Robotics and Autonomous Systems 61 (2013) 1036–1048

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Control of swarm behavior in crossing pedestrians based on temporal/spatial frequencies

Ko Yamamoto [a,*], Masafumi Okada [b]

[a] Nagoya University, Furo-Cho, Chikusa-Ku, 464-8603 Nagoya, Japan

[b] Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, 152-8552 Tokyo, Japan

HIGHLIGHTS

- Continuum model of crossing pedestrian flows is proposed.
- Based on the model, we quantify the congestion degree in the crossing flows.
- Implicit control method of swarm behavior by guide motion is also proposed.
- Control algorithm is derived from the analysis on the temporal/spatial frequencies.

ARTICLE INFO

Article history:
Received 9 October 2012
Received in revised form
28 May 2013
Accepted 1 June 2013
Available online 19 June 2013

Keywords:
Crossing pedestrian flows
Control of swarm behavior
Temporal/spatial frequencies
Nonlinear oscillation

ABSTRACT

In the densely-populated urban areas, pedestrian flows often cross each other and congestion is caused. The congestion makes us feel uncomfortable and sometimes leads to pedestrian accidents. To reduce the congestion or the risk of accidents, it is required to control the swarm behavior of pedestrian flows. This paper proposes modeling and controlling method of the crossing pedestrian flows. In the social/urban engineering, it is well known that the swarm behavior with a diagonal stripe pattern emerges in the crossing area of the flows. This is a self-organized phenomenon caused by the local collision avoidance effect of the pedestrians. To control the macroscopic behavior of the flows, we utilize this self-organized phenomenon. Firstly, we propose the continuum model of the crossing pedestrian flows. In the continuum model, the dynamic change of the congestion in the diagonal stripe pattern is simulated as the density. Secondly, the novel control method to improve average flow velocity is proposed based on the model. The proposed method utilizes the dynamic interaction between the diagonal stripe pattern and *guides*, who are moving in the flows. The authors derive the control algorithm through an analysis on the temporal and spatial frequencies of the crossing flows. The validity is verified with simulations using the continuum model. Moreover, we apply the proposed method to the particle model, assuming the actual pedestrians.

1. Introduction

Congestion is often caused in the densely-populated urban areas, e.g., station, airport, diagonal crossing, museum, or event sites. One reason of the congestion is that pedestrian flows cross each other. For example, pedestrian flows with different destinations often cross around the ticket gate in the station, as shown in Fig. 1(a). The congestion makes us feel uncomfortable and leads to accidents. To reduce the congestion and make pedestrian flows smoother, two techniques are required: modeling and control algorithm of pedestrian flows. There have been a lot of studies on modeling of pedestrian behaviors, for application to urban planning or animation of crowds. In general, the pedestrian behavior in a crowd is considered to be divided into two types. One is the local interaction between pedestrians such as collision avoidance, which is regarded as the microscopic behavior [1–3]. The other is global route choice or trajectory planning, which is regarded as the macroscopic behavior [4–6]. Concerning the microscopic behavior, Helbing and Molnár [1] proposed the *Social Force Model*, in which the pedestrian movement is represented by a motion equation considering attracting force to a goal and collision avoidance force. Berg et al. [2] and Karamouzas [3] proposed modeling methods of the collision avoidance behavior with prediction. In addition to the local collision avoidance, Pettre et al. [4] and Guy et al. [6] proposed global trajectory planning methods. Pelechano [5] simulated the pedestrian behavior in the high-densely situation by combining psychological and geometrical rules with the collision avoidance effect. While these are multi-agent models, there have been the cellular automata model [7,8] or continuum dynamics model [9–11] of the pedestrian behavior. Whereas the multi-agent model is suitable for realistic simulation, the continuum model is suitable for congestion analysis because the congestion degree is calculated as the density.

* Corresponding author.
E-mail addresses: ko.yamamoto@esi.nagoya-u.ac.jp, yamamoto@micro.mep.titech.ac.jp (K. Yamamoto).

0921-8890/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.robot.2013.06.001

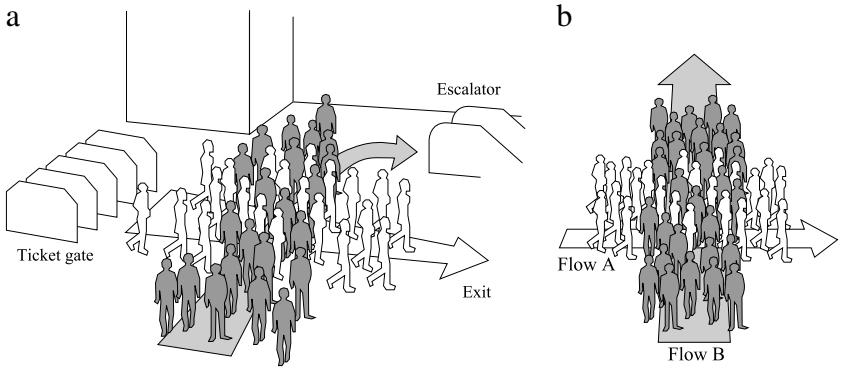


Fig. 1. (a) Congestion in a station and (b) crossing pedestrian flows. When two pedestrian flows cross vertically, a diagonal stripe pattern emerges.

In those studies, the swarm behavior in pedestrian flows is often focused on, which is the self-organized phenomenon due to the local interaction between pedestrians. For example, it is known that a phenomenon called *lane formation* emerges when two flows cross in opposite directions. This phenomenon has been simulated by a lot of studies [1,5,11–13]. Moreover, it is also known that a diagonal stripe pattern emerges and propagates in the crossing area when two flows cross vertically [14], as shown in Fig. 1(b). This dynamic phenomenon in the crossing pedestrian flows has been simulated by [3,8,13].

Concerning the navigation of pedestrians, methods to give commands to each pedestrian have been proposed [15–17]. However, it is difficult to apply this approach to situations with uncertainties such that there are unspecified number of people, it is unknown how many people have a device to receive the navigation commands, or it is not guaranteed all of them follows the commands. Although Narumi et al. [18] proposed a guidance system with a moving image on the wall, their method is limited to *direct* guidance to pedestrians. On the other hand, in research fields of multi-robotics or multi-agent system, the *Shepherding System* [19,20] was proposed, in which a large number of agents are guided to a goal by a limited number of guide robots or agents. Based on a similar concept, Okada et al. [21] proposed an evacuation guidance method by a limited number of guides. They modeled the macroscopic behavior of the evacuee with a velocity vector field and calculated the optimal position of the guides to control the macroscopic behavior. Okada et al. [22] proposed a congestion reducing method in a exhibition hall with the optimal location of partitions. These approaches are regarded as *indirect* control of swarm, dealing with unspecified number. However, they focus on a single flow with the steady congestion, and it is difficult to apply these approaches to the crossing pedestrian flows because the congestion changes dynamically. To control the crossing flows, it is important to utilize its swarm behavior which emerges in the crossing area.

In this paper, we propose modeling and control method of the crossing pedestrian flows. In particular, a novel approach to control the swarm behavior by using a limited number of *guide robots* is proposed. When guide robots move in the flows, as shown in Fig. 2, it is expected that pedestrians try to avoid collisions with the robots. This leads to the dynamic interaction between guide robots and pedestrians. We control the swarm behavior by utilizing this dynamic interaction, without explicit guidance. Hereafter, we call this method *implicit control of swarm behavior*. This is a novel approach integrating robotics and social engineering.

This paper is organized as follows. In Section 2, we present the particle model of the crossing pedestrian flows. In this model, the macroscopic behavior of pedestrian flows is represented by a vector field in a similar way to [21,22]. In order to quantify the dynamic change of the congestion, the continuum model of the crossing flows is presented in Section 3. In Section 4, we explain the proposed control method of the swarm behavior. Focusing on the cyclic phenomenon of the diagonal stripe pattern, we move guide robots in a cycle and control the flows by utilizing the interaction between the guide robots and the flows. The control algorithm to improve the average flow velocity is derived from an analysis on temporal and spatial frequencies of the crossing flows. Then, we address the crossing flows control under different conditions in Section 5. In Section 6, we apply the proposed control method to the particle model. Finally, we summarize and conclude this research in Section 7.

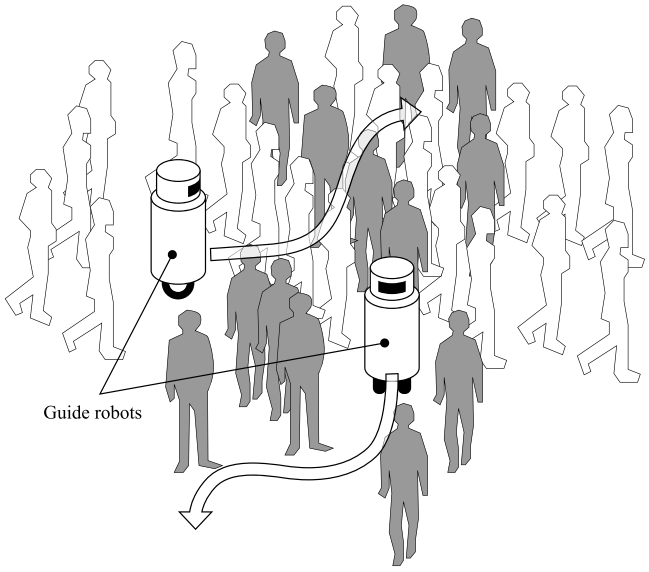


Fig. 2. Implicit control of swarm behavior by guide robots.

This paper is an extension of our previous paper [23]. In addition to the contents of [23], parameters of the particle model are identified from measurement data of the actual pedestrians in Section 2. Moreover, the effectiveness of the control algorithm is verified by using the obtained parameters and simulating different situations in Section 5.

2. Particle model of crossing pedestrian flows

2.1. Modeling of macroscopic behavior in pedestrian flow

We consider pedestrian movement in the two-dimensional space. In a similar way to [21,22], the macroscopic behavior of a

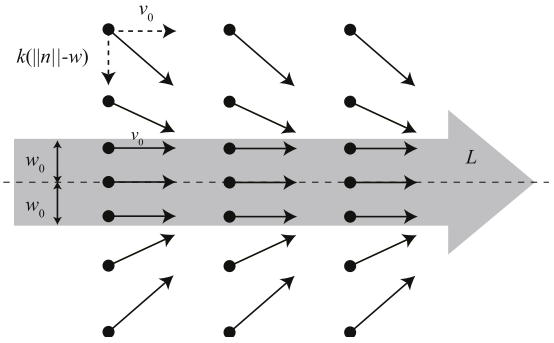


Fig. 3. Velocity vector field representing the macroscopic behavior in a line-shaped pedestrian flow.

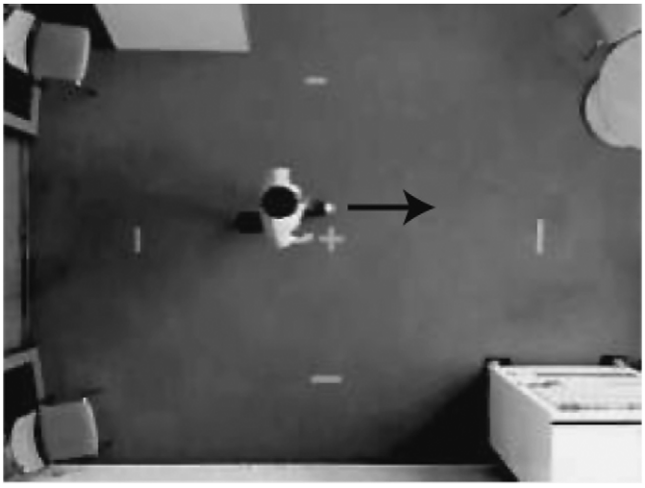


Fig. 4. Measurement of the walking trajectory of a pedestrian.

pedestrian flow is modeled with a velocity vector field: velocity at position $\boldsymbol{x}$ is given by the following vector field $\boldsymbol{f}(\boldsymbol{x})$:

$$\boldsymbol{v} = \boldsymbol{f}(\boldsymbol{x}) \tag{1}$$

$$\boldsymbol{x} = \begin{bmatrix} x & y \end{bmatrix}^T. \tag{2}$$

Focusing on a line-shaped pedestrian flow as shown in Fig. 3, $\boldsymbol{f}(\boldsymbol{x})$ is designed as follows:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{cases} v_0 \boldsymbol{d} & (\|\boldsymbol{n}\| \leq w_0) \\ v_0 \boldsymbol{d} + k(\|\boldsymbol{n}\| - w_0) \dfrac{\boldsymbol{n}}{\|\boldsymbol{n}\|} & (\|\boldsymbol{n}\| > w_0) \end{cases} \tag{3}$$

where $\boldsymbol{d}$ is the unit directional vector of the line, and $\boldsymbol{n}$ is the normal vector from the position $\boldsymbol{x}$ to the line. w_0 and v_0 denote the width of the pedestrian flow and the reference velocity in the flow region, respectively. k is a magnitude of attracting effect to the flow.

2.2. Modeling of microscopic behavior in pedestrian flow

In this section, we verify the diagonal stripe pattern formation in the crossing flows by modeling each pedestrian with a particle. We call this model *particle model*. Suppose that there are two pedestrian flows along x- and y-axes. Let $\boldsymbol{f}_A$ and $\boldsymbol{f}_B$ denote the vector fields of each flows. The velocity of a particle i in flow A is given as follows:

$$\boldsymbol{v}_i = \boldsymbol{f}_A(\boldsymbol{x}_i) - \sum_{i \neq j} s(\|\boldsymbol{r}_{ij}\|) \frac{\boldsymbol{r}_{ij}}{\|\boldsymbol{r}_{ij}\|} \tag{4}$$

$$\boldsymbol{r}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i \tag{5}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{v}_i$ are position and velocity of the particle i, respectively. This model is similar to the Social Force Model [1]. The first term on the right-hand side of (4) represents the attractive effect of the destination, and the second term represents a repulsive effect of the nearby particles (the particle i keeps a distance from nearby particles). $\boldsymbol{r}_{ij}$ is the relative position vector from the particle i to j. The repulsive effect is modeled with $s(r)$, a sigmoid function defined as follows:

$$s(r) = \frac{c}{1 + \exp\{a(r - b)\}} \tag{6}$$

where a, b and c are constant values. This function represents *Personal Space* [24], where b indicates the radius of the personal space, and a is a parameter to smooth the boundary between the personal space and the other. c is a parameter to determine the magnitude of the repulsive effect, which is a scaling factor between the whole setting area and the personal space.

The velocity of particles in flow B is given in a similar way. Note that this is velocity-level modeling, whereas the Social Force Model is an acceleration-level model utilizing the motion equation. Considering pedestrian movement with general temporal and spatial scales, we can assume that each pedestrian velocity reaches usual walking speed instantaneously. Therefore, we consider the velocity-level modeling is valid.

2.3. Parameter identification of particle model

We identify the parameters of the repulsive effect given by (6) by measuring human walking trajectories. As shown in Fig. 4, the walking motion was recorded by a camera from the ceiling. We obtained the walking trajectories by tracking each pedestrian head after the recording. Tau 640 (FLIR Systems, Inc.), an infrared camera, was used to make the image processing easy.

We measured the walking trajectories when a group of pedestrians walk straight, avoiding the collision with a person standing still in the center. Fig. 5 shows walking trajectories obtained by the image processing. Let $\boldsymbol{x}_0$ be the position of the person standing still, and $\boldsymbol{x}_i[k]$ be the trajectory data of the i-th pedestrian at time k. The velocity of the pedestrian $\boldsymbol{v}_i[k]$ can be calculated as follows:

$$\boldsymbol{v}_i[k] = \frac{\boldsymbol{x}_i[k+1] - \boldsymbol{x}_i[k]}{\Delta t} \tag{7}$$

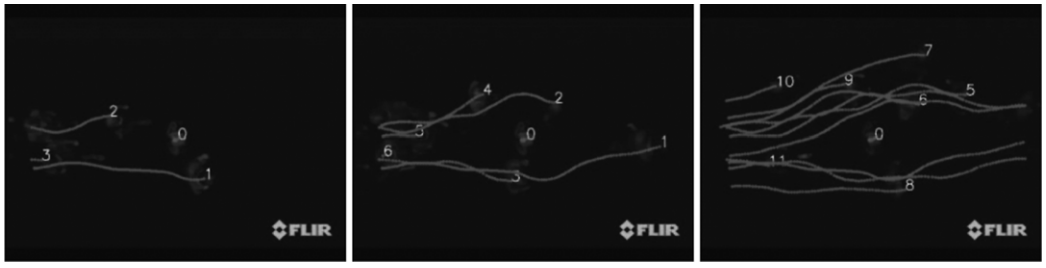


Fig. 5. Walking trajectories obtained from image processing.

where Δt is the sampling time. On the other hand, $\boldsymbol{v}_i[k]$ is given by the following equation in the particle model:

$$\boldsymbol{v}_i[k] = \boldsymbol{f}(\boldsymbol{x}_i[k]) - s(\|\boldsymbol{r}_i[k]\|)\frac{\boldsymbol{r}_i[k]}{\|\boldsymbol{r}_i[k]\|} \tag{8}$$

$$\boldsymbol{r}_i[k] = \boldsymbol{x}_0 - \boldsymbol{x}_i[k]. \tag{9}$$

The second term on the right-hand side of (8) is regarded as a function of the sigmoid function parameters. Therefore, we replace this term as follows:

$$\boldsymbol{g}(\boldsymbol{x}_i[k], \boldsymbol{a}) = s(\|\boldsymbol{r}_i[k]\|)\frac{\boldsymbol{r}_i[k]}{\|\boldsymbol{r}_i[k]\|} \tag{10}$$

$$\boldsymbol{a} = \begin{bmatrix} a & b & c \end{bmatrix}^T \tag{11}$$

where $\boldsymbol{a}$ is a vector of the sigmoid function parameters. Now, (8) is replaced by the following equation:

$$\boldsymbol{f}(\boldsymbol{x}_i[k]) - \boldsymbol{v}_i[k] = \boldsymbol{g}(\boldsymbol{x}_i[k], \boldsymbol{a}). \tag{12}$$

Considering all pedestrians ($i = 1, \ldots, m$) and time series ($k = 1, \ldots, n_i$), we calculate the parameter $\boldsymbol{a}$ which satisfies

$$\left\|\hat{\boldsymbol{f}} - \hat{\boldsymbol{v}} - \hat{\boldsymbol{g}}\right\|^2 \longrightarrow \min \tag{13}$$

where

$$\hat{\boldsymbol{f}} = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{x}_1[1]) & \cdots & \boldsymbol{f}(\boldsymbol{x}_m[n_m]) \end{bmatrix}^T \tag{14}$$

$$\hat{\boldsymbol{v}} = \begin{bmatrix} \boldsymbol{v}_1[1] & \cdots & \boldsymbol{v}_m[n_m] \end{bmatrix}^T \tag{15}$$

$$\hat{\boldsymbol{g}} = \begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_1[1], \boldsymbol{a}) & \cdots & \boldsymbol{g}(\boldsymbol{x}_m[n_m], \boldsymbol{a}) \end{bmatrix}^T. \tag{16}$$

In this paper, we computed $\boldsymbol{a}$ by the Newton–Raphson method, instead of directly solving (13). The parameters of the sigmoid function were calculated as $a = 10, b = 0.8$ and $c = 2.5$. We also obtained the reference velocity $v_0 = 1.34$ (m/s) from the trajectories.

2.4. Simulation of crossing flows with the particle model

We simulate the crossing pedestrian flows based on the particle model. Fig. 6 shows a snapshot of the simulation. Particles in flows A and B were input randomly in $x = -15.0, |y| \leq 7.5$ and $|x| \leq 7.5, y = -15.0$, respectively. We observed that a diagonal stripe pattern emerged after two flows have crossed. This result is consistent with the phenomenon mentioned in [14]. Therefore, it is verified that the modeling of pedestrian flows with the velocity field is appropriate. Note that the diagonal stripe pattern slightly changes from hour to hour because the particles are input randomly.

The particle model is appropriate to model microscopic characteristics of each pedestrian, e.g. age, gender, social relationship and so on. However, it is difficult to quantify the dynamic change of the congestion degree in the crossing area because the particle model is discrete. When we control the swarm behavior, it is important to quantify macroscopic characteristics of pedestrians. Therefore, we propose the continuum model in the following section.

3. Continuum model of crossing pedestrian flows

3.1. Density calculation based on the continuity equation

We quantify the congestion degree of flow $i = A, B$ as the density of the continuum ρ_i. Letting $\boldsymbol{v}_i = \begin{bmatrix} u_i & v_i \end{bmatrix}^T$ denote the velocity

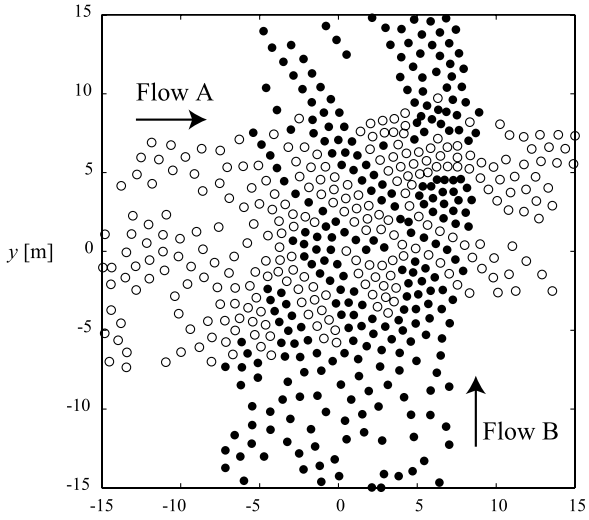


Fig. 6. Simulation of the crossing flows with the particle model.

of the i-th continuum, time variation of the flow density is given by the following continuity equation in a similar way to [10,22]:

$$\frac{\partial \rho_i}{\partial t} = -\rho_i\left(\frac{\partial u_i}{\partial x} + \frac{\partial v_i}{\partial y}\right) - \left(\frac{\partial \rho_i}{\partial x}u_i + \frac{\partial \rho_i}{\partial y}v_i\right). \tag{17}$$

In a similar way to the previous section, suppose that the macroscopic behavior of each pedestrian flow i is given by the vector field $\boldsymbol{f}_i$. Then, velocities of each continuum are given as follows:

$$\boldsymbol{v}_A = \boldsymbol{f}_A(\boldsymbol{x}) - k_1\nabla\rho_A - k_2\nabla\rho_B \tag{18}$$

$$\boldsymbol{v}_B = \boldsymbol{f}_B(\boldsymbol{x}) - k_1\nabla\rho_B - k_2\nabla\rho_A \tag{19}$$

where $\nabla\rho_i$ represents the following density gradient:

$$\nabla\rho_i = \begin{bmatrix} \dfrac{\partial \rho_i}{\partial x} & \dfrac{\partial \rho_i}{\partial y} \end{bmatrix}^T. \tag{20}$$

The second and third terms on the right-hand side of (18) and (19) are diffusion terms of each continuum, where k_1 and k_2 are their coefficients. These terms represent the effect to avoid the collision with other pedestrians, which is equivalent to the second term on the right-hand side of (4) in the particle model. Note that this continuum model given by (17)–(19) is the velocity-level model in a similar way to the particle model.

3.2. Simulation of crossing flows with the continuum model

We simulated the density variation of the crossing flows based on the continuum model. Continuity equation (17) was calculated with the finite volume method. Assuming that pedestrians enter steadily, boundary conditions of the density were given as follows:

$$\begin{cases} \rho_A(\boldsymbol{x}, t) = \rho_{A0} & (x = -2, |y| \leq 0.5) \\ \rho_B(\boldsymbol{x}, t) = \rho_{B0} & (|x| \leq 0.5, y = -2). \end{cases} \tag{21}$$

We call ρ_{i0} *input density*. In this paper, we consider a case of $\rho_{A0} = \rho_{B0} = \rho_0$ for simplicity. Fig. 7 shows a simulation result of spatial distribution of density in steady state, setting $\rho_0 = 14$. In the figure, the white color indicates $\rho_i = 0$, and the black color indicates the highest density. We observed that the diagonal stripe pattern emerged after the flows crossed. This result is consistent with the diagonal stripe pattern formation in the actual phenomenon mentioned in [14]. Therefore, the validity of the continuum model is qualitatively verified. Moreover, the advantage of this model is that it is possible to quantify the dynamic change of the congestion degree in the crossing area.

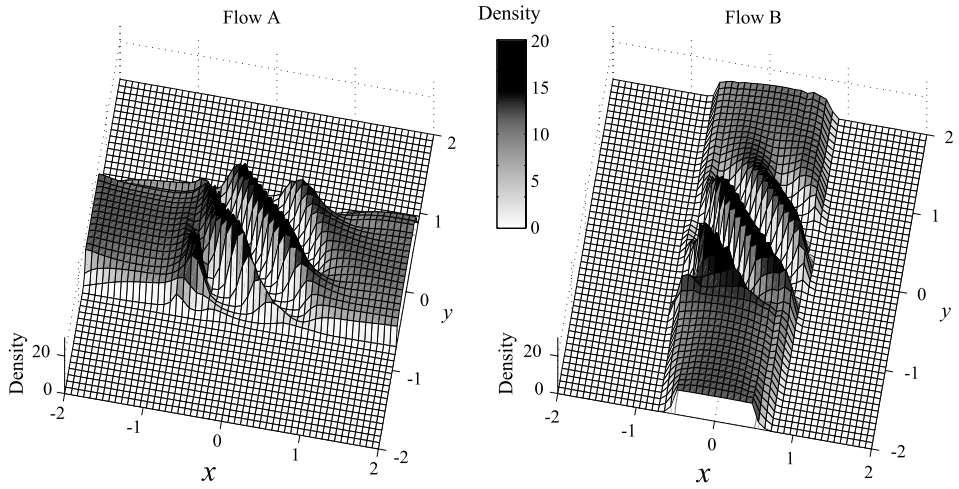


Fig. 7. Simulation result of the density distribution in the crossing flows with the continuum model.

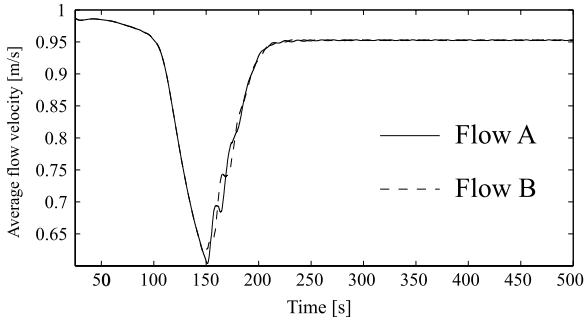


Fig. 8. Time variation of average velocity of the crossing flows.

3.3. Average flow velocity of the continuum model

In order to evaluate how smooth the pedestrian flows are, we define an average flow velocity. From the flow velocity $\boldsymbol{v}_i(\boldsymbol{x}, t)$ at position $\boldsymbol{x}$, we extract an element parallel to the original vector field $\boldsymbol{f}_i(\boldsymbol{x})$ as follows:

$$\hat{\boldsymbol{v}}_i = \boldsymbol{f}_i \boldsymbol{f}_i^{\#} \boldsymbol{v}_i \tag{22}$$

where $\boldsymbol{f}_i^{\#}$ is pseudo-inverse of $\boldsymbol{f}_i$ defined by

$$\boldsymbol{f}_i^{\#} = (\boldsymbol{f}_i^T \boldsymbol{f}_i)^{-1} \boldsymbol{f}_i^T. \tag{23}$$

Using $\hat{\boldsymbol{v}}_i$, we define the average velocity of time t as follows:

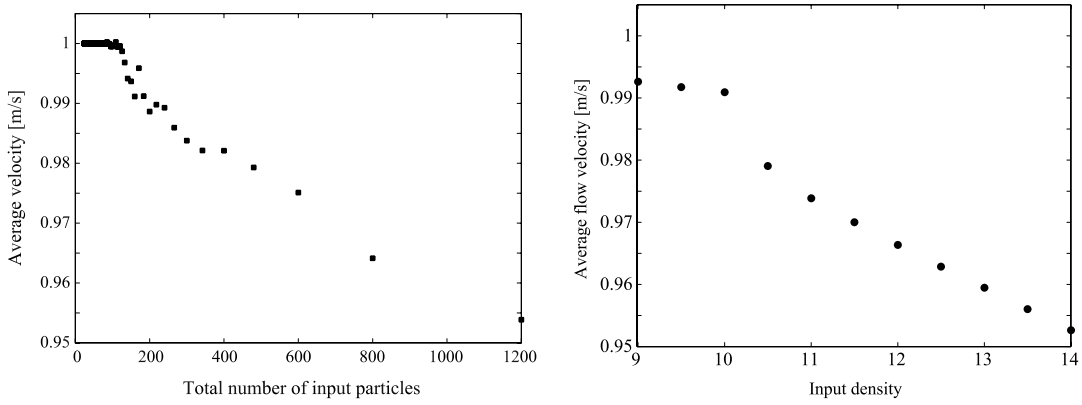
$$\bar{v}_i(t) = \frac{\int \rho_i \|\hat{\boldsymbol{v}}_i\| d\boldsymbol{x}}{\int \rho_i \, d\boldsymbol{x}} \quad (i = \mathrm{A}, \mathrm{B}). \tag{24}$$

The right-hand side of (24) implies proportionality of the total amount of flow rate to the density. Fig. 8 shows time variation of the average velocity in the crossing flows. The two flows collide at around 100 s and the velocity declines rapidly. After that, the stripe pattern begins to emerge at 150 s. After 200 s, the stripe pattern propagates steadily and the velocity recovers.

3.4. Analysis on density and velocity with the particle and continuum model

In order to verify the validity of the continuum model, we compared congestion–velocity relationship between the particle and continuum model.

Firstly, in the particle model, we calculated the average velocity when the number of input particles in the total simulation time was increased. Fig. 9(a) shows the variation of the average velocity of the particles. When the total input number of particles was 0 to 50, particles moved with the reference velocity v_0. As the total input number of particles increased, the particles collided with each other more often. Then, when the total input number became more than about 50, the congestion occurred and the average velocity decreased rapidly.



(a) Particle model. (b) Continuum model.

Fig. 9. Variation of the average velocity when the number of input is increased.

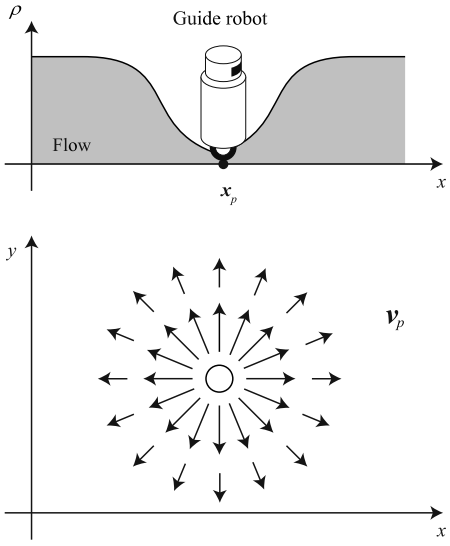


Fig. 10. Modeling of a guide robot and its effect to flows.

Next, in the continuum model, we calculated the average velocity of the crossing flows when the input density ρ_0 was increased. Fig. 9(b) shows the variation of the average velocity. When the input density was smaller than 10, the diagonal stripe pattern did not emerge and decreasing rate of the average flow velocity was relatively small. When the input density was larger than 10, the diagonal stripe pattern emerged in the crossing area and the average flow velocity decreased rapidly, in a similar way to the particle model. From the results shown in Fig. 9(a) and (b), we observed that the characteristics of the particle model did not coincide well with one of the continuum model when the total input number of particles was 0–200. Then, the characteristics of the particle model became similar to one of the continuum model when the total input number of particles increased. Therefore, the validity of the continuum model was verified because both the particle and continuum models had the same congestion–velocity relationship when the number of particles was sufficiently large.

4. Control of swarm behavior in crossing pedestrian flows

4.1. Implicit control of swarm behavior by guide robots

In this paper, we control the swarm behavior in the crossing pedestrian flows by utilizing dynamic interaction caused by guide robots moving in the flows. We call this method *implicit control of swarm behavior* because the robots do not explicitly guide pedestrian. We assume that it is possible to measure the position of robots and remotely control them. In this paper, we derive the desired position of robots to make the flows smooth.

Suppose that a robot moves in a pedestrian flow. Pedestrians try to avoid collisions with the robot. As shown in Fig. 10, let $\boldsymbol{x}_p$ denote the position of the robot p. The collision avoidance effect of the pedestrians is modeled by the following repulsive velocity:

$$\boldsymbol{v}_p = -s(\|\boldsymbol{r}_p\|)\frac{\boldsymbol{r}_p}{\|\boldsymbol{r}_p\|} \tag{25}$$

$$\boldsymbol{r}_p = \boldsymbol{x}_p - \boldsymbol{x} \tag{26}$$

where $\boldsymbol{r}_p$ is the relative position from the robot. $s(r)$ is the sigmoid function defined by (6). In particular, the parameter b defines the size of a guide robot. Adding the repulsive effect to (18) and (19), velocities of the flows are given as follows:

$$\boldsymbol{v}_A = \boldsymbol{f}_A(\boldsymbol{x}) - k_1\nabla\rho_A - k_2\nabla\rho_B + \sum_p \boldsymbol{v}_p \tag{27}$$

$$\boldsymbol{v}_B = \boldsymbol{f}_B(\boldsymbol{x}) - k_1\nabla\rho_B - k_2\nabla\rho_A + \sum_p \boldsymbol{v}_p. \tag{28}$$

One solution to find an algorithm to move the robots would be calculating $\boldsymbol{x}_p$ from (25) to (28) and (17) so that $\boldsymbol{v}_A$ and $\boldsymbol{v}_B$ are increased. However, this calculation is extremely difficult because we need to solve a second-order partial differential equation with respect to time and space.

Therefore, considering the self-organized phenomenon in the flows, we give a rough guideline to move the robots as follows:

1. The self-organized diagonal stripe pattern is regarded as a nonlinear oscillating phenomenon with temporal and spatial frequencies. It is known that the entrainment or synchronization arises from the interaction between nonlinear oscillators. Therefore, it is expected that we can change the diagonal stripe pattern by externally adding a cyclic effect. Therefore, we add an external effect to the flows by moving guide robots in a cycle.
2. From the aspect to cost, the less number of robots is the better. Moreover, the diagonal stripe pattern is a symmetric phenomenon about the two flows. From those reasons, we move two guide robots in the flows.
3. The diagonal stripe pattern is generated by the local collision effect of the flows which propagates to the crossing area. The local collision is caused at borders of the flows. It is considered that we can add the robot effect more efficiently by moving them along the borders.

From the above guideline, we put two guide robots, α and β, in a cycle as shown in Fig. 11. The positions of the two robots, $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}_\beta$, are given as follows:

$$\boldsymbol{x}_\alpha = \boldsymbol{c} + w\{1 - \cos(2\pi\omega_G t)\}\boldsymbol{d}_B \tag{29}$$

$$\boldsymbol{x}_\beta = \boldsymbol{c} + w\{1 - \cos(2\pi\omega_G t + \pi)\}\boldsymbol{d}_A \tag{30}$$

where $\boldsymbol{c}$ is the intersection of the flows as shown in Fig. 11, w is the width of the flow, and ω_G is the frequency of the guide robot movement. Hereafter, we call ω_G *guide frequency*. $\boldsymbol{d}_A = [1\ 0]^T$ and $\boldsymbol{d}_B = [0\ 1]^T$ are directional vectors of each flow. Considering that the density of each flow has the same frequency and the opposite phase, we set the phases of the robots to be opposite.

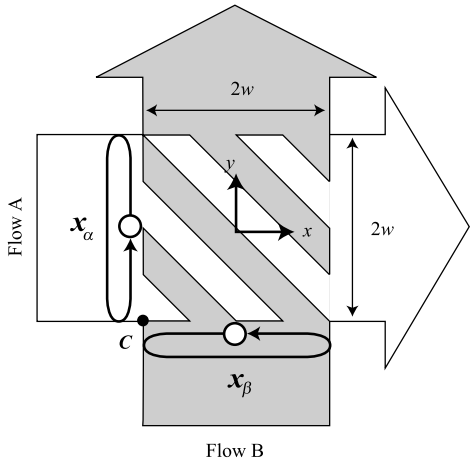
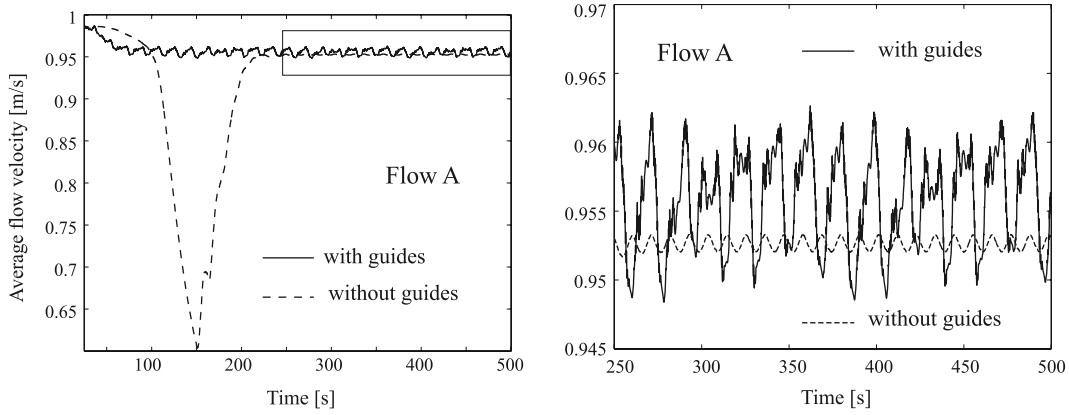
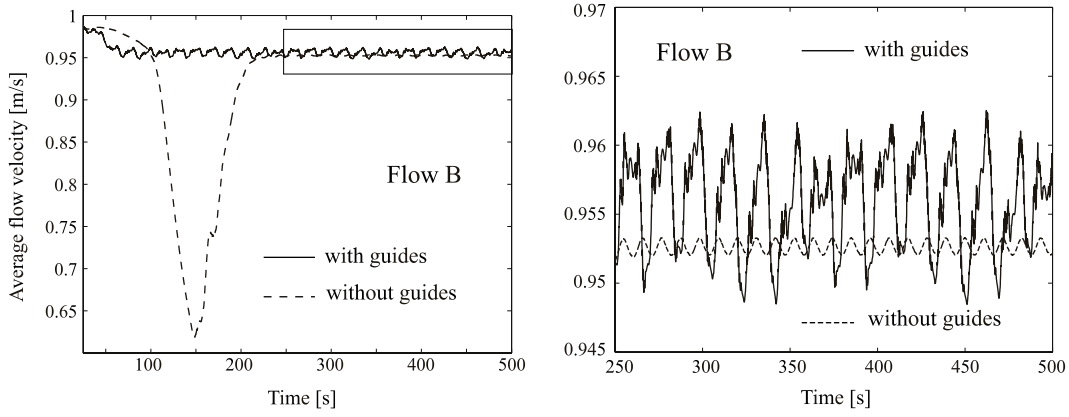


Fig. 11. Cyclic movement of guide robots in the flows.



(a) Flow A.



(b) Flow B.

Fig. 12. Time variation of the average flow velocity when the guide robots moved in a cycle. The guide frequency was set to be $\omega_G = 0.055$ Hz.

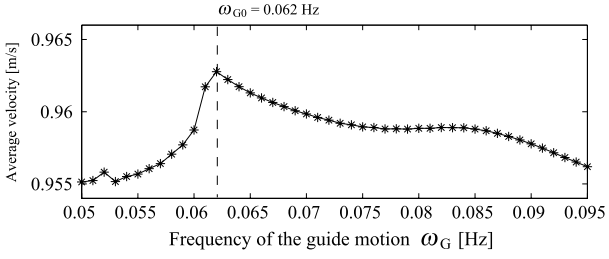


Fig. 13. Relationship between the guide frequency ω_G and the average flow velocity. The input density was set to be $\rho_0 = 14$.

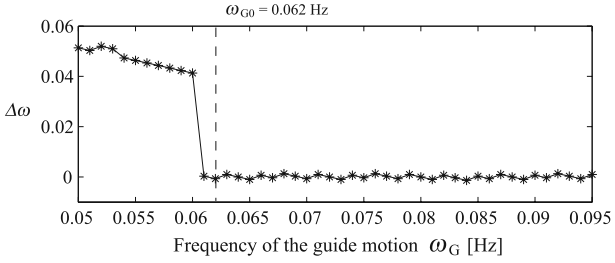


Fig. 14. Relationship between the guide frequency ω_G and temporal frequency of the flows ω_G. The vertical axis indicates the difference between ω_G and ω_F, $\Delta\omega = \omega_F - \omega_G$. The input density was set to be $\rho_0 = 14$.

4.2. Velocity analysis of crossing flows

We analyze how the swarm behavior in the flows changes by the dynamic interaction with the guide robots. First, we analyzed the average flow velocity when the guide robots moved in a cycle. For simplicity, the input densities of two flows are set to be equal, namely $\rho_{A0} = \rho_{B0} = \rho_0$. Fig. 12 shows the time variation of the average flow velocity when $\rho_0 = 14$ and $\omega_G = 0.055$ Hz, for example. The rapid decrease of the velocity due to collision of the flows was prevented. The right columns of Fig. 12 show closeup of steady state between 250 s and 500 s. In both A and B, the average flow velocity was slightly increased.

Next, we analyzed relationship between the average flow velocity and the frequency of the guide robots ω_G. The average flow velocity in the steady state was simulated changing ω_G from 0.050 to 0.095 Hz by 0.001 Hz step size, where the input density was $\rho_0 = 14$. Fig. 13 shows the result of flow A. The result of flow B is omitted because it is almost same with one of flow A. In the figure, the average flow velocity is maximized when $\omega_{G0} = 0.062$ Hz (indicated by the dashed line). In general, ω_{G0} is unknown because it varies depending on the flow rate. If we can find ω_{G0} automatically, the swarm behavior can be controlled so that the average flow velocity is maximized.

4.3. Temporal/spatial frequencies analysis of crossing flows

In order to find ω_{G0}, we analyze the characteristics of the crossing flows phenomenon when the guide frequency ω_G is lower or higher than ω_{G0}. In particular, we focus on temporal and spatial frequencies of the crossing flows. Temporal frequency is computed by FFT analysis from time-series data of the density at a representative position, for example, the center position of the crossing area. Let ω_F denote the temporal frequency of the flows. From some simulations, we found that the value of ω_F is equal for both flows A and

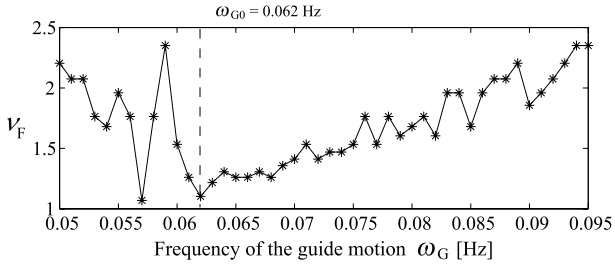


Fig. 15. Relationship between the guide frequency ω_G and spatial frequency of the flows ν_F. The input density was set to be $\rho_0 = 14$.

B. Then, we define the error between ω_F and ω_G as follows:

$$\Delta\omega = \omega_F - \omega_G. \tag{31}$$

Fig. 14 shows the relationship between ω_G and $\Delta\omega$. In the figure, the dashed line indicates the optimal frequency ω_{G0}. From this result, the relationship is summarized as follows:

- In $\omega_G < \omega_{G0}$, $\Delta\omega$ decreases as ω_G increases.
- In $\omega_G \geq \omega_{G0}$, $\Delta\omega$ is almost zero.

Next, we analyze the characteristics of the spatial frequency of the crossing flows. The inverse of the spatial frequency is equivalent to the width of the density stripe pattern, which is computed from spatial distribution of the density. Let ν_F denote the spatial frequency. Similar to the temporal frequency, we found that the value of ν_F is equal for both flows A and B. Fig. 15 shows relationship between ω_G and ν_F. From this result, the relationship is summarized as follows:

- In $\omega_G < \omega_{G0}$, no obvious relationship is found.
- In $\omega_G \geq \omega_{G0}$, ν_F increases as ω_G increases.

The above results were obtained under the condition $\rho_0 = 14$. Results when the density input is given as $\rho_0 = 11, 12, 13, 15$ are shown in Fig. 16. The upper, middle and lower rows of Fig. 16 show the average flow velocity, $\Delta\omega$ and ν_F, respectively. These results are similar to those shown in Figs. 13–15. We consider that the average flow velocity is increased by some kinds of resonance effect between the guide robots and the flows, though it is qualitative discussion.

4.4. Control algorithm based on temporal/spatial frequencies

From the above discussion, it is possible to find the optimal frequency ω_{G0} by adjusting ω_G as follows:

1. In the low-frequency area, ω_G is increased so that $\Delta\omega$ becomes zero.
2. In the high-frequency area, ω_G is decreased so that ν_F becomes low.

We can switch between these two strategies depending on if $\Delta\omega$ is equal to zero or not.

Let ${}^i\omega_G$ denote the guide frequency in the i-th period. From the temporal and spatial frequencies of the flows, ${}^i\omega_F$ and ${}^i\nu_F$, we determine the frequency of the robots in the next period ${}^{i+1}\omega_G$ as follows:

$${}^{i+1}\omega_G = \begin{cases} {}^i\omega_G + k_\omega \Delta\omega & (\Delta\omega \geq \Delta\omega_0) \\ {}^i\omega_G + k_\nu(\nu_0 - {}^i\nu_F) & (\Delta\omega < \Delta\omega_0) \end{cases} \tag{32}$$

where k_ω and k_ν are the gains of the temporal and spatial frequencies, respectively. ν_0 is an offset value of the spatial frequency to prevent ω_G from changing rapidly near ω_{G0}. $\Delta\omega_0$ is a threshold of $\Delta\omega$.

4.5. Simulation of crossing flows control

We simulated the density variation in the flows by applying the proposed control method. The input density was set to be $\rho_0 = 14$, and the control parameters were set as follows: $k_\omega = 0.08$, $k_\nu = 0.001$, $\nu_0 = 1.0$. Fig. 17 shows the average flow velocities, in which the right column shows closeup of the steady state between 250 and 500 s. We observed that the average flow velocities were increased by applying the proposed control method. Fig. 18 shows time variation of the guide robot frequency and the temporal frequency of the flows. We observed that the guide robot frequency was adjusted so that $\Delta\omega$ becomes zero. Fig. 19 shows snapshots of the spatial distribution of the density. White poles indicate the position of the guide robots. We also observed that the width of the stripe increased. This is because the spatial frequency was minimized by the proposed algorithm. As mentioned in Section 4.3, the inverse of the spatial frequency is equivalent to the stripe width. Therefore, minimizing the spatial frequency resulted in maximizing the stripe width.

5. Crossing pedestrian flows control under different conditions

5.1. Control by a single guide robot

In the previous section, we assumed *symmetric* situation, which means input densities of the both flows are equal and time-invariant. From this point of view, we moved two guide robots in the flows. In this section, we consider the case that there is a single guide robot and analyze the effect of interaction between the flows and the robot.

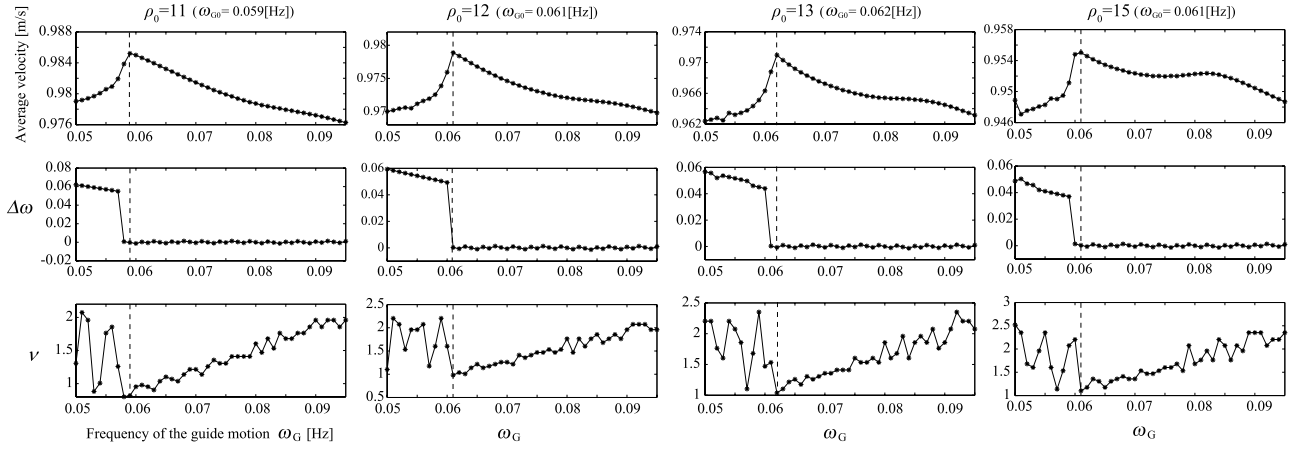
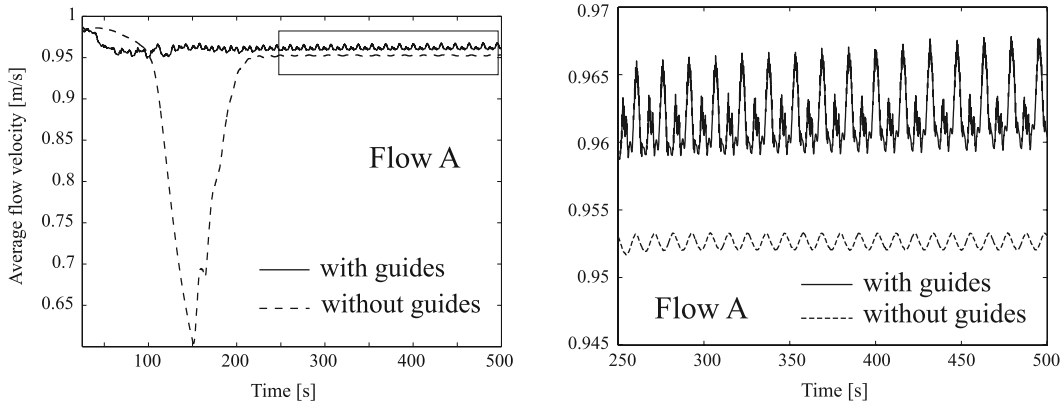
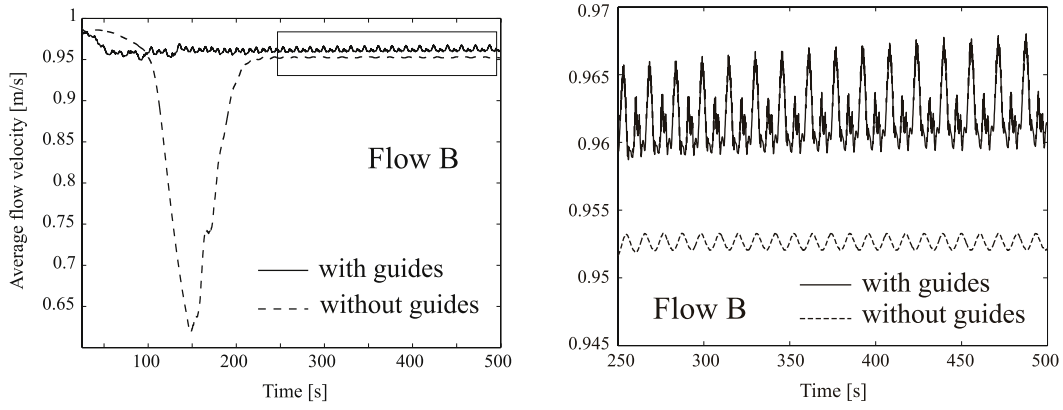


Fig. 16. Result of the temporal and spatial frequencies analysis.



(a) Flow A.



(b) Flow B.

Fig. 17. Time variation of the average flow velocity with the proposed control method.

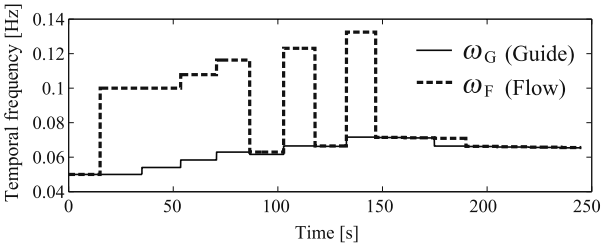


Fig. 18. Time variation of the guide robot frequency ω_G and the temporal frequency of the flows ω_F.

In a similar way to Section 4.2, we analyze the relationship between the average flow velocity and the frequency of the single robot when removing robot β. The input density was set to be $\rho_0 = 14$ in a similar way to Fig. 13. The result is shown in Fig. 20, where white and black circles indicate the velocities of flows A and B, respectively. The dashed line indicates the result with two guides, which is shown in Fig. 13. We can observe that the velocity of A becomes less than B, by removing the robot β. Therefore, at least two guide robots are required to improve the velocities of both flows simultaneously.

5.2. Frequency analysis under different input densities

In the above discussion, we supposed that input densities of the both flows are equal and time-invariant. In the real environments, however, the input densities are not always equal and constant. Instead they change dynamically. In this section, we verify that the proposed control algorithm is applicable to such a situation. Firstly, we verify that the proposed algorithm is applicable to the crossing pedestrian flows with different input densities. Fig. 21 shows the result of the temporal/spatial frequencies analysis when the input densities are given by $\rho_{A0} = 15$ and $\rho_{B0} = 7.5$. If the proposed algorithm is applied, the guide frequency will be modulated to the frequency indicated by the dashed line in the figure. In this frequency, the average flow velocity of both flows are almost maximized. Therefore, it is expected that the proposed algorithm improves both the flow velocities simultaneously even though the input densities are different.

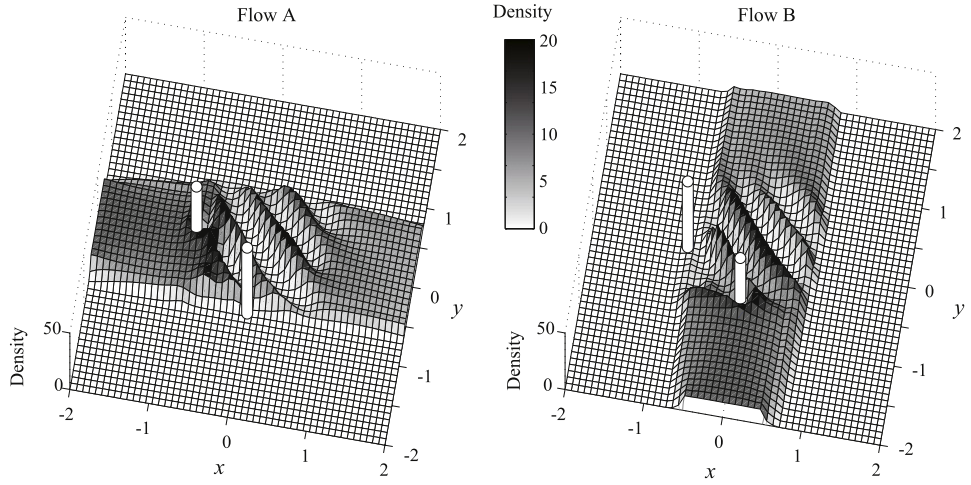
5.3. Crossing flows control with changing input densities

We simulated the control of the crossing pedestrian flows with the input densities changing. In the simulation, a stepwise change of the input densities was given as shown in Fig. 22. Fig. 23 shows time variation of the average flow velocities from 250 to 600 s. We observed that both velocities were improved by applying the proposed method.

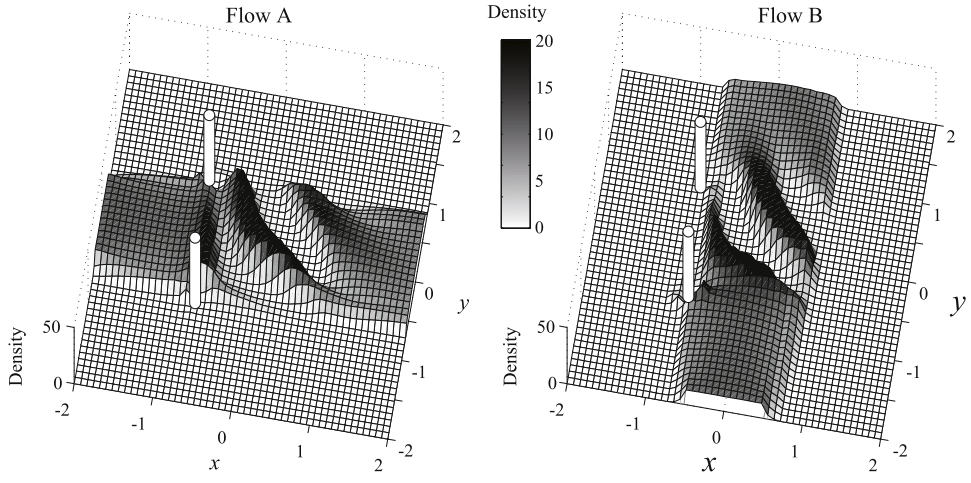
6. Application of the control algorithm to the particle model

6.1. Computation of virtual density from particles

In the proposed control algorithm, we utilize information of the density distribution, which is continuous quantity. On the other hand, information of position or number of actual pedestrians is discrete quantity. Therefore, we need to compute the density from the position of pedestrians in order to apply the proposed control method to the actual environments. In this section, we compute virtual density from the position information of pedestrians.



(a) $t = 50$ s.



(b) $t = 250$ s.

Fig. 19. Simulation result of spatial distribution of density in the flows with the proposed control method.

Virtual density $\hat{\rho}(\boldsymbol{x})$ at a position $\boldsymbol{x}$ is calculated as follows:

$$\hat{\rho}(\boldsymbol{x}) = \sum_i W(\|\boldsymbol{r}_i\|, h) \tag{33}$$

$$\boldsymbol{r}_i = \boldsymbol{x} - \boldsymbol{x}_i \tag{34}$$

where $\boldsymbol{x}_i$ is the position of a pedestrian, and $\boldsymbol{r}_i$ is the relative position vector from the pedestrian, respectively. $W(x, h)$ is the cubic spline function defined by the following equation:

$$W(x,h) = \begin{cases} \dfrac{10}{7\pi h^2}\left\{1 - \dfrac{3}{2}\left(\dfrac{x}{h}\right)^2 + \dfrac{3}{4}\left(\dfrac{x}{h}\right)^3\right\} & (0 \le x < h) \\ \dfrac{5}{14\pi h^2}\left\{2 - \left(\dfrac{x}{h}\right)^3\right\} & (h \le x < 2h) \\ 0 \quad (x \ge 2h). \end{cases} \tag{35}$$

$W(x, h)$ represents pseudo density distribution set to one particle, and h specifies its radius.

6.2. Control simulation of crossing pedestrian flows

We applied the proposed algorithm to the particle model and simulated the control of the crossing pedestrian flows. In the simulation, we used the parameters identified from the measurement data of the actual walking trajectories. The parameter b of the robot, which defines the robot size, was set to be twice as much as one of pedestrians. Fig. 24 shows a snapshot of the simulation.

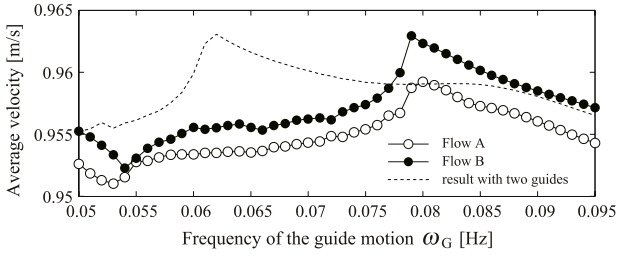


Fig. 20. Relationship between the frequency of the guide and the average flow velocities when there is one guide.

The white and black circles indicate particles of flows A and B, respectively. The white squares indicate the guide robot α and β. Fig. 25(a) and (b) show time variation of the average velocity of each flow. In the figures, the dashed lines indicate the result described in Section 2.4, which means there is no guide robots and control. The solid lines indicate results with the guide robots. Table 1 shows the time average of the values in Fig. 25(a) and (b). We observed that the average velocities of both flows were increased by applying the control with the guide robots. Moreover, comparing Fig. 24 with Fig. 6, we observed that the stripe width became larger by applying the control. This result is consistent with one of the continuum model mentioned in Section 4.5. Fig. 26 shows time variation of the density in the crossing area. We observed that the density decreased by applying the control method.

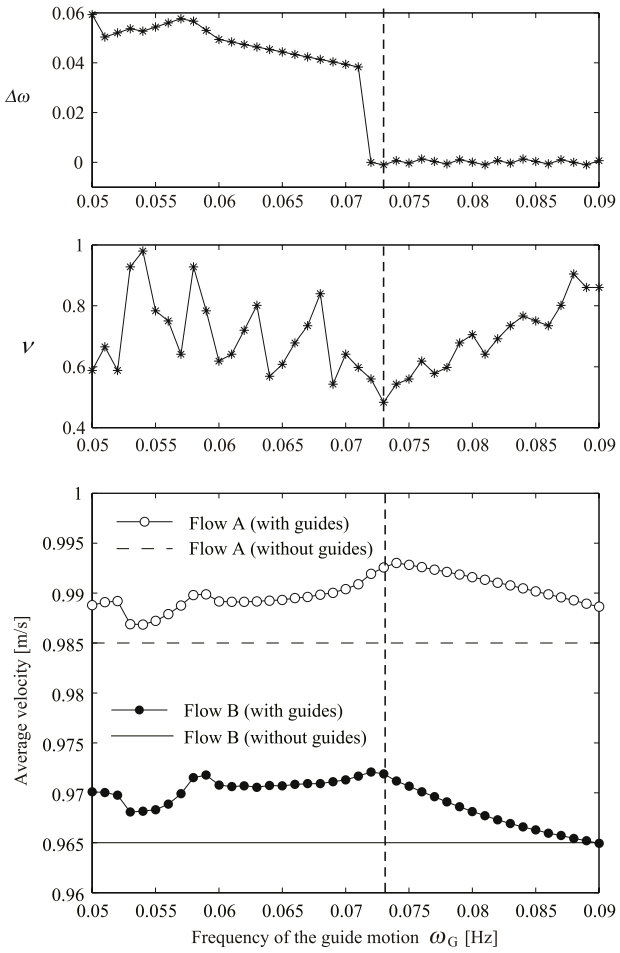


Fig. 21. Relationship between the frequency of guides and the average velocity ($\rho_{A0} = 15$, $\rho_{B0} = 7.5$).

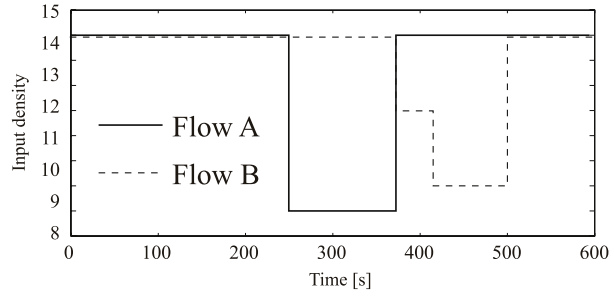


Fig. 22. Time variation of the input density.

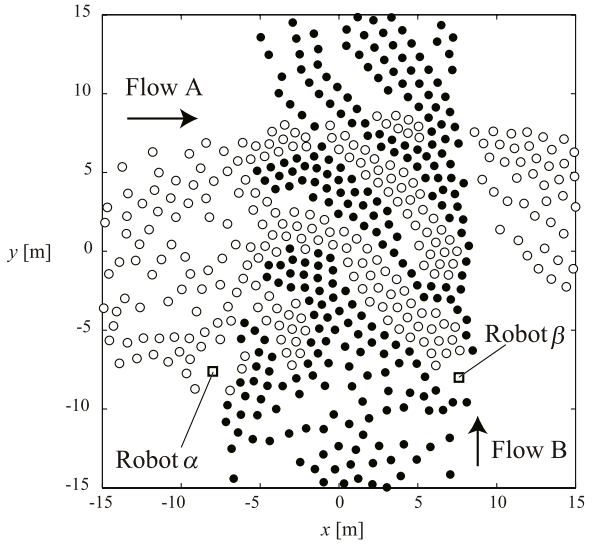


Fig. 24. Snapshot of control simulation of the crossing pedestrian flows.

Table 1
Temporal average velocity (m/s) of the particles.

	Flow A	Flow B
Without guides	1.11	1.14
With guides	1.20	1.19

7. Conclusion

The results of this paper are summarized as follows:

1. The particle and continuum models of the crossing pedestrian flows were proposed. Both models makes it possible to simulate the diagonal stripe pattern phenomenon. In particular, the continuum model makes it possible to quantitatively evaluate the dynamic congestion change.
2. Implicit control of swarm behavior in the crossing pedestrian flows was proposed. Focusing on the nonlinear oscillating phenomenon of the diagonal stripe pattern, we can change the swarm behavior by utilizing the dynamic interaction between guide robots and flows. In this paper, we analyzed the relationship between frequencies of the guide robots and flows. From the analysis results, we derived the control algorithm to improve the average flow velocities.

The proposed control algorithm is valid even when the input densities are different. Moreover, it was validated that we can apply the proposed algorithm to the particle model by calculating virtual density. The parameters of the particle model were identified by

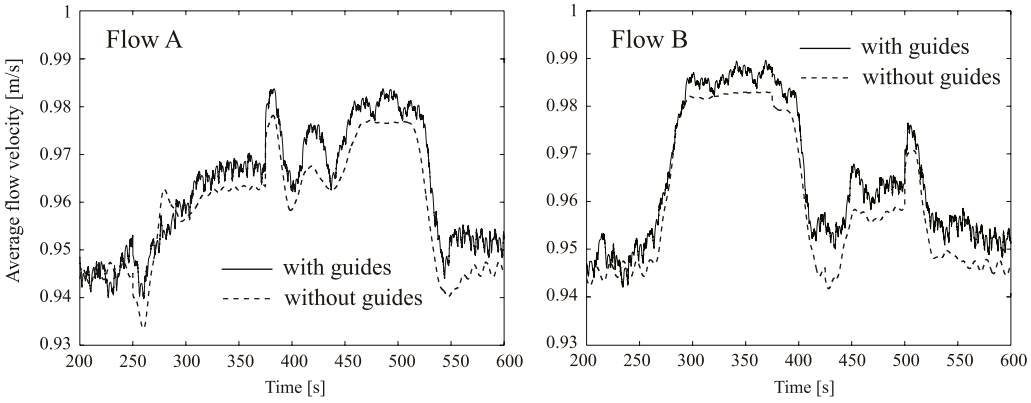
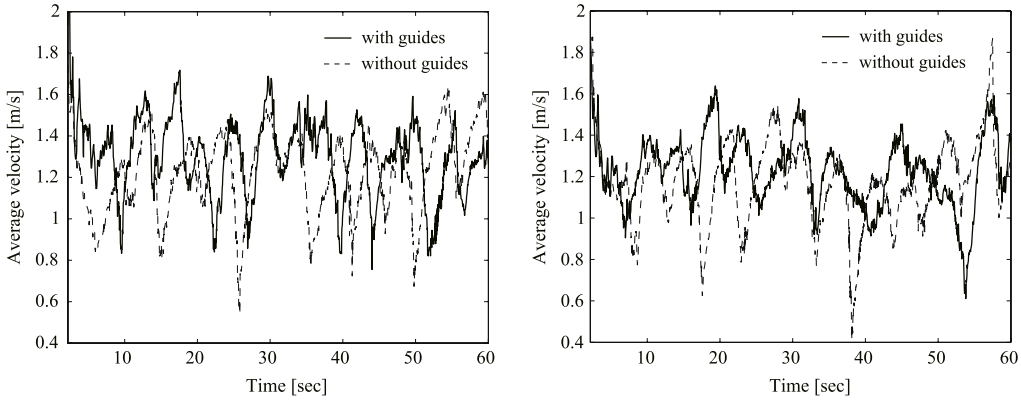


Fig. 23. Average flow velocity with the proposed control method when the density varies from hour to hour.



(a) Flow A. (b) Flow B.

Fig. 25. Time variation of the average velocities in the control simulation.

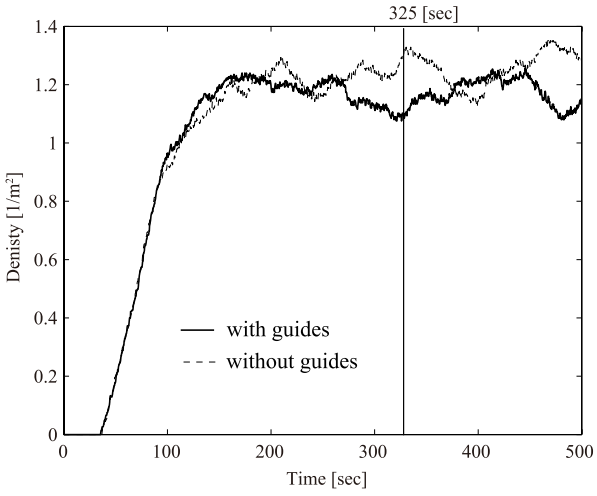


Fig. 26. Time variation of the density in the crossing area.

measuring the actual pedestrians. Using the obtained parameters, the validity of the proposed algorithm was observed more clearly than the result of our previous paper [23], which is that the stripe width was changed by the control.

The proposed model is homogeneous because we focus on the macroscopic behavior which emerges in the crossing pedestrian flows. In the real situations, there is some heterogeneity in the pedestrian flows because of the individual differences, such as age, gender or social relationship. We consider that the proposed modeling and control methods are effective when there is a large number of people and the control is applied for a long period. In this paper, we considered line-shaped pedestrian flows. The proposed modeling and control methods can be extended to more complicated situation by designing the velocity vector field based on [22].

Acknowledgment

This research was supported by the Research on Macro/Micro Modeling of Human Behavior in the Swarm and its Control under the Core Research for Evolutional Science and Technology (CREST) Program (research area: Advanced Integrated Sensing Technologies), Japan Science and Technology Agency (JST).

References

[1] D. Helbing, P. Molnár, Social force model for pedestrian dynamics, Physical Review E 51 (1995) 4282–4286.
[2] J.V. den Berg, M. Lin, D. Manocha, Reciprocal velocity obstacles for real-time multi-agent navigation, in: Proceedings of the 2008 IEEE International Conference on Robotics and Automation, pp. 1928–1935.
[3] I. Karamouzas, M. Overmars, Simulating human collision avoidance using a velocity-based approach, in: Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS.
[4] J. Pettre, J.-P. Laumond, D. Thalmann, A navigation graph for real-time crowd animation on multilayered and uneven terrain, in: Proceedings of the First International Workshop on Crowd Simulation, V-CROWD'05, pp. 81–89.
[5] N. Pelechano, J.M. Allbeck, N.I. Badler, Controlling individual agents in high-density crowd simulation, in: Eurographics/ACM SIGGRAPH Symposium on Computer Animation, pp. 99–108.
[6] S. Guy, J. Chhugani, S. Curtis, P. Dubey, PLEdestrians: a least-effort approach to crowd simulation, in: Proceedings of Eurographics/ACM SIGGRAPH Symposium on Computer Animation.
[7] V.J. Blue, J.L. Adler, Modeling four directional pedestrian movements, Transportation Research Record 1710 (2000) 20–27.
[8] M. Asano, M. Kuwahara, A. Sumalee, S. Tanaka, E. Chung, Pedestrian simulation considering stochastic route choice and multidirectional flow, in: International Symposium on Transport Simulation.
[9] D. Helbing, A fluid dynamic model for the movement of pedestrians gaskinetic equations, Complex Systems 6 (1992) 391–415.
[10] R.L. Hughes, A continuum theory for the flow of pedestrians, Transportation Research Part B: Methodological 1438 (2002) 51–535.
[11] A. Treuille, S. Cooper, Z. Popović, Continuum crowds, ACM Transactions on Graphics 25 (2006).
[12] D. Helbing, P. Molnár, Self-organization phenomena in pedestrian crowds, Statistical Mechanics (1998).
[13] S. Hoogendoorn, P.H.L. Bovy, Simulation of pedestrian flows by optimal control and differential games, Optimal Control Applications and Methods 172 (2003) 153–172.
[14] K. Ando, H. Oto, T. Aoki, Forecasting the flow of people, Railway Research Review 45 (1988) 8–13 (in Japanese).
[15] K. Kurumatani, Social coordination with architecture for ubiquitous agents: CONSORTS, in: Proceedings of International Conference on Intelligent Agents, Web Technologies and Internet Commerce, IAWTIC2003.
[16] T. Daito, J. Tanabe, M. Tange, K. Satou, S. Ota, T. Uchida, Pedestrian navigation system using RFID-tag: architecture and its evaluation based on a field trial, in: Proceedings of the 10th International Conference on Mobility and Transport for Elderly and Disabled People, TRANSED2004.
[17] M. Kourogi, N. Sakata, T. Okuma, T. Kurata, Indoor/outdoor pedestrian navigation with an embedded GPS/RFID/self-contained sensor system, in: Proceedings of the 16th International Conference on Artificial Reality and Teleexistence, ICAT 2006, pp. 1310–1321.
[18] T. Narumi, Y. Hada, H. Asama, K. Tsuji, Pedestrian route guidance system using moving information based on personal feature extraction, in: Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, pp. 94–99.
[19] R. Vaughan, N. Sumpter, J. Henderson, A. Frost, S. Cameron, Robot control of animal flocks, in: Proceedings of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference, pp. 277–282.
[20] J.-M. Lien, O.B. Bayazit, R.T. Sowell, S. Rodoriguez, N.M. Amato, Shepherding behaviors, in: Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA2004, pp. 4159–4164.
[21] M. Okada, T. Ando, Optimization of personal distribution for evacuation guidance based on vector field, in: Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS2011, pp. 3673–3678.
[22] M. Okada, Y. Homma, Amenity design for congestion reduction based on continuum model of swarm, in: Proceedings of the 13th International Conference on Mechatronics Technology.
[23] K. Yamamoto, M. Okada, Pedestrian swarm control based on temporal/spatial frequency of crossing flows, Journal of the Robotics Society of Japan 29 (2011) 737–744 (in Japanese).
[24] S. Shibuya, A study of the shape of personal space, Bulletin of Yamanashi Medical University 2 (1985) 41–49 (in Japanese).

Ko Yamamoto is an assistant professor at Ecotopia Science Institute, Nagoya University. He was born in Okinawa, Japan, in 1980. He received the B.S. and M.S. degrees in mechanical engineering from the University of Tokyo, Japan, in 2004 and 2006, respectively. He also received his Ph.D. from the University of Tokyo in 2009. He was a postdoctoral research fellow at Tokyo Institute Technology. He moved to Nagoya University as an assistant professor in May, 2012. His research interests include mechanical design and motion control of humanoid robots and control of swarm behavior in pedestrian flows. He is a member of the IEEE and Robotics Society of Japan.

Masafumi Okada received his Ph.D. from Kyoto University in 1996. He was a research associate of The University of Tokyo in 1997, he was a Lecture of The University of Tokyo in 2000, and he is now an Associate Professor of Tokyo Institute of Technology from 2004. His research interest includes the simultaneous design of the robot mechanism and intelligent control using the nonlinear dynamics point of view, and human behavior control.