IEICE Proceeding Series

Digital Circuit of Multi-Agent System for Two Pedestrian Streams in A Narrow Path

Takashi TOYOFUKU, Kunihiko MITSUBORI

Vol. 2 pp. 471-474
Publication Date: 2014/03/18
Online ISSN: 2188-5079

2013 International Symposium on Nonlinear Theory and its Applications
NOLTA2013, Santa Fe, USA, September 8-11, 2013

Digital Circuit of Multi-Agent System for Two Pedestrian Streams in A Narrow Path

Takashi TOYOFUKU and Kunihiko MITSUBORI

Department of Electronics and Computer Systems, Takushoku University
815-1, Tatemachi, Hachiouji-shi, Tokyo, 193-0985, Japan
Email: y2m308@st.takushoku-u.ac.jp, mitubori@es.takushoku-u.ac.jp

Abstract– We consider the model of two pedestrian streams which move opposite directions in a narrow path, based on multi-agent system. In this model, each pedestrian has a "personal space". The personal space is the region surrounding a person where it regards as psychologically the person's. This space plays an important role for the pedestrian's collision avoidance. In this paper, we design the digital circuit of this model, and verify its behavior by using Verilog-HDL.

1. Introduction

An agent is the entity which determines its action, based on the knowledge collected from its environment [1]. Multi-Agent System (MAS) consists of the agents which have influences on each other. Each agent has its state and action rule, and it executes the action determined by them. The agents have interactions between them which are caused by their actions, and the interactions have influences on the states of the agents.

The environment in MAS is often represented by lots of the unit squares. This type of MAS deeply relates to cellular automata. Based on MAS, many researchers have proposed the models of the vehicular traffic flow [2] and the pedestrian stream [3].

This paper designs the digital circuit of MAS. This design is oriented to the implementations on VLSI chip and FPGA. In these implementations, the accelerated and/or parallelized computation can be expected.

In the vehicular traffic flow, a simple and rational model has been proposed by Nagel et al. [4]. Also, the digital circuit of this model has been implemented on FPGA [5]. The behavior of a car on the road can be regarded as 1-dimensional flow, except an intersection. There are the traffic lights at the intersection, and they regulate the plural traffic flows. Due to this property, the digital circuit of the model of this flow is relatively easy to be designed.

In the pedestrian stream, many models have been proposed ever [3] [6]. However, the digital circuit of such model has not been designed yet. The pedestrian walks free in 2-dimenstional space, while it avoid the collisions with the obstacle and the other pedestrians. Then, there is not "the traffic light" for the pedestrian, and the pedestrian itself determines its action to avoid the collisions. Due to this property, the digital circuit of the model of this stream is hard to be designed.

In this paper, we consider the model of two pedestrian streams which move opposite directions in a narrow path. This is a simplified version of the model which has been proposed to investigate the pedestrian streams in a station yard [7]. In this model, each pedestrian has a "personal space" [8]. The personal space is the region surrounding a person where it regards as psychologically the person's [9]. Most people feel discomfort if the others intrude into this space. It plays an important role for the pedestrian's collision avoidance. We design the digital circuit of this model, and verify its behavior by using Verilog-HDL.

2. A Model of the Pedestrian Streams

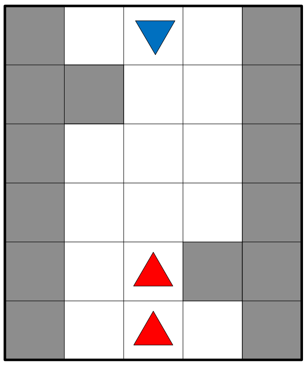

Fig.1 A model of the pedestrian streams.

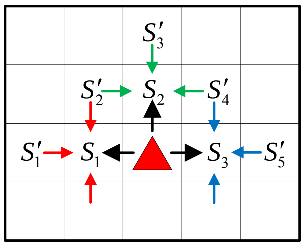

Fig.2 The agent's rule of the movement.

We first introduce a model of two pedestrian streams which move in the opposite directions in a narrow path [7]. Fig.1 shows a model of the pedestrian streams. This model consists of the stage and the agents. We refer to the unit square as "the grid". The stage consists of two kinds of the grid: normal grid (white) and no-entry grid (gray). The no-entry grids represent the walls and the obstacles.

In this stage, the upper edge is connected to the lower edge. The agent is a model of pedestrian. The agents are divided into two groups: “up-agents” and “down agents”. The up-agents go from the bottom to the top in this stage. Their candidates to move in a step are the left, the center, and the right of the current location. If the up-agent is turned upside down, it is the down-agent. The agents regard the agents who belong to the different group from theirs, as the obstacles. Each agent obeys the following rule:

We assign the grids around the agent to the symbols as shown in Fig.2. The grid S_i' has the point $P(S_i')$ to indicate the desirability of the kind or the situation of the grid. Also, the grid S_k has the evaluation value of the agent’s move, $Q(S_k)$. $Q(S_k)$ is defined by

$$Q(S_k) = \sum_{i=2k-1}^{2k+1} P(S_i') \quad (k = 1, 2, 3) \quad (1)$$

where if either of the following conditions is satisfied $P(S_i')$ is -1, otherwise it is zero:

(a) S_i' is the no-entry grid.

(b) S_i' is occupied by the agent who belongs to the different group from itself.

The agent selects S_k which maximizes $Q(S_k)$, as the next location. If S_k is occupied by the obstacles, the agent avoid it. If the agent has two candidates to move, it selects the next location by the above rule.

This rule is based on the concept of “personal space” [7]. The personal space is the region surrounding a person where it regards as psychologically the person’s [8]. This space plays an important role for the pedestrian’s collision avoidance.

3. Architecture of the Digital Circuit

Fig.3 shows architecture of the digital circuit which is governed by the model in the section-2. This circuit has the layered structure which consists of “the agent layer”, “the stage layer”, and “the decision-making layer”. We refer to the unit square which constitutes each layer, as “the cell”. The grid in the model expressed by the overlapping of the corresponding cells in these layers. We define the role and the function of each layer:

Agent layer: It memorizes the location of the agents. Each cell memorizes the value which is 1if this cell is occupied by the agent, or zero otherwise. This value changes every step as the agent moves. We prepare two agent layers for “up-agents” and “down-agents”.

Stage layer: It memorizes the location of the normal grid and the no-entry grid. Each cell memorizes the value which is zero if this cell corresponds to the normal grid or 1 if it does the no-entry grid. This layer determines the area where the agent is permitted to move.

Decision making layer: It receives the values from the cells in the above two layers and it determines the next location of the agent.

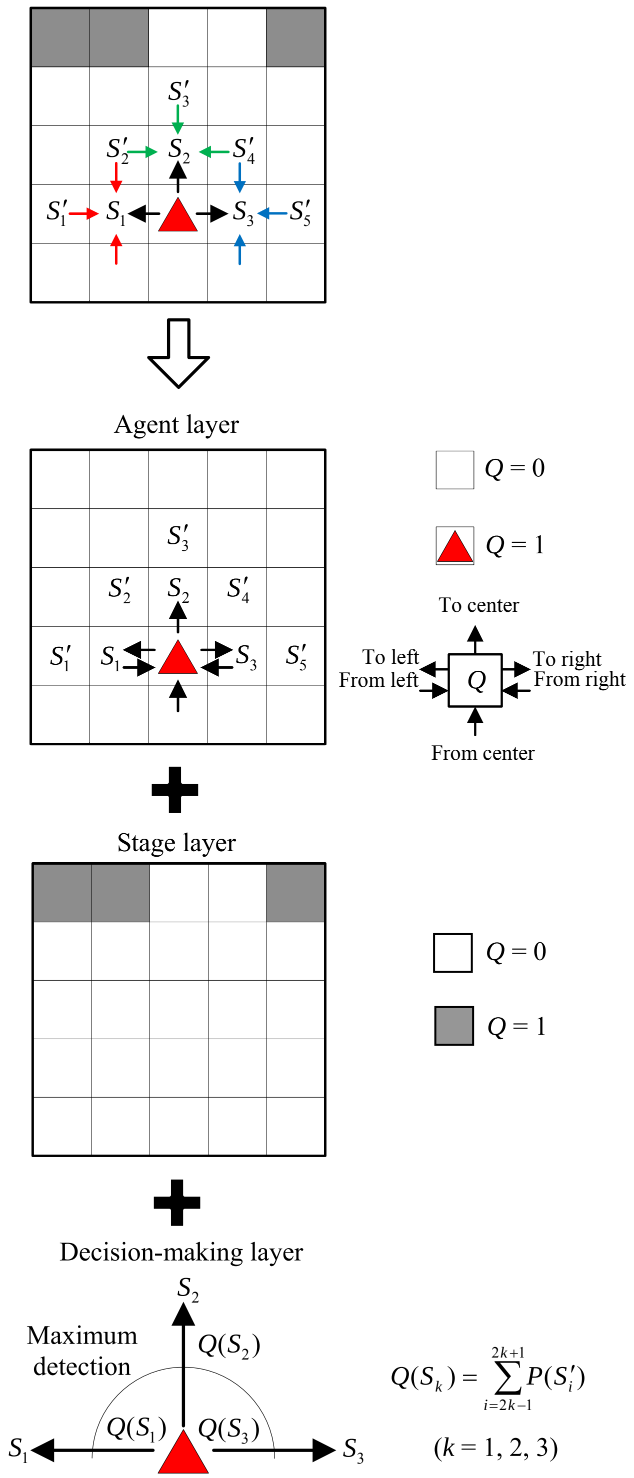

Fig.3 Architecture of the digital circuit.

This determination follows the agent’s rule. This layer works to update the value of the cell in the agent layer.

4. Implementations of the Cells in Three Layers

This section explains the circuits for the up-agents unless otherwise noted.

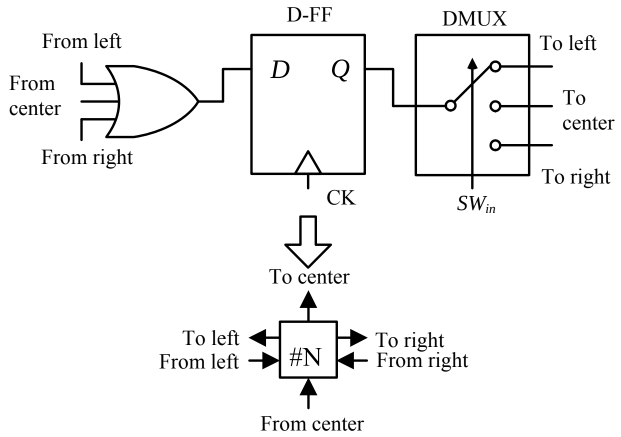

Fig.4 Implementation of a cell in the agent layer.

4.1 Agent Layer

Fig.4 shows the circuit to implement a cell in the agent layer. Three input of this circuit mean the agent's movements to this cell from the other ones, and three outputs mean them from this circuit to the other ones. In the D-FF, the output value 1 means that this cell is occupied by the agent, and zero means that it is not so. The demultiplexer (DMUX) has 1-bit 3-channels output, and it is controlled by 2-bits switching signal SW_{in}. This signal is generated by the decision making layer.

4.2 Stage Layer

In the stage layer, each cell is implemented by D-FF. This D-FF corresponds to the grid in the stage, and its value is zero if the grid is the normal grid or 1 if it is the no-entry grid.

4.3 Decision-Making Layer

A cell in the decision-making layer consists of some sub-cells. Fig.5 shows the circuit to implement the sub-cell. As shown in Fig.3, $Q(S_k)$ is given by the equation (1). According to the agent's rule, $P(S_i')$ in this equation has a negative value. Its absolute value $|P(S_i')|$ is equal to the logical sum of the values of the cells which correspond to S_i' in the agent layer and the stage layer. Each adder (⊕) calculates the following sum:

$$|Q(S_k)| = \sum_{i=2k-1}^{2k+1} |P(S_i')| \quad (k = 1, 2, 3) \quad (2)$$

$|Q(S_k)|$ is the 2-bits binary number. The block "×(−1)" adds the sign bit which means "minus" to $|Q(S_k)|$. Thus, we obtain $Q(S_1)$, $Q(S_2)$, and $Q(S_3)$ which are 2-bits with the sign bit. The maximum detection circuit finds the maximum value among them, and it outputs the agent's movement to the grid which gives the maximum value. This paragraph have explained the case where the agent has three candidates to move, while the agent faces the case of two candidates, and the case of only one candidate. We can implement the sub-cell for two candidates, in a manner similar to the case of three candidates.

The behavior of a cell in this layer depends on the number of the no-entry grids in the agent's candidates to move. We do not consider the case where all of the candidates are the no-entry grids. If there is not the no-entry grid in the candidates, the agent can move to one of all the candidates. Then, we use the sub-cell for three candidates as shown in Fig.5. If there is a no-entry grid in the candidates, the agent can move to one of two candidates. Then, we disable the wiring which corresponds to the movement from the agent's current location to the no-entry grid, and we use the sub-cell for two candidates in the decision-making layer. If there are two no-entry grids in the candidates, the agent can move to only one candidate. Then, we disable the wiring which corresponds to the movements from the agent's current location to the no-entry grid, and we wire to the candidate. Fig.6 shows a cell in this layer which integrates them.

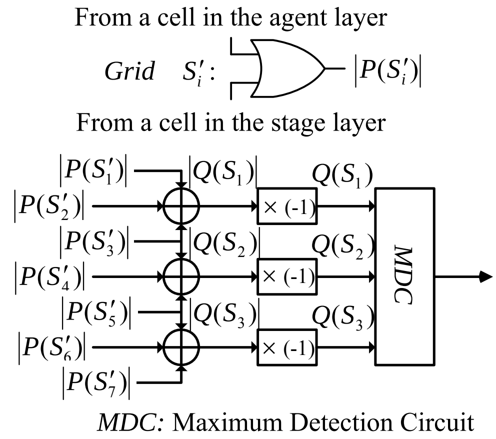

Fig.5 A sub-cell in the decision-making layer.

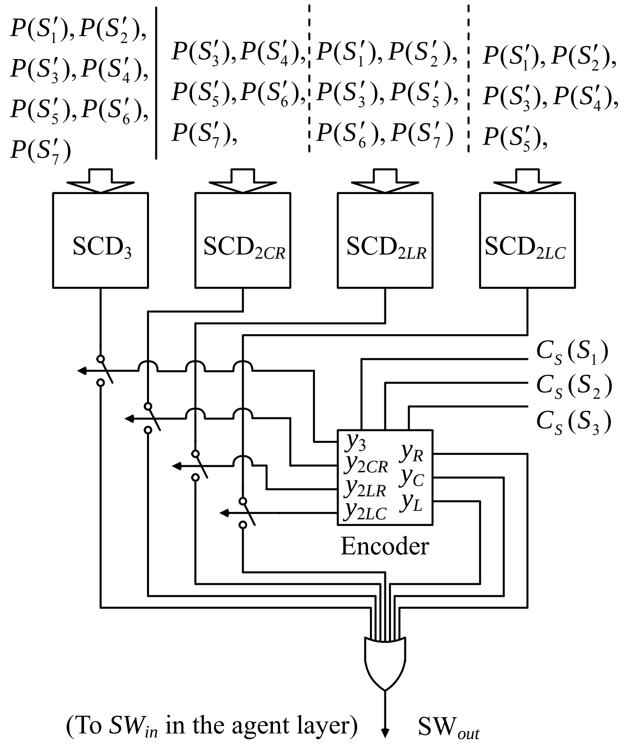

Fig.6 Implementation of a cell in the decision-making layer

"SCD" means a Sub-Cell in the Decision-making layer. SCD_3 and SCD_2 are the sub-cells for three candidates and two candidates, respectively. In the index of SCD_2 "L", "C", and "R" means the left, the center, and the right, respectively. Two of them indicate the grid where the agent is permitted to move. The encoder switches the validity or the invalidity of each sub-cell's output. Letting C_S (S_k) denote the value of the cell which corresponds to the grid S_k in the stage layer, the encoder has three inputs C_S (S_1), C_S (S_2), and C_S (S_3). In the encoder's output y_3, y_{2CR}, y_{2LR}, and y_{2LC} control the outputs of SCD_3, SCD_{2CR}, SCD_{2LR}, and SCD_{2LC}, respectively. y_R, y_C, and y_L correspond to the cases where the agent can move only to the left, the center, and the right, respectively. In these cases, the encoder's outputs directly indicate the agent's movement. One of 7-outputs in the encoder is 1 and then the others are zero in each combination of (C_S (S_1), C_S (S_2), C_S (S_3)), except that all of the outputs are zero in (1, 1, 1). The OR operation of the decision-makings is SW_{out}, and it is connected to SW_{in} of the cell in the agent layer in Fig.4.

5. Verilog-HDL Simulatuions

We verify the behavior of the designed digital circuit by the simulation based on Verilog-HDL. Fig.7 shows the timing charts of the agent layer in the simulation result. Two up-agents: up1, up2 and a down-agent are prepared, and they are initially placed at the locations shown in Fig.1. Each grid in Fig.1 is numbered. In the bottom row, the rightmost one is #0, and the leftmost one is #4. In the top row, the rightmost one is #25, and the leftmost one is #29. Fig.7 (a) and (b) are the timing charts for the up-agent and the down-agent, respectively.

The H-level in a time interval means that the grid is occupied by the agent. The number over each timing chart is the number of the grid which is occupied by the agent in the corresponding time interval. In both of (a) and (b), there is no common time interval in which the same number of the grid is shown. It means that each agent moves while it avoids the collision with the other agents.

6. Conclusions

We have designed the digital circuit of MAS for two pedestrian streams in a narrow path, and have verified its behavior by Verilog-HDL simulations. Now we are trying to implement this circuit on FPGA.

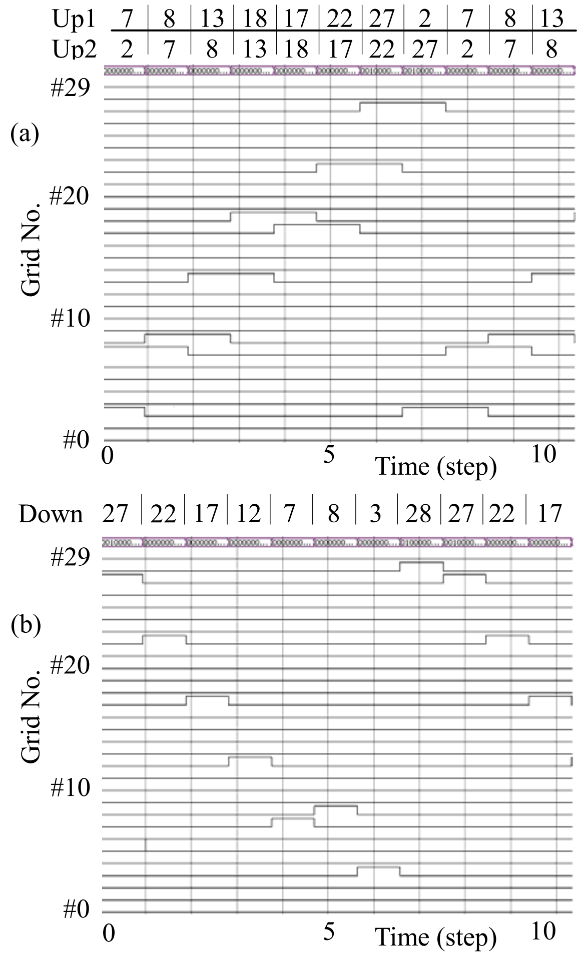

Fig.7 Timing charts of the agent layer: (a) the up-agent, and (b) the down-agent

References

[1] G. weiss, "Multiagent System", The MIT Press, 1999.

[2] X. Li, B. Jia, Z. Gao, R.Jiang, "A realistic two-lane cellular automata traffic model considering aggressive lane-changing behavior of fast vehicle", Physica A, **vol. 367**, pp. 479-486, 2006.

[3] K. Nishinari, A. Kirchner, A. Namazi, and A. Schadshneider, "Extended Floor Field Model for Evaculation Dynamics", IEICE Trans., **Vol. E87-D**, No. 3, pp.726-732, 2004.

[4] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic", J. Phys. I France 2, pp.2221– 2229, 1992.

[5] J. L. Tripp, H. S. Mortveit, A. A. Hansson, M. Gokhale, "Metropolitan Road Traffic Simulation on FPGAs", Proc. of IEEE Symposium on Field-Programmable Custom Computing Machines, pp.117-126, 2005.

[6] C. Burstedde, K. Klauck, A. Schadschneider, J. Zitlartz, "Simulation of pedestrian dynamics using a two-dimensional cellular automaton", Physica A, 295, pp.507-525, 2001.

[7] S. Morishita, S. Harada, and T. Nakano, "Flow analysis of pedestrian in a yard using cellular automata", JSME the Transportation and Logistics Conference, vol.7, pp.539-542, 1997, in Japanese.

[8] E. T. Hall, "The Hidden Dimension", Garden City, N.Y., Doubleday, 1966.

[9] D. P. Kennedy, J. Glascher, J. M. Tyszka and R. Adolphs, "Personal space regulation by the human amygdale", Nature Neuroscience, vol.12, no.10, pp.1226-1227, 2009.