ON VARIANTS OF STATE-DEPENDENT QUEUEING SYSTEMS FOR EGRESS MODELLING

Ruzelan Khalid (1), Gerrit K. Janssens (2)*

(1) Institute of Strategic Industrial Decision Modelling (ISIDM), School of Quantitative Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah; (2) Research Group Logistics, Faculty of Business Economics, Hasselt University, B-3590 Diepenbeek
[1] Malaysia, [2] Belgium

* Corresponding Author, e-mail: ruzelan@uum.edu.my

Abstract: An M/G/c/c state dependent network is a quantitative model for replicating and analyzing the behavior of occupants travelling through a network. The model, however, assumes that its inter-arrival times are independent and exponentially distributed, which inadequately reflects the dynamic and high-stakes nature in emergency situations. This paper simulates the M/G/c/c model for emergency evacuations using Erlang-k distributions—representing the real-world arrival patterns of evacuees in a more controlled and less random arrivals—and correlated inter-arrival times—common during evacuations as occupants moving in clusters due to panic or structural flow patterns. To achieve this, an M/G/c/c simulation model was first developed using Arena simulation software. The model was used to analyze the impact of various arrival rates on system performance metrics. Consequently, the performance metrics were compared with those obtained by replacing exponential inter-arrival times with Erlang-k distributions and independent arrivals with correlated cluster arrivals. The results show that Erlang-k distributions lead to better performance and smoother flow since the arrival is more controlled, while correlated arrivals increase congestion.

Key words: state dependent queueing systems, building egress, discrete event simulation.

1. INTRODUCTION AND BACKGROUND

These days, buildings are higher and more complex than before. At the same time, there is an exponential increase in the number of people affected by disasters. These disasters can be either natural, such as hurricanes, floods and earthquakes, or they can have an unnatural cause, such as terrorist attacks, or chemical releases of toxic and harmful substances. The disasters call for emergency evacuations. Often, due to human nature, the initial reaction is to start panicking, which can quickly escalate and lead to chaotic situations. Research in this field aims to formulate strategies in order to minimize losses from the disasters. Evacuation procedures try to minimize chaos and panic. The construction of evacuation procedures is the process of getting all evacuees outside the building in an orderly and organised way, as fast as possible.

When a disaster happens, the people should be alarmed and guided. Alarming systems can be divided into three categories: automatic detection systems, alarm alerting systems, and emergency lighting [1]. An automatic detection system is based on sensors which can detect heat, smoke or radiation. Alerting alarm systems include sounders, call points with a bell, and an internal communication system such as telephone or intercom.

There are several important dynamic aspects in an evacuation. At different moments in time, a building is busier or less busy. The topology of the building could also play a role during an emergency escape. Large rooms with few furniture, wide corridors and multiple large exit doors can facilitate a quicker evacuation. Another influencing factor is the type of occupants of the building. The evacuation of elderly people is different from the evacuation of children. The type of hazard also affects the course of the evacuation. In case of a fire, the smoke can lead to problems with inhalation and reduced visibility of the path.

All of these disasters have a big impact, not only in terms of casualties or even deaths, but also in terms of property damage, economic loss, and environmental loss. Efficient evacuation methods can help saving lives. In case of a real emergency evacuation, being prepared will reduce stress and hence save time. Saving time in these situations is key, as it will also save lives. Europe has made big improvements in fire safety over the years. Fire fatalities have decreased by 65% over the last 30 years, due to an improved approach to building.

The occupant evacuation time can be described as the time lapse from the beginning of the disaster until the occupant is safely evacuated [2]. It consists of three stages, namely the detection, pre-evacuation and movement stage. The detection time is the time between the start of the disaster and the moment it is noticed by the occupant. The pre-evacuation time covers the initial reaction of occupants, from the moment they notice the disaster until they start the actual evacuation process. The third stage is the movement time.

The influencing factors that determine the duration of each phase can be divided into three main categories: human characteristics and behaviour, building properties and disaster characteristics. An Italian research study investigated the effect of instructions on the evacuation process [3]. It was found that participants who received specific instructions during an evacuation experiment were able to reach the safe zone quicker than those participants that did not receive instructions. Information Technology can help a lot for safety and security. New technologies such as Internet of Things (IoT), Big Data (BD) and others can create new positive possibilities and challengeable deficits of safety [4].

Evacuation routes are part of this evacuation policy, and thus efficient routes also reduce property loss. The quicker occupants of a building are safely evacuated, the quicker the fire services can focus on controlling the disaster and limiting the building damage.

Queueing models are very useful to capture the specific dynamics of evacuees. A doorway, staircase or room is considered a server, in front of which evacuees' queue up according to the dynamics of a queueing model. The service time is related to the walking speed of the evacuees, as well as the time it takes to cross the doorway, staircase or room. Lino, et al. [5] uses two different queueing models to capture the emergency evacuation process. One queueing model accounts for rooms, corridors and stairways, and the capacity of the servers is related to the area of the considered spaces. A second queueing model accounts for the bottlenecks in the building, i.e. doors, exits, entrances and gateways, and in this case the capacity of the servers is related to the width of the bottleneck. These queueing models can then be used to estimate congestion and overall evacuation time.

This research focuses on state-dependent queueing models, specifically variants of the *M/G/c/c* models. These models are studied and compared. However, they typically assume independent, exponentially distributed inter-arrival times, which may not accurately capture the dynamic and critical nature of emergency situations. To address this, we simulate the *M/G/c/c* model for emergency evacuations using alternative arrival processes. Two types of arrival processes are considered. The first is the Erlang-k distribution, which better represents the real-world arrival patterns of evacuees with more controlled and less random arrivals. The second involves correlated inter-arrival times, common during evacuations when occupants move in clusters due to panic or structural flow patterns. The impact of these arrival processes on performance measures is then analysed.

This paper is organized as follows. Section 2 elaborates on the related work on state-dependent queueing systems. Section 3 proposes the experimental design and explains the discrete event simulation (DES) model to run the experiments. Section 4 shows the results of the experiments and discusses them. In the last section, conclusions are formulated.

2. STATE-DEPENDENT QUEUEING SYSTEMS: RELATED WORK

A queueing system is called state-dependent if either the arrival rate or the service rate depends on a certain state. The most obvious system with a state-dependent arrival rate is the single server queueing system with finite capacity, the *M/G/*1 queue with only K waiting places. The system is state dependent as the arrival rate $\lambda_j = \lambda, j < K$ and $\lambda_j = 0, j \geq K$, where new customers arrive at a rate λ_j when j customers are in the system. Another example refers to the single repairman problem. A closed queueing system with K machines is served by a repairman. Machines operate between breakdowns during an exponential time with mean $1/_{\lambda}$. When a machine breaks down, it queues for repair. The state of the system is defined as the number of machines j not working. The arrival rate to the repair facility is $\lambda_j = (K - j)\lambda, 0 \leq j \leq K$ [. Another interesting example is the switched Poisson process. The arrival rate switches alternatively between λ_1 and λ_2, governed by some random mechanism. The model is interesting since the arrival process covers both renewal and non-renewal processes with coefficients of variation larger than one [6].

When studying occupant movements, such as during the evacuation of a building, the service rate—representing the speed of movement—is state-dependent. The speed at which individuals can move and exit the building varies depending on the system's current conditions, such as crowd density, available exit routes, and the movement urgency. In the *M/G/c/c* state-dependent model, this relationship is modeled by treating the service rate as a dynamic function of the number of occupants present and their interactions.

As crowd density increases, the service rate decreases due to congestion, bottlenecks at exits, or the natural slowing down of movement within the crowd. Conversely, when fewer occupants are present, the service rate increases, facilitating quicker evacuation. Thus, the *M/G/c/c* model provides a powerful model for analyzing how different states of the system impact evacuation efficiency and can further be used to predict outcomes under various scenarios, helping to optimize evacuation strategies.

3. THE *M/G/c/c* ANALYTICAL MODEL

The effect of the density of occupants to the current walking speed was formularized by Yuhaski and Smith [7]. They presented an exponential model of walking speed in a confined space, such as a corridor:

$$V_n = A\exp\left[-\left(\frac{n-1}{\beta}\right)^{\gamma}\right] \tag{1}$$

where $\gamma = \dfrac{\ln\left[\dfrac{\ln\left(V_a/V_1\right)}{\ln\left(V_b/V_1\right)}\right]}{\ln\left(\dfrac{a-1}{b-1}\right)}$, and $\beta = \dfrac{a-1}{\left[\ln\left(\dfrac{V_1}{V_a}\right)\right]^{1/\gamma}} = \dfrac{b-1}{\left[\ln\left(\dfrac{V_1}{V_b}\right)\right]^{1/\gamma}}$

γ, β = Shape and scale parameters for the exponential model,
V_n = Average walking speed for n occupants in a confined space,
V_a = Average walking speed when crowd density is 2 peds/m^2 = 0.64 m/s,
V_b = Average walking speed when crowd density is 4 peds/m^2 = 0.25 m/s,
V_1 = Average walking speed for a single occupant = 1.5 m/s,
n = Number of occupants in a space,
$a = 2 \times l \times w$,
$b = 4 \times l \times w$,
l = space length in meters, and
w = space width in meters.

Based on the model, Yuhaski and Smith [7] developed the limiting probabilities for the number of occupants:

$$P_n = \frac{\left[\lambda E(S)\right]^n}{n!\,f(n)f(n-1)\ldots f(2)f(1)}P_0 \quad n = 1, 2, 3, \ldots, c \tag{2}$$

where $P_0^{-1} = 1 + \sum_{n=1}^{c}\left[\dfrac{\left[\lambda E(S)\right]^i}{i!\,f(i)f(i-1)\ldots f(2)f(1)}\right]$.

λ is the arrival rate to a space, $E(S)$ is the expected service time of a single occupant in the space, i.e., $E(S) = l/1.5$, P_n is the probability of having n occupants in the space, P_0 is the probability of having no occupant in the space, and $f(n)$ is the service rate, given by $f(n) = \frac{V_n}{V_1}$. c refers to the capacity of the space. The probability of such blocking (P_{balk}) is equal to P_n where n equals to c. Cheah and Smith [8] showed that *M/G/c/c* networks are equal to *M/M/c/c* networks. As a result, various performance measures of the space can then be computed as:

$$\theta = \lambda(1 - P_{balk}), \quad E(N) = \sum_{n=1}^{C} nP_n \quad \text{and} \quad E(T) = \frac{E(N)}{\theta}. \tag{3}$$

θ is the throughput of the space (in occupants per second), $E(N)$ is the expected number of occupants in the space, and $E(T)$ is the expected service time in seconds.

4. SIMULATING AN *M/G/c/c* ANALYTICAL MODEL

Simulating the *M/G/c/c* analytical model using programming languages, or simulation software, such as Arena [9], involves modelling a model that dynamically captures the nature of state-dependent queuing systems. The process starts by defining the system's states, with occupants arriving according to a Poisson process and their service rates depending on the current occupant number in the system. The simulation model involves initializing the system, scheduling its events (arrivals and departures), and updating its states accordingly. The model must dynamically adjust the service rate as occupants enter or exit, capturing the crowd density effect on system performance. Key metrics, such as throughput, blocking probability, expected number of occupants, and expected service time, are then collected and analyzed to assess the model's behavior under various conditions.

The simulation model should sufficiently represent complex state-dependent behaviour and non-linear interactions, allowing flexible scenario testing and performance measurement, which is not possible with analytical models. Scenario testing with simulations, such as adjusting arrival rates and using other speed-density models [10], provides insights into how the system behaves under different conditions and shows how key parameter changes impact performance, aiding in risk assessment and strategy planning.

Several *M/G/c/c* models have been discussed in the literature [11-14], along with computer programs that implement traditional analytical models using a Poisson process [15,16]. We used and modified the simulation model developed by Khalid, et al. [12]. To develop the *M/G/c/c* simulation model in Arena, key parameters such as arrival rate, service times, and the number of service channels, were defined. Various modules were also used. A *Create* module generates occupants based on a Poisson process to reflect the stochastic nature of inter-arrival times. The system's capacity, representing *c* servers, was modelled using *Seize*, *Delay*, and *Release* modules, which simulate service allocation and adjust the service rate as occupancy levels change.

The core of simulation involves dynamically updating the system's state as events such as arrivals or departures occur. Each event alters the service rates, reflecting the impact of varying crowd densities on system performance. An *Assign* module was used to track performance metrics. *Decide* and *Signal* modules were integrated to manage conditional flows and ensure accurate event synchronization and timely state updates.

After setting up the model, it was run multiple times to generate various outputs. Additionally, Arena's *Process Analyzer* was used to evaluate key performance metrics. This tool offers insights into how the system behaves under different conditions, such as varying arrival rates or service channels. The outputs can then be analysed to improve overall system performance. Details and graphical representation of the model can be found in Khalid, et al. [12].

5. SIMULATION EXPERIMENTS WITH EXTENDED VERSION OF THE MODEL

5.1. Base *M/G/c/c* Simulation Model

To demonstrate the impact of changing correlated inter-arrival times and the Erlang-*k* distribution on the performance metrics, the base *M/G/c/c* model was first simulated and evaluated. The simulation results were conducted over 20,000 seconds and 30 replications for a confined space of 8m x 2.5m. The results (including confidence intervals for each performance measure) are presented in Table 1.

Table 1. Performance measures versus arrival rates

λ	θ	$P(c)$	$E(N)$	$E(T)$
1.000	1.000 [0.998, 1.003]	0.000 [0.000, 0.000]	6.021 [6.004, 6.037]	6.019 [6.017, 6.021]
2.000	1.998 [1.995, 2.000]	0.000 [0.000, 0.000]	14.461 [14.427, 14.494]	7.239 [7.232, 7.246]
3.000	1.967 [1.956, 1.977]	0.343 [0.339, 0.347]	97.444 [96.745, 98.144]	49.572 [48.957, 50.187]
4.000	1.931 [1.931, 1.931]	0.516 [0.516, 0.517]	99.759 [99.751, 99.768]	51.660 [51.649, 51.671]

5.2. Changing the Exponential to Erlang-*k* Interarrival Times

The Erlang-*k* distribution simplifies to the exponential distribution when the shape parameter k=1. As k increases, the Erlang-*k* distribution approaches to a normal distribution, with reduced variability and more predictable arrival times. This characteristic makes the Erlang-*k* distribution suitable for modelling *M/G/c/c* networks in emergency evacuations, as it more accurately reflects the real-world arrival patterns of evacuees in controlled and less random situations compared to the exponential distribution.

By reducing variability and providing more controlled modelling, the Erlang-*k* distribution improves the predictability of system performance metrics, such as congestion and expected service time. This predictability is crucial for planning evacuation strategies and routes. Using the Erlang-*k* distribution in *M/G/c/c* networks enhances the safety and efficiency of emergency evacuations.

The Erlang-*k* distribution is also effective for modelling multi-phase processes such as evacuations. While the *M/G/c/c* model typically assumes independent arrivals without accounting for phases, the Erlang-*k* distribution can represent multiple phases. In an evacuation, for instance, occupants may pass through various phases, including evacuating offices, moving to corridors and stairwells, and exiting the building. The Erlang-*k* distribution models these phases as a separate exponential distribution, offering a more realistic depiction of the total evacuation time and capturing real-world complexities.

Smaller *k* values (2 to 5) are suitable for systems with moderate variability, balancing structure and randomness. Moderate *k* values (6 to 10) provide greater predictability and reduced variability, concentrating the distribution around the mean. Larger *k* values (over 10) offer very low variability and high predictability.

To assess the impact of changing the shape parameter *k* on performance metrics, four *k* values were tested. For each *k*, four different arrival rates were evaluated. Figure 1 shows the plots of throughput and expected service time versus arrival rates.

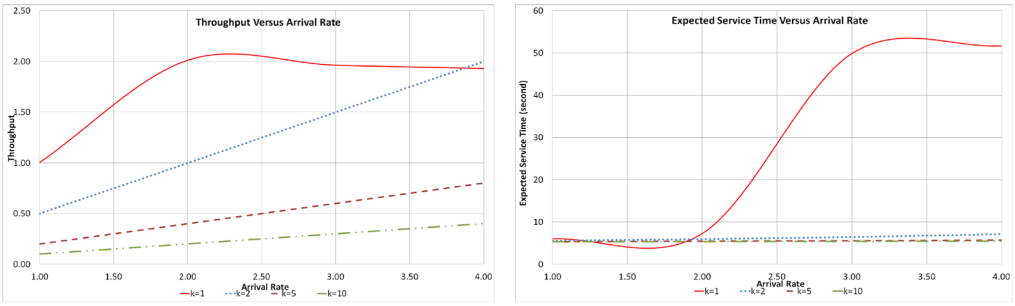

Figure 1. Throughput and expected service time (in seconds) versus arrival rates

Table 2. The impact of arrival rates on performance for a set of parameter k values

λ	*k=1*				*k=2*			
	θ	*P(c)*	*E(N)*	*E(T)*	*θ*	*P(c)*	*E(N)*	*E(T)*
1	1.004	0.000	6.044	6.022	0.500	0.000	2.786	5.576
2	2.012	0.000	14.614	7.263	0.998	0.000	5.934	5.943
3	1.963	0.348	97.754	49.822	1.499	0.000	9.635	6.428
4	1.931	0.519	99.765	51.670	1.999	0.000	14.191	7.098
	k=5				*k=10*			
1	0.200	0.000	1.074	5.374	0.100	0.000	0.533	5.334
2	0.400	0.000	2.192	5.480	0.200	0.000	1.072	5.363
3	0.600	0.000	3.362	5.605	0.300	0.000	1.625	5.414
4	0.800	0.000	4.598	5.745	0.400	0.000	2.188	5.469

The detailed simulation results are presented in Table 2. As observed, increasing the *k* value, while maintaining the same arrival rate, reduces variability in the arrival process and increase predictability. This leads to improved system performance.

To further explore how varying *k* affects system performance, *k* values are examined specifically for arrival rates of 2 and 4 occupants/sec. Figure 2 illustrate the throughput and expected service time (in seconds) for different *k* values at these arrival rates. Detailed performance metrics are provided in Table 3. Changing exponential inter-arrival times to the Erlang-*k* distribution enhances performance metrics by lowering the blocking probability and expected service time.

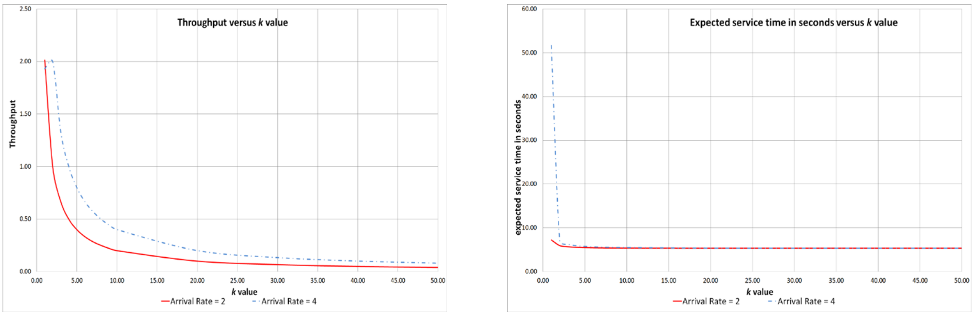

Figure 2. Throughput and expected service time (in second) versus k values

Table 3. The impact of the shape parameter k on performance for a set of arrival rates

K	$\lambda = 2$				$\lambda = 4$			
	θ	*P(c)*	*E(N)*	*E(T)*	θ	*P(c)*	*E(N)*	*E(T)*
1	2.012	0.000	14.614	7.263	1.931	0.519	99.765	51.67
2	0.998	0.000	5.934	5.943	1.999	0.000	14.191	7.098
3	0.667	0.000	3.780	5.667	1.332	0.000	8.286	6.220
4	0.499	0.000	2.770	5.546	0.999	0.000	5.898	5.906
5	0.400	0.000	2.192	5.480	0.800	0.000	4.598	5.745
6	0.333	0.000	1.811	5.439	0.666	0.000	3.762	5.645
7	0.285	0.000	1.545	5.411	0.572	0.000	3.192	5.579
8	0.250	0.000	1.349	5.391	0.500	0.000	2.767	5.532
9	0.222	0.000	1.194	5.375	0.444	0.000	2.440	5.496
10	0.200	0.000	1.072	5.363	0.400	0.000	2.188	5.469
20	0.100	0.000	0.533	5.333	0.200	0.000	1.070	5.357
30	0.067	0.000	0.356	5.333	0.133	0.000	0.711	5.334
40	0.050	0.000	0.267	5.333	0.100	0.000	0.534	5.333
50	0.040	0.000	0.214	5.333	0.080	0.000	0.427	5.333

In a real-life situation, this reflects the benefits of having a well-coordinated and planned approach to managing the arrival of people during emergency evacuations. For example, if evacuees arrive at evacuation points in a controlled and predictable manner, such as through staggered arrivals, emergency services can operate more efficiently. This results in fewer bottlenecks, smoother movement of people, and less congestion, making the evacuation process safer and more effective.

5.3. Correlated Inter-Arrival Times

Correlated inter-arrival times occur when the time between consecutive arrivals is influenced by previous arrivals, rather than being independent. In an emergency evacuation, this might happen as people exit in quick succession due to panic or crowd flow, creating clusters of arrivals. Modeling these correlated arrivals can improve the accuracy of evacuation simulations, leading to better predictions of bottlenecks, more precise evacuation time estimates, and more effective planning for safety and efficiency.

The *M/G/c/c* model uses a Poisson process, assuming that arrivals are independent and memoryless, which may not reflect how people behave and move in emergencies. In such situations, people often arrive due to panic, urgency, and crowd dynamics, leading to highly correlated or time-dependent arrivals. These arrival rates are typically non-stationary, increasing as the situation escalates. As time passes, more people rush to exits, increasing the arrival rate. This increasing intensity cannot be captured by a basic Poisson process. To address this, a non-homogeneous Poisson process with a time-dependent rate for arrivals should be used, where the rate of arrivals increases over time to reflect the growing urgency and crowd density to exit the area.

To model a time-dependent Poisson process in Arena, we used a combination of the *Create* module, *Variables* spreadsheet, and *Expressions* spreadsheet. First, we determined the function that describes how the arrival rate increases over time. For this, we considered an arrival rate λ(t) that increases linearly over time:

$$\lambda(t) = \lambda_0 + kt \tag{4}$$

where $\lambda(t)$ is the arrival rate at time *t*, λ_0 is the initial arrival rate, and k is the rate of increase per time unit.

We set $\lambda_0 = 0.5$ (0.5 occupant per second) and vary *k* values from 0 to 0.05. This expression can be included in an *Expression* spreadsheet in Arena as *ArrivalRate = Lambda0 + k * TNOW* where *TNOW* refers to the current simulation time in Arena. The *ArrivalRate* variable was then used in a *Create* module to set the *Time Between Arrivals* to *EXPO(1/ArrivalRate)*. As the simulation runs, *TNOW* increases, leading to an increase in *ArrivalRate*. Two cases are considered: one with no batch arrivals, where occupants arrive individually, and another with batch arrivals, where occupants arrive in clusters up to three. The impact of correlated arrivals on performance measures under different scenarios is shown in Table 4.

As the rate of arrival per time unit, *k*, increases, the blocking probability, the expected number of occupants, and the expected time spent in the system consistently rise for both cases: individual arrivals and batch arrivals.

Table 4. The impact of correlated arrivals on performance measures

k	No batch arrival				Batch arrival			
	θ	*P(c)*	*E(N)*	*E(T)*	θ	*P(c)*	*E(N)*	*E(T)*
0.000	0.100	0.000	0.561	5.633	0.199	0.000	1.236	6.219
0.010	0.385	0.905	18.551	48.149	0.387	0.907	18.828	48.671
0.020	0.385	0.952	19.143	49.728	0.386	0.952	19.287	49.935
0.030	0.385	0.968	19.352	50.307	0.385	0.976	19.535	50.742
0.040	0.385	0.976	19.490	50.590	0.385	0.984	19.634	50.991
0.050	0.384	0.981	19.527	50.797	0.385	0.988	19.689	51.118

In the *no batch* arrival case, the blocking probability starts at 0.100 when *k*=0, indicating low congestion when arrivals are evenly spaced. As *k* increases to 0.05, the blocking probability gradually increases to 0.384, reflecting a higher chance of congestion as arrivals become more clustered over time. Similarly, the expected number of occupants and expected time spent show a significant rise, indicating that more occupants are in the system for longer periods as *k* increases.

In the *batch* arrival scenario, the results are slightly higher across all metrics. For example, when *k*=0, the blocking probability is 0.199 compared to 0.100 in the no batch case, and the expected time spent is also longer. This suggests that batch arrivals, even when limited to clusters of up to three, significantly impact system performance, worsening congestion and delays compared to individual arrivals.

Overall, the data show that increasing the arrival rate and allowing batch arrivals worsen performance measures such as higher blocking probability and longer expected service time, as shown in Figure 3. Batch arrivals further increase the impact of correlated arrivals as congestion grows. These findings highlight the need to manage arrival patterns to reduce congestion and improve system performance.

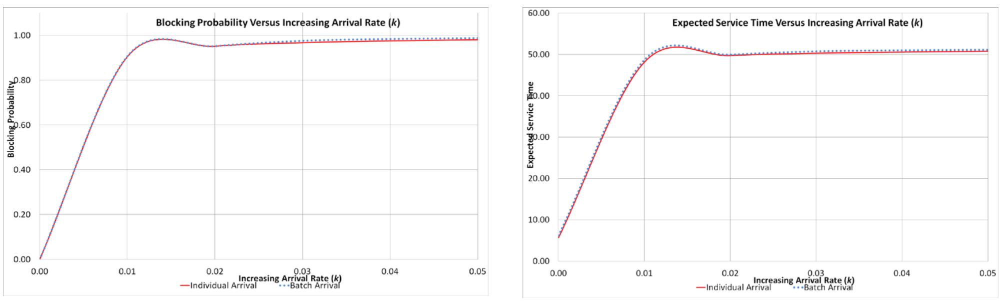

Figure 3. Blocking probability and expected service time versus increased arrival rate

6. CONCLUSION

This study demonstrates that incorporating Erlang-*k* distributions and correlated inter-arrival times into *M/G/c/c* models improve evacuation simulations. Erlang-*k* distributions reflect the more structured and less random nature of real-world evacuee arrival patterns, compared to the independent and memoryless exponential inter-arrival times. This controlled and predictable arrival process leads to better system performance, smoother flow, and reduced congestion during evacuations. This highlights the importance of planning and coordination for safer and more effective emergency evacuations.

Considering correlated arrivals is also important since it mirrors real-world behaviors during emergencies, where people often move in clusters due to panic or structural flow patterns. Correlated arrivals increase congestion, highlighting the challenges of managing evacuations and the need for models that can capture such dynamics. By integrating both correlated and cluster arrivals, the model more accurately represents real-world evacuation scenarios, offering critical insights for optimizing evacuation strategies, predicting performance, and effectively utilizing available space during emergencies.

REFERENCES

[1] Galea, E., M. Patel. *Principles and Practice of Evacuation Modelling: A Collection of Lecture Notes for a Short Course*, CMS Press, London, 2016.

[2] Liu, Y., Z. Zhang, Z. Mao. Analysis of influencing factors in pre-evacuation time using Interpretive Structural Modeling. *Safety Science*, vol 128, no 104785, 2020. https://doi.org/10.1016/j.ssci.2020.104785

[3] Carattin, E., C. Meneghetti, V. Tatano, F. Pazzaglia. Human navigation inside complex buildings: using instructions and maps to reach an area of refuge. *International Journal of Design Creativity and Innovation*, vol 4, no 2, 2016, pp.105–118. https://doi.org/10.1080/21650349.2015.1135760

[4] Kharchenko, V. Big data and Internet of Things for safety critical applications: chanllenges, methodologies and industry cases. *International Journal of Information Technologies & Security,* vol. 10 no. 4, 2018, pp.3-17.

[5 Lino, P., G. Maione, B. Maione, Modeling and simulation of crowd egress dynamics in a discrete event environment, *2009 IEEE Control Applications, (CCA) & Intelligent Control, (ISIC)*, 2009, pp. 843–848.

[6] van Hoorn, M.H., L.P. Seelen. The SPP/G/1 queue: A single server queue with a switched Poisson process as input process. *Operations-Research-Spektrum*, vol 5, no 4, 1983, pp.207–218. https://doi.org/10.1007/BF01719844

[7] Yuhaski Jr, S.J., J.M. Smith. Modeling circulation systems in buildings using state dependent queueing models. *Queueing Systems*, vol 4, no 4, 1989, pp.319–338. https://doi.org/10.1007/BF01159471

[8] Cheah, J., J.M. Smith. Generalized M/G/C/C state dependent queueing models and pedestrian traffic flows. *Queueing Systems*, vol 15, no 1, 1994, pp.365–386. https://doi.org/10.1007/BF01189246

[9] Kelton, W.D., R. Sadowski, N. Zupick. *Simulation with Arena*, McGraw-Hill Education, New York, 2024.

[10] Khalid, R., M.K.M. Nawawi, N. Ishak, M.A. Baten. Optimising pedestrian flow in a topological network using various pairwise speed-density models. *Operations Research and Decisions*, vol 33, no 4, 2023, pp. 53–69. https://doi.org/10.37190/ord230404

[11] Cruz, F.R.B., J. MacGregor Smith, R.O. Medeiros. An M/G/C/C state-dependent network simulation model. *Computers & Operations Research*, vol 32, no 4, 2005, pp.919–941. https://doi.org/10.1016/j.cor.2003.09.006

[12] Khalid, R., M.K.M. Nawawi, L.A. Kawsar, N.A. Ghani, A.A. Kamil, A. Mustafa. A discrete event simulation model for evaluating the performances of an M/G/C/C state dependent queuing system. *PLoS ONE*, vol 8, no 4, 2013, pp.1–9. https://doi.org/10.1371/journal.pone.0058402

[13] Cruz, F.R.B., J. MacGregor Smith, D.C. Queiroz. Service and capacity allocation in M/G/c/c state-dependent queueing networks. *Computers & Operations Research*, vol 32, no 6, 2005, pp.1545–1563. https://doi.org/10.1016/j.cor.2003.11.019

[14] Khalid, R., M.K.M. Nawawi, L.A. Kawsar, N.A. Ghani, A.A. Kamil, A. Mustafa. The evaluation of pedestrians' behavior using M/G/C/C analytical, weighted distance and real

distance simulation models. *Discrete Event Dynamic Systems*, vol 26, no 3, 2016, pp.439–476. https://doi.org/10.1007/s10626-015-0215-0

[15] Khalid, R., M.A. Baten, M.K.M. Nawawi, N. Ishak. Analyzing and optimizing pedestrian flow through a topological network based on M/G/C/C and network flow approaches. *Journal of Advanced Transportation*, vol 50, no 1, 2016, pp.96–119. https://doi.org/10.1002/atr.1330

[16] Khalid, R., M.K.M. Nawawi, L.A. Kawsar, N.A. Ghani, A.A. Kamil, A. Mustafa. Optimal routing of pedestrian flow in a complex topological network with multiple entrances and exits. *International Journal of Systems Science*, vol 51, no 8, 2020, pp.1325–1352, https://doi.org/10.1080/00207721.2020.1756524

Information about the authors:

Ruzelan Khalid is an Associate Professor of Decision Science at Universiti Utara Malaysia. He earned his Ph.D. in Computer Science from the University of Canterbury in New Zealand. His research focuses on Discrete Event Simulation and Data Envelopment Analysis.

Gerrit K. Janssens is an Emeritus Professor of Operations Management and Logistics from Hasselt University in Belgium. He holds a Ph.D. in Computer Science from the Free University of Brussels, Belgium. His research focuses on the application and development of operations research models in the fields of production, transportation and logistics.

Manuscript received on 28 September 2024