Methods for Improving Efficiency of Queuing Systems

D. Yanagisawa[1,2], Y. Suma[1], Y. Tanaka[1], A. Tomoeda[3,4], K. Ohtsuka[4], and K. Nishinari[4,5,1]

[1]Department of Aeronautics and Astronautics, School of Engineering, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan.
[2]Research Fellow of the Japan Society for the Promotion of Science, Kojimachi, Chiyoda-ku, Tokyo 102-8471, Japan.
[3]Meiji Institute for Advanced Study of Mathematical Sciences, Meiji University, Tama-ku, Kawasaki-city, Kanagawa, 214-8571, Japan.
[4]Research Center for Advanced Science and Technology, The University of Tokyo, Komaba, Meguro-ku, Tokyo, 153-8904, Japan.
[5]PRESTO, Japan Science and Technology Agency, Sanbanmachi, Chiyoda-ku, Tokyo 102-0075, Japan.

Corresponding Author: tt087068@mail.ecc.u-tokyo.ac.jp

Abstract We have considered the methods for improving efficiency in queuing systems by theoretical analysis and experiments. First, a queuing system which has plural service windows is studied. There are mainly two kinds of systems which are a parallel-type queuing system and a fork-type queuing system. Queuing theory is often used to analyze these queuing systems; however, it does not include the effect of walking distance from the head of the queue to service windows; thus, a walking-distance introduced queuing theory is investigated. By using this model, we have discovered that the suitable type of system changes according to the utilization of the system. We have also verified that when we keep one person waiting at each service window in the fork-type queuing system, the waiting time dramatically decreases. Secondly, we consider queuing systems in amusement parks. Plural people waiting in the queue move to get on a roller coaster at the same time; therefore, the efficiency of the system is improved by shortening the moving time. The result of the experiments indicates that the moving time decreases if people keep walking in the queue to start instantaneously.

Introduction

Pedestrian dynamics has been studied vigorously over last decades and many successful models have been developed [1,2]. There are two major topics focused on by many researches in pedestrian dynamics. One is one dimensional flow [3],

R.D. Peacock et al. (eds.), *Pedestrian and Evacuation Dynamics*,
DOI 10.1007/978-1-4419-9725-8_27,

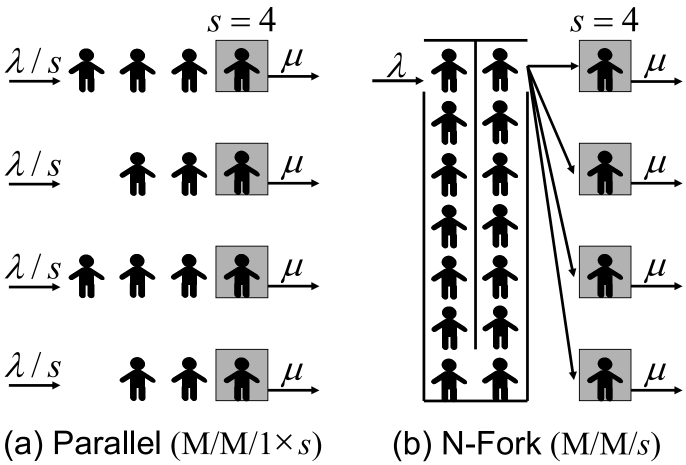


Fig. 1. Schematic views of queuing systems in the case that the number of service windows $s = 4$. (a) Parallel (M/M/1×4). (b) N-Fork (M/M/4), which is an ideal fork-type queuing system. A person at the head of the queue moves to the service windows instantaneously when one of them becomes vacant.

which is very important to investigate the basic features of pedestrian flow, and the other is an evacuation [4-6], which is an essential theme since it is strongly connected with people's life in an emergency situation.

A queue is one of the typical phenomenon, which is observed everywhere in the cities, for instance, at a supermarket, a bank, a concert hall, and so on. Thus, research on a queuing system is one of an important topic in pedestrian dynamics; however, there are not as many researches on it as those of one dimensional flow and evacuation since queuing dynamics is usually studied by using queuing theory [7].

Although the queuing theory is very successful and has been used to study queuing systems, it does not consider the effect of walking distance in the system, which is a very important factor to study pedestrian queuing systems. Thus, we combine the queuing theory and the cellular automaton to obtain a realistic model which includes the effect of walking distance. Two kinds of advanced queuing systems are studied in this paper. In the following sections, the recipes for improving efficiency of queuing systems with plural service windows (in Sec. 2) and in amusement parks (in Sec. 3) are proposed.

Queuing System with Plural Service Windows

There are mainly two kinds of queuing systems with plural service windows, which are a *parallel-type* queuing system (Parallel), i.e., queues for each service

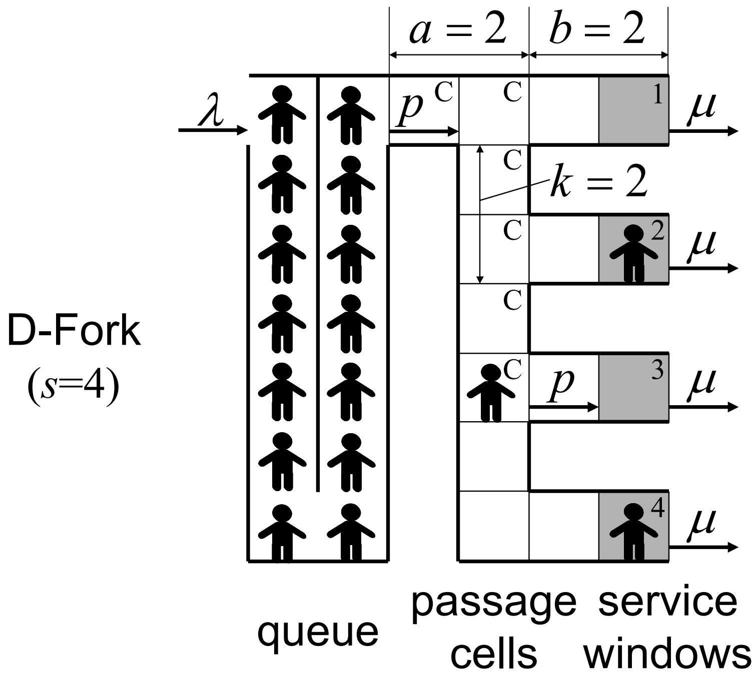


Fig. 2. **Schematic view of D-Fork (s = 4)**

windows, and a *fork-type* queuing system (Fork), which collects people into a single queue. According to the queuing theory, the waiting time in Fork is always shorter than that in Parallel (Fig. 1 (a)). However, Fork considered in the normal queuing theory (N-Fork) (Fig. 1 (b)) does not reflect the effect of the walking distances from the head of the queue to the service windows. The effect of the distance may be possible to neglect in a small queuing system; however, it should not be ignored in a large one such as an immigration inspection floor in an international airport. Thus, we have introduced the effect of walking-distance to the queuing theory as follows.

Distance introduced Fork-type Queuing System: D-Fork

In Parallel, people wait just behind the former person, so that there is no delay in walking. While, in N-Fork, people take some time to walk from the head of the queue to the service windows; therefore, we consider D-Fork as in Fig. 2 by representing the walking distance using cellular automaton. The gray cells are window cells, and the numbers described in them are window numbers. The white cells are passage cells. Note that the letter “C” described in the passage cells represents the common passage cells. For example, both persons who are going to the window 3 and 4 pass them. People sometimes cannot go forward in the common passage cells since there is a possibility that other people stand in front of them. The place that people are waiting, which is not divided into cells, is a queue. $s \in \mathbf{N}$, $\lambda \in [0,\infty)$, and $\mu \in [0,\infty)$ represent the number of service windows, the arrival rate, and the service rate, respectively. a and b represent the length of the passage, and k is the interval length between two service windows. The distance from the head of the queue to the service window $n \in [1,s]$, is described as $d_n = a + b + k(n-1)$. Fig. 2 represents the case s=4, a=2, b=2, and k=2. Service windows have two states: vacant and occupied. When a person at the head of the queue decides to move to the vacant service window n, it changes into occupied state. The person proceeds to the service window by one cell with the rate $p \in [0,\infty)$ as the asymmetric simple exclusion process [8]. A service starts when the person arrives at the service window, and after it finishes the state of the service window changes into vacant state.

Update Rules

The simulation of D-Fork consists of the following five steps per unit time step.

1. If there is at least one vacant service window and one person in the queue, and the first cell of the passage is vacant, then the person decide to proceed to a vacant service window which is the nearest to the head of the queue, and the state of the service window becomes occupied.
2. Add one person to the queue with the probability $\lambda\Delta t$, where Δt is the length of the unit time step.
3. Proceed each person in the passage cells to his/her service windows with the probability $p\Delta t$ if there is not other person at their proceeding cell.
4. Remove people at the service windows and change their states into vacant state with the probability $\mu\Delta t$.
5. If 1. takes into practice, proceed the person at the head of the queue to the first passage cell with the probability $p\Delta t$.

Stationary Equations

We define the sum of the walking time and the service time at service window *n* as a throughput time τ_n and its reciprocal as a throughput rate μ_n. Here, we calculate the mean throughput rate $\hat{\mu}_n$ when *n* service windows are occupied, and obtain stationary equations of D-Fork. We suppose that all passage cells are vacant by mean field approximation. Then, the mean value of the throughput time $E(\tau_n)$ is described as follows.

$$E(\tau_n) = \frac{1}{\mu} + \frac{a+b+k(n-1)}{p}. \tag{1}$$

The throughput rate μ_n is obtained as

$$\mu_n = \frac{1}{E(\tau_n)} = \frac{1}{\frac{1}{\mu} + \frac{a+b}{p} + \frac{k(n-1)}{p}} = \frac{\mu}{1+\alpha+2\beta(n-1)}, \tag{2}$$

where

$$\alpha = \frac{\mu(a+b)}{p}, \qquad \beta = \frac{k\mu}{2p}. \tag{3}$$

In the case $2\beta(n-1)/(1+\alpha) \ll 1$, we calculate the mean throughput rate $\hat{\mu}_n$ as

$$\hat{\mu}_n = \frac{1}{n}\sum_{l}^{n} \mu_l \approx \frac{\mu}{1+\alpha+\beta(n-1)}. \tag{4}$$

By using (4) the stationary equations are described as follows:

$$\begin{aligned} &\lambda P_0 = \hat{\mu}_1 P_1 \\ &\lambda P_{n-1} + (n+1)\hat{\mu}_{n+1} P_{n+1} = (\lambda + n\hat{\mu}_n) P_n \qquad (1 \le n \le s-1) \\ &\lambda P_{n-1} + s\hat{\mu}_s P_{n+1} = (\lambda + s\hat{\mu}_s) P_n \qquad (n \ge s). \end{aligned} \tag{5}$$

We obtain the mean waiting time W_q by solving (5) analytically. In the case $\alpha=\beta=0$, we have the stationary equations of M/M/s [7] from (5), thus α and β represent the effect of walking time.

In our simulation the distribution of the throughput time is gamma distribution. We approximate it as exponential distribution in this calculation, however, when β

is small the results from the exponential distribution approximated well to those from gamma distribution.

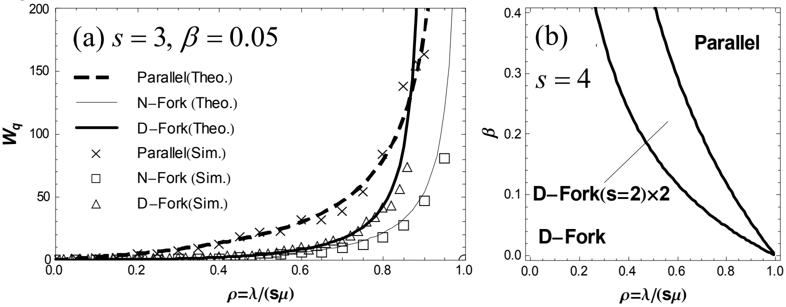


Fig. 3. (a) Comparison of W_q of Parallel, N-Fork, and D-Fork in the case α=0, β=0.05, μ=0.05. (b) Queuing system which makes the mean waiting time W_q minimum in the case s=4.

Comparison between Parallel Queue and Fork Queue

We compare the mean waiting time W_q of Parallel (Fig. 1 (a)), N-Fork (Fig. 1 (b)), and D-Fork (Fig. 2 (a)). Figure 3 (a) shows W_q against the utilization $\rho = \lambda/(s\mu)$. The results of analysis agree with those of the simulation very well. We see that W_q of N-Fork is smaller than that of Parallel and D-Fork in $0 \leq \rho < 1$. There is a possibility that more than one person is waiting in one queue and no one is in the other queue in Parallel $(s \geq 2)$, however there is no vacant service window in N-Fork when people are waiting in the system. This is the reason why W_q of N-Fork is always smaller than that of Parallel. Since N-Fork does not take into account of the effect of the walking distances, i.e., $\alpha = \beta = 0$, it is obvious that W_q of N-Fork is smaller than that of D-Fork. The N-Fork is the most efficient of the three; however, it is an ideal system and does not exist in reality. By focusing on the curves of Parallel and D-Fork, we can clearly observe the crossing of them. This means that when the utilization ρ is small, i.e., there are not sufficiently many people in the system; we should form D-Fork to decrease the waiting time. On the contrary, when the utilization ρ is large, i.e., there are many people in the system, we should form Parallel. When β become large, the crossing point moves to the left. The strong effect of the walking distances extends the suitable region for Parallel. This agrees with our intuition, since D-Fork is influenced by the distances but Parallel does not. The reversal phenomenon of W_q is obtained for the first time by introducing the effect of distance.

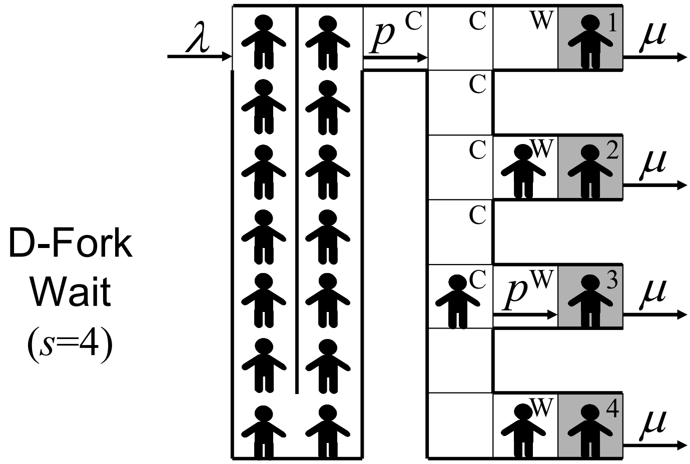
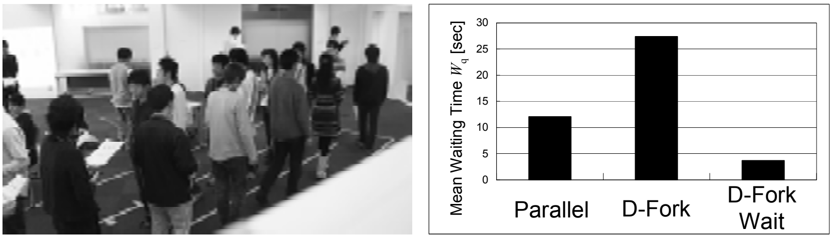


Fig. 4. Schematic view of D-Fork-Wait. People can wait at the cells, which are described as "W".

Figure 3 (b) shows the type of queuing system, which minimize W_q against ρ and β in the case s=4. This figure is useful for designing queuing systems. The curves divide the plane into three regions. In the lower left region W_q of D-Fork (s=4) is the smallest, and in the upper right region W_q of Parallel is the smallest. Surprisingly, W_q of D-Fork (s=2) $\times$ 2 is the smallest in the middle region. This indicates that the choice of the type of queuing systems is not only Parallel and D-Fork, but also a combination of them. According to (3), β represents the ratio of walking time and service time. Therefore, D-Fork is suitable when service time is much longer than walking time. The value of β is small in most D-Fork in reality, however, in large queuing system such as an immigration inspection floor in an international airport, we should divide the large D-Fork into the several small D-Forks to decrease the effect of the walking distance.

Keep One Person Waiting at the Windows: D-Fork-Wait

The walking distance in D-Fork is essentially problematic, since it delays the start of services, i.e., people have to walk the passage before they start to receive the service. Thus, we propose to keep one person waiting at the service windows. We call a queuing system which this method is applied to as D-Fork-Wait (Fig. 4). Since people are waiting just next to the service windows, they can receive service instantaneously when their former people leave there. The delay in walking is almost removed by this method, i.e., the effect of walking distance does not need to be considered. Therefore, the mean waiting time is approximately calculated by the mathematical expression for N-Fork.



(a) (b)

Fig. 5. (a) Schematic view of the experiment. (b) The experimental mean waiting time for Parallel, D-Fork, and D-Fork-Wait.

Experiments

We have performed the experiments to examine the two results in the former section:

1. There is a case that W_q in Parallel becomes smaller than that in D-Fork.
2. We can decrease W_q by keeping one person waiting at each service window.

We have constructed the queuing system, whose parameters are s=4, a=1[m], b-=0.5 [m], k=3 [m], λ=188/600 [persons/sec], and μ=1/8 [persons/sec]. Figure 5 (a) is the snap shot of the experiment. Participants of the experiments enter the system and line up in the queue when the staff says to do so. They proceed to the windows and receive service. After that they wait at the starting position until the staff let him/her enter the system again. We put 188 people in 600 [sec] in one experiment. Note that the arrival was random while the service was deterministic for simplicity. According to the Pollaczek-Khintchine formula [7], W_q becomes small when the service is deterministic. Since this effect acts on the all kinds of the queuing systems in the same way, the results are not critically influenced by deterministic service. Therefore, we can examine the result of the theoretical analysis and simulations by these experiments. Figure 6 (b) shows the result of the experiments. We see that W_q in Parallel is smaller than that in D-Fork. This result verifies our theoretical analysis and simulation by using the walking-distance introduced queuing theory. The reversal of W_q between Parallel and D-Fork is observed experimentally. We also find that W_q becomes dramatically small in D-Fork-Wait. This new result indicates that the method “Keep one person waiting” is an effective way to shorten the waiting time empirically.

(a) (b)

Fig. 6. (a) Schematic view of the experiment. (b) The experimental mean traveling time for the normal start and the moving start.

Queuing System in Amusement Parks

We study queuing systems observed in amusement parks, bus stops, and concert halls in this section. In such queuing systems, plural pedestrians begin to move when a bus arrives or a gate of the hall opens.

Two starting methods, which are the *normal start* and the *moving start*, are compared. In the normal start case, people start to move when their former pedestrian moves, therefore, the last pedestrian has to wait some time from the movement of the first pedestrian. In the moving start case, people keep walking in the queue until the gate opens. The last pedestrian can move some distance before the information of the first pedestrian's movement is transmitted. Thus, the total travel time is expected to become smaller when we adapt the moving start.

We have performed the experiment of the normal start and the moving start as in Fig. 6 (a). First, twenty pedestrians are stand in the circuit. The density in the circuit is approximately 1 [person/m]. In the normal start case, the first pedestrian starts to move and go out the circuit when our staff claps his hands. Then the next pedestrian follows the first one and so on. Finally, the last pedestrian gets out of the circuit. In the moving start case, all pedestrians go around the circuit once and then start to go out. We measure the total travelling time, which is a time between the first pedestrian's departure and the last pedestrian's departure from the circuit. The results are described in Fig. 6 (b). We see that the total traveling time in the moving start case is smaller than that of the normal start case since pedestrians need not to wait their former pedestrian's start in the moving start case. The normal start in the various density conditions are studied with cellular automata simulation in Ref. [9].

Conclusion

We have studied the methods for improving the efficiency of the two kinds of queuing systems. In a queuing system with plural service windows, the mean waiting time decreases when we form the suitable system according to the degree of congestion and keep one person waiting at the service windows. In a queuing system at an amusement park, people can get on a roller coaster quickly if they keep walking when they are waiting in the queue.

Acknowledgments We thank Dai Nippon Printing Co. and Kozo Keikaku Engineering Inc. in Japan for the assistance of the experiments, which are described in Sec. 2 and Sec. 3, respectively. This work is supported by JSPS and JST.

References

1. Helbing, D.: Traffic and related self-driven many-particle systems. Rev. Mod. Phys. **73**, 1067–1141 (2001)
2. Nishinari, K., Kirchner, A., Namazi, A., Schadschneider, A.: Extended floor field CA model for evacuation dynamics. IEICE Trans. Inf. Syst. **E87-D**, 726-732 (2004)
3. Seyfried, A., Steffen, B., Klingsch, W., Boltes M.: The fundamental diagram of pedestrian movement revisited. J. Stat. Mech. P10002 (2005)
4. Yanagisawa, D., Kimura, A., Tomoeda, A., Nishi, R., Suma, Y., Ohtsuka, K., Nishinari, K.: Introduction of frictional and turning function for pedestrian outflow with an obstacle. Phys. Rev. E. **80**, 036110 (2009)
5. Kretz, T., Grunebohm, A., Schreckenberg., M.: Experimental study of pedestrian flow through a bottleneck. J. Stat. Mech**.** P10014 (2006)
6. Seyfried. A., Rupprecht, T., Passon, O., Steffen, B., Klingsch, W., Boltes., M.: New insights into pedestrian flow through bottlenecks. Transportation Science. **43,** 395–406 (2009)
7. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.: Queuing Networks and Markov Chains. A Wiley-Interscience Publication, U.S.A. (1998)
8. Blythe, R.A., Evans, M.R.: Nonequilibrium steady states of matrix product form: A solver's guide. J. Phys. A: Math. Gen. **40,** R333 (2007)
9. Tomoeda, A., Yanagisawa, D., Nishinari, K.: Sonic Speed on Pedestrian Dynamics: Relation between Sonic Speed and Density (in this proceedings)