# Deep Reinforcement Learning for Pedestrian Guidance

Hitoshi Shimizu[1,2], Takanori Hara[2], and Tomoharu Iwata[1]

[1] NTT Communication Science Laboratories, Kyoto, Japan
{hitoshi.shimizu.kg,tomoharu.iwata.gy}@hco.ntt.co.jp
[2] Nara Institute of Science and Technology, Nara, Japan
hara.takanori.hm8@is.naist.jp

**Abstract.** In large-scale events where many people gather, it is important to give them appropriate guidance about where to go and when to stop for visitors' efficiency and safety by easing congestion. In order to find appropriate guidance, we can evaluate guidance candidates using a pedestrian flow simulator. However, evaluating many candidates by simulation requires high computational cost, which prohibits real-time guidance. We propose a method for finding appropriate guidance in real-time for the observed situation based on deep reinforcement learning. The proposed method learns a function that outputs appropriate guidance given the observed situation. We would like to minimize the average travel time of pedestrians. However, since the pedestrian's travel time needs to track the individual, it is difficult to be measured in the real world for privacy issues. Our method uses the observed number of pedestrians moving on the roads as a reward, which can be obtained without locating the individuals, and is guaranteed by Little's law to be equivalent to minimizing the average travel time. The experimental results for unknown pedestrian flow show that the proposed method outperforms rule-based controls, and the guidance by the proposed method is as effective as the one selected from many candidates by repeated simulation with massive computational cost.

**Keywords:** Crowd simulation · Reinforcement learning · Pedestrian Guidance.

## 1 Introduction

In large-scale events where many people gather, it is important to give them appropriate guidance about where to go and when to stop for visitors' efficiency and safety by easing congestion. In order to find appropriate guidance, we can evaluate guidance candidates on a pedestrian flow simulator. Yamashita et al. [25] developed a technique for simulating all of the candidates. However, such exhaustive simulations take computational costs proportionally to the number of candidate guidances, which becomes enormous especially when the guidance is determined by a combination of multiple parameters. For searching for a better guidance with fewer simulations, Otsuka et al. [13] proposed a method using Bayesian optimization (BO). Shigenaka et al. [16] also proposed a method to search for the optimal guidance in pedestrian flow simulation using Covariance Matrix Adaptation Evolution Strategy (CMA-ES).

Although both BO and CMA-ES methods require fewer simulations than exhaustive search, many evaluations with simulators are unavoidable and prohibit real-time

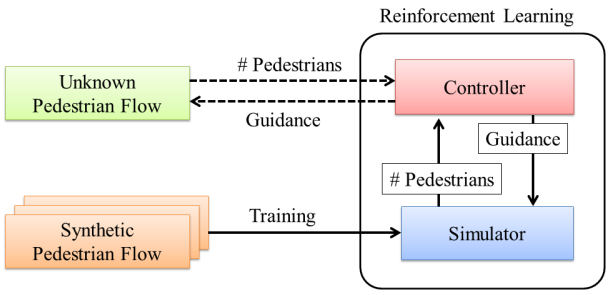


Fig. 1: Our proposed scheme for realizing pedestrian flow control using deep reinforcement learning and simulator.

guidance for unknown pedestrian flow. We propose a method for real-time guidance based on deep reinforcement learning (RL). The proposed method learns a function that output appropriate guidance given an observerved situation.

We evaluate the guidance by the average travel time of pedestrians, where the shorter average travel time is the better guidance. However, since the pedestrian's travel time needs to track the individual, it is difficult to be measured in the real world for privacy issues. Our method uses the observed number of pedestrians moving on the roads as a reward, which can be obtained without locating the individuals. The number of pedestrians is guaranteed to be equivalent to the average travel time by Little's law.

By learning with various simulated pedestrian flow data, the proposed method outputs a guidance for unknown pedestrian flow. Figure 1 shows an overview of our proposed scheme. To the best of our knowledge, this paper is the first to apply RL to control pedestrians. We experimentally demonstrate the effectiveness of our proposed method using pedestrian flow simulator. We consider an example problem selecting roads to block and encouraging detours as a guidance when the number of pedestrians moving on each road is observed as input. The main contributions of our work are as follows:

- To handle the situation in real-time, we propose a method to learn a function with deep RL that outputs appropriate guidance given the observation.
- The proposed reward based on the number of pedestrians is guaranteed to be equivalent to the average travel time by Little's law.
- Experiment results show its performance is better than a rule-based guidance policy, and close to the one selected from many candidates by repeated simulation.

## 2 Related Work

In pedestrians guidance field, there are various researches: optimal route selection for presentations [10], evacuation guidance [12], and control for the whole city [1]. Xu and González [23] argue that pedestrian flow control should adopt collective recommendations for collective benefits. Collective means that it is not sufficient to control individuals independently. For example, if all individuals move to the same destination

using the same route and means, the means of transportation may be crowded (congestion occurs on the road) and arrival at the destination may be delayed. When controlling an individual, it is necessary to fully consider the situation after the control.

Chen and Cheng [2] and Seshadri et al. [14] used multiagent simulation for this purpose. Multiagent simulation is useful because it can assume a virtual situation in advance of a large-scale event, and can evaluate the interaction of control measures. Multiple simulators can be switched according to the required area, granularity, and calculation time [9].

RL is a framework for maximizing the *reward* obtained by selecting the *action* based on the *state* observed by the agent [18]. RL has beed used in the field of transportation (for example, vehicle flow control), but not in pedestrian flow control. For vehicle flow, there are many studies about signal control [3, 6, 22]. Various indicators have been used as rewards and states [21], such as number of vehicles waiting for traffic lights, speed of vehicles, and traffic volume passing traffic lights, etc. However, when considering the theory of traffic engineering, Zheng et al. [26] showed that only number of vehicles on the lane is sufficient for states. As rewards, on the other hand, pressure [20] is shown to be more powerful than the number of vehicles waiting for traffic lights [26]. For pedestrian flow control, however, pressure is difficult to define and use as reward because pedestrian guidance is often performed outside of intersections equipped with traffic signals. Therefore, we propose a new indicator as reward for pedestrian flow control. The indicator, observed number of pedestrians moving on the roads, can be obtained without privacy issue, and is guaranteed to be equivalent to minimizing the average travel time.

## 3 Problem settings

We consider a situation where there are many people who start walking at different times from different start points to different end points via roads. The controller agent selects a guidance (action) from a set of actions at each time step. The problem is to find the sequence of the guidances that minimizes the average travel time of people $\frac{1}{I}\sum_{i=1}^{I}\tau_i$, where $\tau_i$ is travel time of pedestrian $i$ and $I$ is the number of pedestrians. The definitions of each symbol in the paper are summarized in Table 1.

## 4 Proposed method

### 4.1 Reward

The total travel time of the pedestrians is equivalent to the time integral of the number of pedestrians moving at each time. This relationship is called Little's law [8], and shown in Figure 2. The gray area $S$ enclosed by the red line indicating the cumulative number of departures and the blue line indicating the cumulative number of arrivals at each time can be expressed by two types of expressions:

$$S = \sum_{i=1}^{I}\tau_i = \int_{t=0}^{T} N_t dt \approx \sum_{t=1}^{T} N_t \Delta, \tag{1}$$

| Symbol | Description |
|---|---|
| $I$ | number of pedestrians in system: $i \in \{1, \cdots, I\}$ |
| $J$ | number of roads: $j \in \{1, \cdots, J\}$ |
| $T$ | number of time steps: $t \in \{1, \cdots, T\}$ |
| $\Delta$ | interval between adjacent time steps |
| $x_t^j$ | number of pedestrians on the road $j$ at time $t$ |
| $N_t$ | number of moving pedestrians at time $t$ |
| $v_t^i$ | velocity of pedestrian $i$ at time $t$ |
| $\rho_t^i$ | density of a road in front of pedestrian $i$ at time $t$ |
| $\rho_t^j$ | averaged density of a road $j$ at time $t$ |
| $\tau_i$ | travel time of pedestrian $i$ |

Table 1: Notation

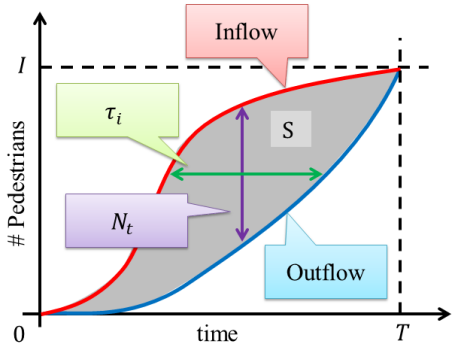


Fig. 2: Little's law: the red line represents the cumulative number of departures, and the blue line represents the cumulative number of arrivals. Because everyone who has left will arrive in a long enough time, there is a point where the red and blue lines meet, i.e. $(T, I)$. The gray area surrounded by the red and blue lines is S.

where $N_t$ is the number of moving pedestrians at time $t$, and $\Delta$ is the interval between the adjacent time steps. $\sum_{t=1}^{T} N_t \Delta$ is summation for time direction, and $\sum_{i=1}^{I} \tau_i$ is summation for each pedestrian. Approximation is acceptable when $\Delta$ is small enough for fluctuation in $N_t$. Therefore, the average travel time $\frac{1}{I}\sum_{i=1}^{I} \tau_i = \frac{S}{I}$ can be minimized by taking actions that minimize the total number of pedestrians traveling at each time $\sum_{t=1}^{T} N_t = \frac{S}{\Delta}$ because $I$ and $\Delta$ are constants.

In addition, if the absolute values of rewards vary in wide range, it is difficult to adjust other parameters in RL. Therefore, it is important to normalize rewards, for ex-


Fig. 3: Road network around the National Stadium in Tokyo, Japan. Numbers (1 to 6) represent stations, and alphabets (A to F) represent stadium gates.

ample, into the range of -1 to 1[3]. Thus we propose the following reward:

$$r_t = \begin{cases} \max\left(-1, \frac{N_t^{\mathrm{o}} - N_t}{N_t^{\mathrm{o}}}\right) & N_t^{\mathrm{o}} > 0 \\ 0 & N_t^{\mathrm{o}} = 0 \text{ and } N_t = 0 \\ -1 & N_t^{\mathrm{o}} = 0 \text{ and } N_t > 0, \end{cases} \tag{2}$$

where $N_t^{\mathrm{o}}$ is the total number of pedestrians on the roads when all gates are always open. This reward satisfies $-1 \leq r_t \leq 1$, and $r_t = 1$ when $N_t = 0$, and $r_t = 0$ when $N_t = N_t^{\mathrm{o}}$ if $N_t^{\mathrm{o}} > 0$.

### 4.2 State

States can be set independently of reward. However, if the number of pedestrians is observed for reward, it is considered more convenient and efficient to use the observation as state. For the measurement of the number of pedestrians, just measuring the total number of pedestrians does not tell where the congestion is occurring. Also, observing the number of people only at one time step does not tell whether it is increasing or decreasing. For example, therefore, we can use the number of pedestrians on each road of multiple time steps as state.

## 5 Experiments

We evaluated the proposed method on the task of finding guidance as an example to ease congenstion around the entrance gate at the start of a big event. Figure 3 shows the

[3] https://github.com/Unity-Technologies/ml-agents/blob/master/docs/Learning-Environment-Best-Practices.md

| station ID | usage ratio of pedestrians |
|---|---|
| 1 | 29% |
| 2 | 11% |
| 3 | 6% |
| 4 | 11% |
| 5 | 20% |
| 6 | 22% |

| gate ID | throughput [person / sec] |
|---|---|
| A | 3 |
| B | 8 |
| C | 3 |
| D | 3 |
| E | 5 |
| F | 3 |

Table 2: Left: ratio of pedestrians emerging from each station. Right: the maximum number of people that can pass in each second at each gate.

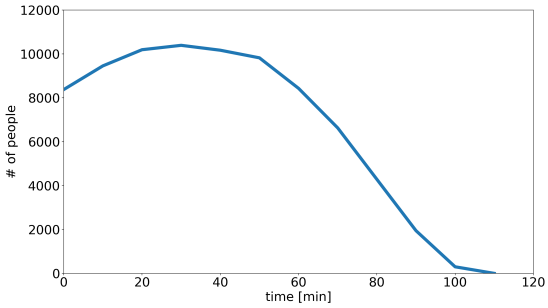


Fig. 4: Distribution of time when pedestrians start moving. The horizontal axis is the elapsed time in minutes from the start of the simulation. The vertical axis is the number of pedestrians who start moving in every 10 minutes for $I = 80000$.

road network around the National Stadium in Tokyo, Japan, which is the stage of the simulation. Pedestrians start to walk from six stations to the six gates of the stadium, and are crowded at the entrance gate of the stadium. There are 317 roads where pedestrians pass through. For state, we used the number of pedestrians on these roads for the past four steps, which gives a 1268-dimensional vector.

The number of pedestrians in one scenario varies from 10,000 to 90,000 by 10,000. In each scenario, the proportion of stations where pedestrians appear was varied using random numbers from the Dirichlet distribution. The expected value was set to be the ratio of Table 2 (left) by refering to the actual number of station users.

The timing of pedestrians appearing from the station was distributed as shown in Figure 4. At the entrance of the stadium, assuming that the number of security checking staff varies depending on the gate, the maximum numbers of people who pass the gate per second were set as in Table 2 (right).

We used the state-of-the-art RL method called Advantage Actor-Critic (A2C) [11, 24] as a learning model. It learns based on the experience during the episode after every episode is completed. The value function was approximated by a neural network with

two hidden layers, each of which has 100 units. We used the ReLU function [4] to make each layer output nonlinear.

### 5.1 Guidance actions

In order to avoid the congestion of the gate, we consider a guidance of temporarily closing the gate. When the gate is closed, it is assumed that pedestrians head to the gate that is not closed. Since there are six gates, there are $2^6 = 64$ combinations of opening and closing. However, we added a constraint that three or more adjacent gates are not closed simultaneously so as not to make the detour route too long. Then, we have $39$ guidance candidates. Guidance last at least $10$ minutes, and different guidance can be selected every $10$ minutes. The simulation time is set to $250$ minutes so that all pedestrians could enter the stadium no matter what guidances were performed. Therefore, guidance is selected $25$ times per episode.

### 5.2 Pedestrian model in simulation

The guidance is evaluated through multiagent simulation according to the following pedestrian model. Each pedestrian is given the start and end nodes, the time to start walking and the maximum walking speed $v_{\max}$. Pedestrians moving from the station to the stadium select the nearest gate and shortest route when there is no guidance. However, when the gate is closed, the pedestrian will head to the nearest gate that is not closed by the shortest route. The maximum speed $v_{\max}$ for each pedestrian was determined to follow a normal distribution with an average of $1.2$ meter / sec and a standard deviation of $0.2$. In multiagent simulation, it calculates which position on which road various pedestrians are moving at each time step. In this case, a speed reduction model according to the following formula is used according to the population density calculated based on the width of the road on which a pedestrian is passing and the position of other pedestrians. When the population density in the area of 6 meter ahead of a pedestrian is $\rho_t^i$, the speed of the pedestrian $v_t^i$ is

$$v_t^i = \begin{cases} v_i^{\max} & (0 \leq \rho_t^i < \frac{1.8}{v_i^{\max}+0.3}) \\ \frac{1.8}{\rho_t^i} - 0.3 & (\frac{1.8}{v_i^{\max}+0.3} \leq \rho_t^i < 6) \\ 0 & (\rho_t^i \geq 6). \end{cases} \tag{3}$$

This simulator updates the agent's position sequentially. The time step is 1 second on our implementation, and the pedestrian's speed is updated by Eq. (3) at each time step. Therefore, if a large number of pedestrians try to pass a common road at the same time, congestion will occur and the moving speed will be slow. In order to avoid congestion, it is effective to limit the number of pedestrians flowing into the road where congestion is likely to occur. This simulator completes one episode of a scenario in this paper within about one minute. An example of visualization of the simulation is shown in Figure 7.

| name | reward |
|---|---|
| EDGE/OPEN | proposed at Eq. (2) |
| EDGE | $\frac{I-N_t}{I}$ |
| GOAL | $\frac{1}{I}\sum_i^I \mathbb{1}((t-1)\Delta < \tau_i \leq t\Delta)$ |
| GOALCUM | $\frac{1}{I}\sum_i^I \mathbb{1}(\tau_i \leq t\Delta)$ |
| SPEED | $\frac{\bar{v}^{\max}-v_t}{\bar{v}^{\max}}$, |
| | where $\bar{v}^{\max} = \frac{1}{I}\sum_i^I v_i^{\max}$ |
| | and $v_t = \frac{1}{N_t}\sum_j x_t^j \times v(\rho_j)$ |
| TIME/OPEN | $\sum_i \frac{\tau_i^{\mathrm{o}}-\tau_i}{\tau_i^{\mathrm{o}}} \mathbb{1}((t-1)\Delta < \tau_i \leq t\Delta)$ |
| TIMEONCE/OPEN | $\frac{\sum_i \tau_i^{\mathrm{o}}-\tau_i}{\sum_i \tau_i^{\mathrm{o}}} \quad (t=T)$ |
| | $0 \quad (t \neq T)$ |
| TIMEONCE | $\frac{-\sum_i \tau_i}{TI} \quad (t=T)$ |
| | $0 \quad (t \neq T)$ |

Table 3: Rewards for deep RL.

### 5.3 Comparing Rewards

We prepared the rewards shown in Table 3 as comparing methods, referring to the study of RL in traffic signal control. EDGE/OPEN is the proposed method, and is the value obtained by normalizing the total number of observed people with OPEN policy. EDGE is the value obtained from the total number of observed people by the number of all pedestrians without normalization of OPEN. GOAL is the value obtained by normalizing the number of people arriving at the gates from the previous step to the current step with the number of all pedestrians. GOALCUM is the value obtained by normalizing the number of people arriving at the gates from the start of the simulation to the current step by the number of all pedestrians. SPEED is the average speed of pedestrians calculated based on the population density of each road, normalized with the maximum speed. TIME/OPEN is the value obtained by normalizing the travel time of the pedestrian who arrived at the gates from the previous step to the current step with the travel time of OPEN. TIMEONCE/OPEN is the value obtained by normalizing the travel time of the pedestrians with the travel time of OPEN at the end of the episode. TIMEONCE is the value obtained from the travel time of the pedestrians without normalization of OPEN at the end of the episode. Note that the rewards named with /OPEN use the result of OPEN for normalizing.

### 5.4 Comparing methods

We compared the proposed method with OPEN as the baseline, where all gates are always open and no guidance is applied. We also prepared a rule-based guidance shown as RULE, where all gates are open if the population densities (number of people / road area) of all roads in front of the gates are less than a threshold, and the gate with the highest density road is closed if there is a road above the threshold. The threshold was set to 1.0 person / square meter.

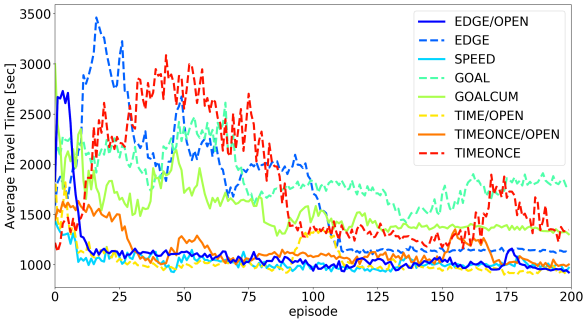


Fig. 5: The horizontal axis is the number of episodes. The vertical axis is the average travel time.

GREEDY shows the results of the guidance obtained by repeated simulations for comparison. With 25 time steps and 39 actions, there are $39^{25} \sim 10^{40}$ combinations of guidances. Since the computation time required to execute all combinations of simulations is too long, GREEDY uses the following procedure. (1) Set all time steps to open as candidate guidances. (2) Run a simulation with 39 different guidances for a certain time step. (3) The guidance with the best evaluation value among the 39 actions is updated as a candidate. (4) Repeat steps (2) and (3) for all time steps. (5) When the procedure is completed for all time steps, output the candidate policy.

## 6 Results

### 6.1 Comparing Reward

Figure 5 shows the average travel time for each episode when training with rewards shown in Table 3. We used 16 scenarios for training, which consist of eight types of number of pedestrians, ranging from 10,000 to 80,000, each with two different station use ratios. The number of simulations performed for training is 200 pisodes $\times$ 16 scenarios in total, 3200 times for each deep RL. In 200 episodes, the evaluation value of EDGE/OPEN, SPEED, TIME/OPEN and TIMEONCE/OPEN are stable and smaller than other rewards.

### 6.2 Performance for Unknown Pedestrian Flow

Table 4 shows the result of applying guidances for pedestrian flow not included in the training data. We created 90 test scenarios, which consist of ten scenarios for each 10,000 pedestrians from 10,000 to 90,000. FIX is the result of randomly selecting the guidance policy obtained by GREEDY for each scenario, regardless of the actual scenario. Although the average travel time of FIX was similar to that of RULE, its effect was not as good as GREEDY. Note that GREEDY and FIX methods need iterative evaluation ($39 \times 25 = 975$ times of simulations) for the target scinario. This results took about 25 minutes to execute 39 parallel simulations 25 times.

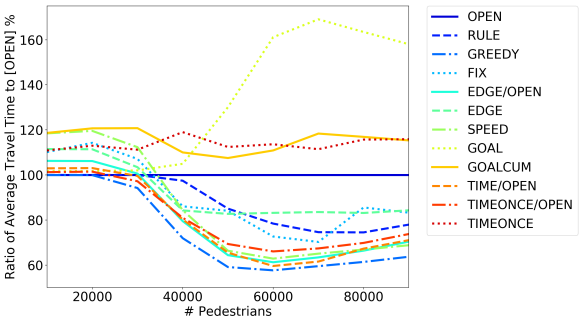


Fig. 6: The horizontal axis is the number of pedestrians. The vertical axis is ratio of average travel time to OPEN.

TIME/OPEN and TIMEONCE/OPEN give almost the best results in RL for all $I$. However, they are considered difficult to use due to privacy issues. SPEED also gives good results when $I$ is large, but its performance is poor when $I$ is small. This method increases the moving speed by detouring, which may take extra travel time. Therefore, the proposed EDGE/OPEN yields the best result as the RL reward. The time required for the method to make a decision was about 5 milliseconds each time, which satisfies the demand for real-time use.

Figure 7 shows simulations with 80,000 pedestrians in OPEN and EDGE/OPEN. At 40 minutes after the start, the pedestrian does not select gate D in OPEN, but EDGE/OPEN guides the pedestrian to gate D by closing other gates. At 80 minutes, EDGE/OPEN has queues at five gates with a better balance than OPEN. At 120 minutes, while OPEN has a long queue at gate A, most pedestrians of EDGE/OPEN have entered the stadium.

## 7 Discussion

Little's law holds even for a single pedestrian. The tasks of minimizing the time for a moving object to reach its goal has been frequently addressed in the history of reinforcement learning [18]. A small negative reward to each step usually results in the policy of arriving at the goal in the shortest time [1]. Little's law discussed in this paper makes it clear that a negative reward to each step leads to the shortest travel time, and such explanation has not been given so far. Our proposed method will be useful for the tasks of making a moving object reach its goal in the shortest time.

Data assimilation technology [7, 15, 17] that reproduces pedestrian flow measured on the roads by simulation has been developed. Using such data assimilation techniques, we can use a realistic simulation by compensating for missing observations. Combined with data assimilation technology, our proposed method will be an important element of a system to avoid congestion by real-time guidance [19].

| | method | Ratio to OPEN % |
|---|---|---|
| | RULE | 87.5 |
| | FIX | 90.4 |
| RL: reward | EDGE/OPEN (proposed) | **79.8** |
| | EDGE | 91.9 |
| | SPEED | 85.0 |
| | GOAL | 132.5 |
| | GOALCUM | 115.5 |
| | TIME/OPEN | **79.0** |
| | TIMEONCE/OPEN | 80.8 |
| | TIMEONCE | 113.7 |
| Ref. | GREEDY | 74.1 |

Table 4: Average ratio of travel time to OPEN for each method for 90 scenarios. Ref.(GREEDY) represents reference methods for comparison. OPEN took 1493.2 [sec] on average. Bold indicates results that are not significantly different from the best result (TIME/OPEN) except for GREEDY in paired t-test ($p < 0.05$).

## 8 Conclusion

In this paper, we proposed a method to find efficient guidance by learning using deep RL and pedestrian flow simulator. The evaluation experiment on the simulation data showed that the proposed method finds a better guidance than a rule-based control, and its performance is close to the one selected from many candidates by repeated simulation with massive computational cost.

Our proposed method learns only for the fixed road network, and we assume that the number of pedestrians on all roads are observed in our experimental settings. However, the target road network is not always constant, and it is not expected in practice to obtain the number of pedestrians on all roads. Therefore, we would like to improve our method to handle unknown roads using Graph Convolutional Networks (GCN) [5], and to utilize a limited number of road observations as a future work.

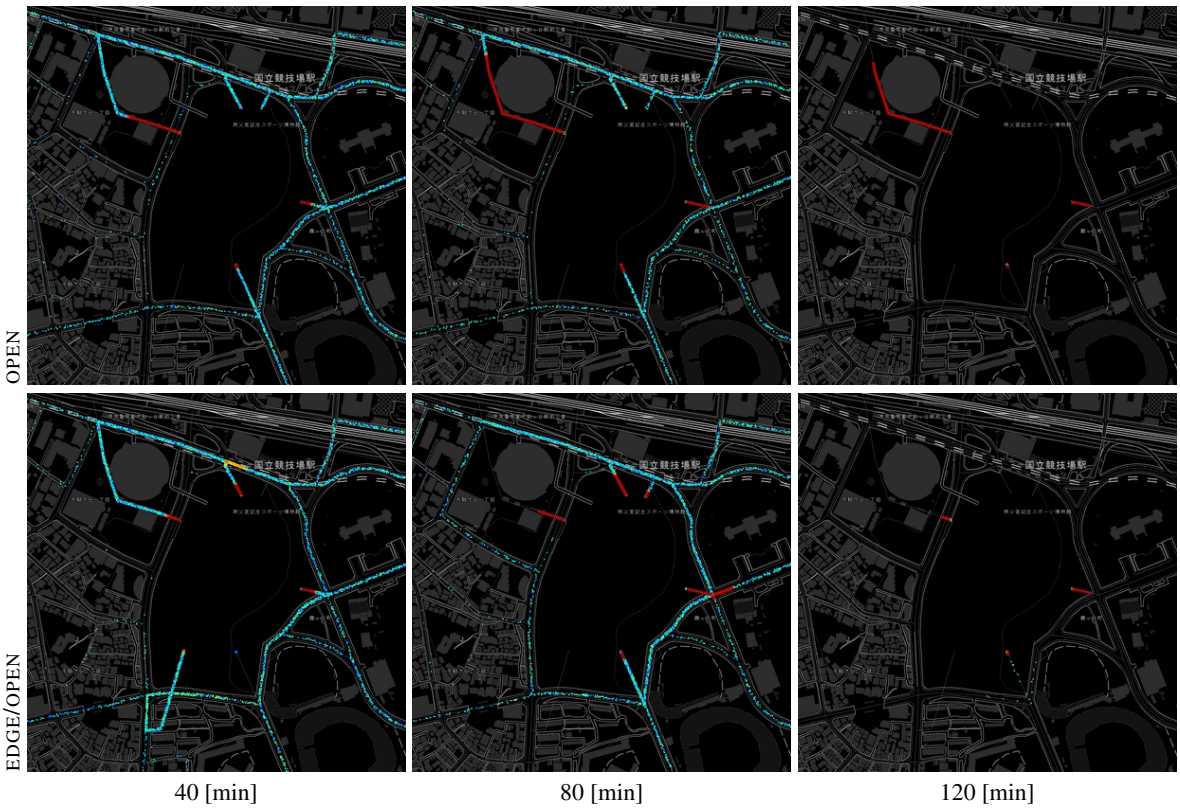


Fig. 7: $I = 80000$. Average travel times of OPEN and EDGE/OPEN were 2481.0 and 1658.3 [sec], respectively. Color of dots represents speed of the pedestrian: blue is fast and red is slow. The red lines in front of the gates are the pedestrian queues waiting for entry.

# Bibliography


[1] Almeida, J.E., Rosseti, R.J., Coelho, A.L.: Crowd simulation modeling applied to emergency and evacuation simulations using multi-agent systems. arXiv preprint arXiv:1303.4692 (2013)

[2] Chen, B., Cheng, H.H.: A review of the applications of agent technology in traffic and transportation systems. IEEE Transactions on intelligent transportation systems **11**(2), 485–497 (2010)

[3] Dusparic, I., Monteil, J., Cahill, V.: Towards autonomic urban traffic control with collaborative multi-policy reinforcement learning. In: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 2065–2070, IEEE (2016)

[4] Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics, pp. 315–323 (2011)

[5] Iwata, T., Otsuka, T., Shimizu, H., Sawada, H., Naya, F., Ueda, N.: Finding appropriate traffic regulations via graph convolutional networks. arXiv preprint arXiv:1810.09712 (2018)

[6] Khamis, M.A., Gomaa, W.: Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework. Engineering Applications of Artificial Intelligence **29**, 134–151 (2014)

[7] Kiyotake, H., Kohjima, M., Matsubayashi, T., Toda, H.: Multi agent flow estimation based on bayesian optimization with time delay and low dimensional parameter conversion. In: International Conference on Principles and Practice of Multi-Agent Systems, pp. 53–69, Springer (2018)

[8] Little, J.D., Graves, S.C.: Little's law. In: Building intuition, pp. 81–100, Springer (2008)

[9] Mario, M., DellOrco, M., Ottomanelli, M.: Pedestrian evacuation management of large areas: a bi-level simulation approach based on fuzzy logic. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 190–195, IEEE (2015)

[10] May, A.J., Ross, T., Bayer, S.H., Tarkiainen, M.J.: Pedestrian navigation aids: information requirements and design implications. Personal and Ubiquitous Computing **7**(6), 331–338 (2003)

[11] Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: International conference on machine learning, pp. 1928–1937 (2016)

[12] Murakami, Y., Minami, K., Kawasoe, T., Ishida, T.: Multi-agent simulation for crisis management. In: Proceedings. IEEE Workshop on Knowledge Media Networking, pp. 135–139, IEEE (2002)

[13] Otsuka, T., Shimizu, H., Iwata, T., Naya, F., Sawada, H., Ueda, N.: Bayesian optimization for crowd traffic control using multi-agent simulation. In: 2019 22st

International Conference on Intelligent Transportation Systems (ITSC), IEEE (2019)
[14] Seshadri, M., Cao, Z., Guo, H., Zhang, J., Fastenrath, U.: Multiagent-based cooperative vehicle routing using node pressure and auctions. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–7, IEEE (2017)
[15] Shigenaka, S., Takami, S., Onishi, M., Yamashita, T., Noda, I.: Estimating pedestrian flow in crowded situations with data assimilation. In: 10th International Workshop on Optimization in Multiagent Systems (OptMAS) (2019)
[16] Shigenaka, S., Takami, S., Ozaki, Y., Onishi, M., Yamashita, T., Noda, I.: Evaluation of optimization for pedestrian route guidance in real-world crowded scene. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, pp. 2192–2194, International Foundation for Autonomous Agents and Multiagent Systems (2019)
[17] Shimizu, H., Matsubayashi, T., Tanaka, Y., Iwata, T., Ueda, N., Sawada, H.: Improving route traffic estimation by considering staying population. In: International Conference on Principles and Practice of Multi-Agent Systems, pp. 630–637, Springer (2018)
[18] Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
[19] Ueda, N., Naya, F., Shimizu, H., Iwata, T., Okawa, M., Sawada, H.: Real-time and proactive navigation via spatio-temporal prediction. In: Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers, pp. 1559–1566, ACM (2015)
[20] Wei, H., Chen, C., Zheng, G., Wu, K., Gayah, V., Xu, K., Li, Z.: Presslight: Learning max pressure control to coordinate traffic signals in arterial network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1290–1298, ACM (2019)
[21] Wei, H., Zheng, G., Gayah, V., Li, Z.: A survey on traffic signal control methods. arXiv preprint arXiv:1904.08117 (2019)
[22] Wiering, M.: Multi-agent reinforcement learning for traffic light control. In: Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000), pp. 1151–1158 (2000)
[23] Xu, Y., González, M.C.: Collective benefits in traffic during mega events via the use of information technologies. Journal of The Royal Society Interface **14**(129), 20161041 (2017)
[24] Xu, Z., van Hasselt, H.P., Silver, D.: Meta-gradient reinforcement learning. In: Advances in neural information processing systems, pp. 2396–2407 (2018)
[25] Yamashita, T., Okada, T., Noda, I.: Implementation of simulation environment for exhaustive analysis of huge-scale pedestrian flow. SICE Journal of Control, Measurement, and System Integration **6**(2), 137–146 (2013), https://doi.org/10.9746/jcmsi.6.137
[26] Zheng, G., Zang, X., Xu, N., Wei, H., Yu, Z., Gayah, V., Xu, K., Li, Z.: Diagnosing reinforcement learning for traffic signal control. arXiv preprint arXiv:1905.04716 (2019)