear temporal logic. The characteristic of the model is the use of local properties of binary relations between agents in order to determine agent position and neighborhoods at the next time. Such low-level behavior, that is, the maintenance or abandonment of the local properties of binary relations, was suggested to generate seemingly higher-level actions as though the agents could attain a goal with such actions. In fact, through our simulation, we verified higher-level actions such as crowding, forming clusters, avoidance, pursuing, and escaping generated from lower-level agent behavior maintaining or abandoning local properties of binary relations.

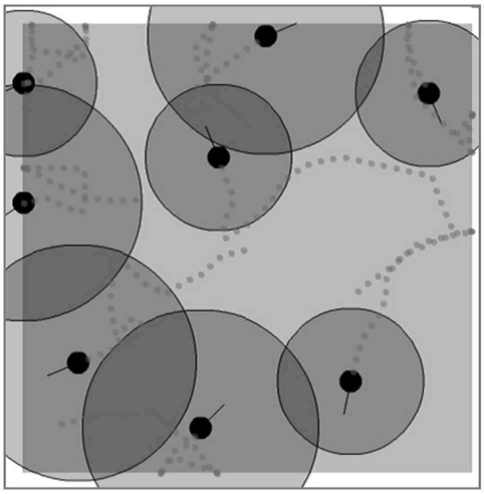

**Fig. 13.** Avoidance: eight agents with $(S_{00})$ and $[GN]$.

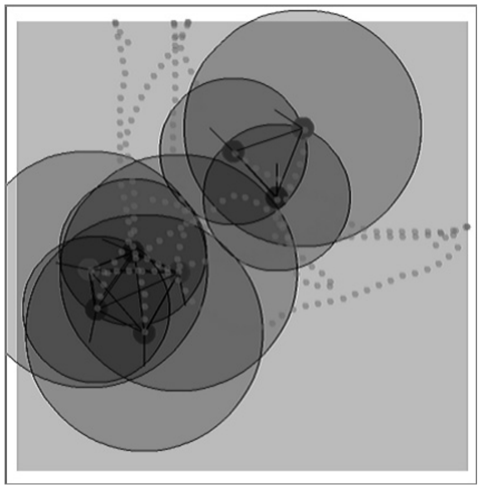

**Fig. 14.** Crowding: eight agents with $(S_{11})$ and $[LN]$ (1).

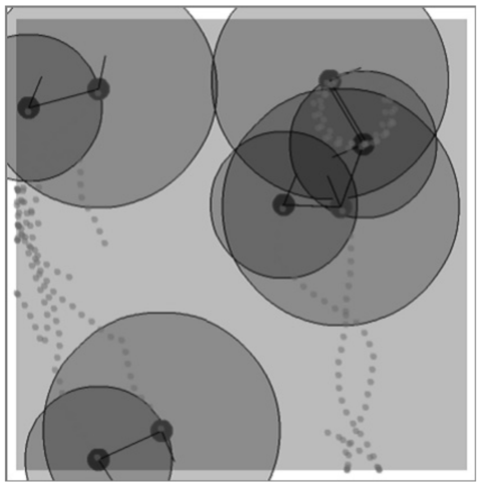

**Fig. 15.** Clusters: eight agents with $(S_{11})$ and $[LN]$ (2).

For future tasks, we plan to investigate the following:

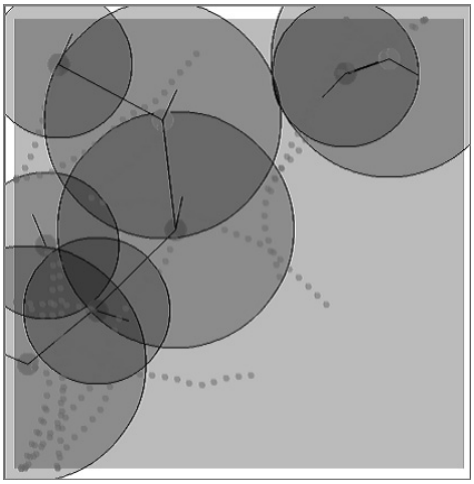

**Fig. 16.** Parallel running: eight agents with $(S_{10})$ and $[LN]$

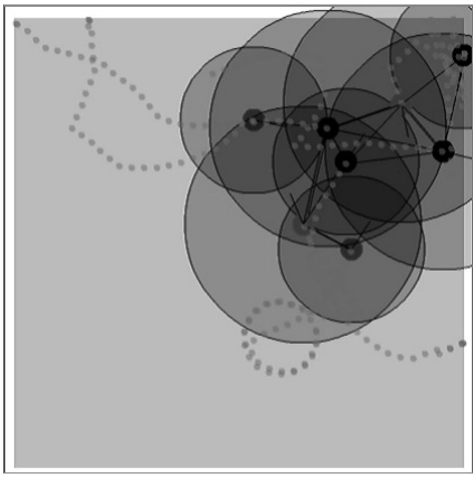

**Fig. 17.** Chasing: four agents with $(S_{11})$ and $[LN]$ and four agents with $(S_{00})$ and $[GN]$.

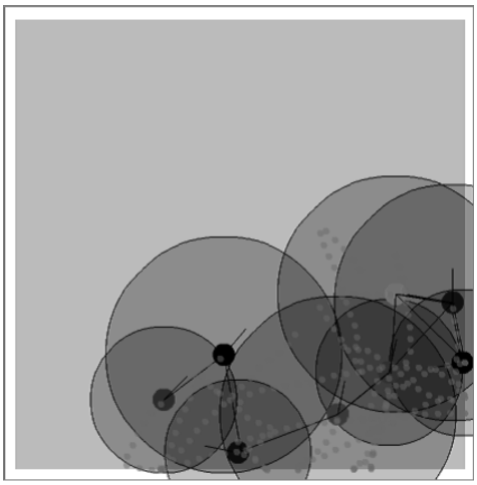

**Fig. 18.** Many actions: two agents with $(S_{11})$ and $[LN]$; two with $(S_{10})$ and $[LN]$; two with $(S_{01})$ and $[LN]$; and two with $(S_{00})$ and $[GN]$.

1. Agent control in various types of topological spaces.
2. Introduction of a granularity property to agents. In particular, the interaction of agents with different degrees of granularity.
3. Effect of other properties of binary relations to agent control.
4. Resolution of rule conflicts and inconsistency in the temporal logical setting.

**Acknowledgements**

This work was partially supported by JSPS Grant-in-Aid for Exploratory Research 24650099.

**References:**

[1] S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach," Prentice Hall, 1995.
[2] R. A. Brooks, "Intelligence Without Representation," Artificial Intelligence, Vol.47, pp. 139-159, 1991.
[3] S. Ubukata, Y. Kudo, and T. Murai, "An Agent Control Method Based on Rough-set-based Granularity," Soft Computing and Intelligent Systems and 9th Int. Symp. on Advanced Intelligent Systems (SCIS&ISIS 2008) CD-ROM, FR-G3-1, 2008.
[4] S. Ubukata, Y. Kudo, and T. Murai, "An Agent Control Method Based on Variable Neighborhoods," Knowledge-Based and Intelligent Information and Engineering Systems, LNAI 5712, pp. 356-363, Springer, Berlin, 2009.
[5] S. Ubukata, Y. Kudo, and T. Murai, "Autonomous Agent Control Based on Variable Neighborhoods," Int. J. of Reasoning-based Intelligent Systems (IJRIS), Vol.3, No.1, 2010.
[6] W. Sierpiński, "General Topology," Univ. of Toronto Press, 1956.
[7] J. R. Munkres, "Topology," 2nd edition, Prentice Hall, 2000.
[8] R. Goldblatt, "Logics of Time and Computation," CSLI, 1987.
[9] B. F. Chellas, "Modal Logic: an Introduction," Cambridge University Press, Cambridge, 1980.

**Name:**
Seiki Ubukata

**Affiliation:**
Assistant Professor, Department of Systems Innovation, Graduate School of Engineering Science, Osaka University

**Address:**
1-3 Machikaneyama-cho, Toyonaka, Osaka 560-8531, Japan


**Brief Biographical History:**
2009 Received the Master degree of Information Science and Technology from Hokkaido University
2009-2014 Doctor Course Student, Devision of Computer Science, Hokkaido University
2014- Assistant Professor, Osaka University

**Main Works:**
• S. Ubukata, Y. Kudo, and T. Murai, "Autonomous Agent Control Based on Variable Neighborhoods," Int. J. of Reasoning-based Intelligent Systems (IJRIS), Vol.3, No.1, 2010.

**Membership in Academic Societies:**
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japan Society of Kansei Engineering (JSKE)


**Name:**
Tetsuya Murai

**Affiliation:**
Associate Professor, Graduate School of Information Science and Technology, Hokkaido University

**Address:**
Kita 14, Nishi 9, Kita-ku, Sapporo 060-0814, Japan


**Brief Biographical History:**
1985 Received M.E. degree in Information Engineering from Hokkaido University
1987-1992 Lecturer, School of Allied Health Professions, Sapporo Medical College
1992-1995 Associate Professor, Hokkaido University of Education
1994 Received D.E. degree from Hokkaido University
1995- Associate Professor, Hokkaido University

**Main Works:**
• T. Murai, S. Miyamoto, M. Inuiguchi, Y. Kudo, and S. Akama, "Fuzzy Multisets in Granular Hierarchical Structures Generated from Free Monoids," MDAI 2013, pp. 248-259, 2013
• T. Murai, S. Miyamoto, M. Inuiguchi, and S. Akama, "Granular Hierarchical Structures of Finite Naïve Subsets and Multisets," Int. J. Reasoning-based Intelligent Systems, Vol.4, No.3, pp. 118-128, 2012.

**Membership in Academic Societies:**
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• Japan Society of Kansei Engineering (JSKE)


**Name:**
Yasuo Kudo

**Affiliation:**
Associate Professor, College of Information and Systems, Graduate School of Engineering, Muroran Institute of Technology

**Address:**
27-1 Mizumoto, Muroran, Hokkaido 050-8585, Japan


**Brief Biographical History:**
1997 Received M.E. degree in Systems and Information Engineering from Hokkaido University
2000 Received D.E. degree in Systems and Information Engineering from Hokkaido University
2000-2003 Post Doctoral Fellow, Muroran Institute of Technology
2003-2007 Research Associate, Muroran Institute of Technology
2007-2010 Assistant Professor, Muroran Institute of Technology
2010- Associate Professor, Muroran Institute of Technology

**Main Works:**
• Y. Kudo, T. Murai, and S. Akama, "A Granularity-Based Framework of Deduction, Induction and Abduction," Int. J. of Approximate Reasoning, Vol.50, Issue 8, pp. 1215-1226, 2009.

**Membership in Academic Societies:**
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)
• Japan Society of Kansei Engineering (JSKE)


**Name:**
Seiki Akama

**Affiliation:**
C-Republic

**Address:**
1-20-1, Higashi-Yurigaoka, Asao-ku, Kawasaki-shi 215-0012, Japan


**Brief Biographical History:**
1984 B.S., Science University of Tokyo
1984-1993 Engineer, Fujitsu Ltd.
1990 Ph.D., Keio University
1993-2006 Lecturer, Teikyo University of Technology (Teikyo Heisei University)

**Main Works:**
• "Annotated Logics $Q\tau$ and Ultraproducts," Logique et Analyse, Vol.160, pp. 335-343, 1997 (Published in 2000).
• "On Prior's Three-Valued Logic $Q$," IEEE Proc. 35th Int. Symp. on Multiple-Valued Logic (ISMVL 2005), pp. 14-19, 2005.
• "Constructive Logic and Situation Theory," Advances in Logic Based Intelligent Systems, IOS Press, Vol.132, pp. 1-8, 2005.

**Membership in Academic Societies:**
• Association of Symbolic Logic (ASL)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)