Transportation Research Part C 144 (2022) 103913

Contents lists available at ScienceDirect

Transportation Research Part C

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

System Identification for the design of behavioral controllers in crowd evacuations

Miguel A. Lopez-Carmona

Universidad de Alcala, Escuela Politecnica Superior, Departamento de Automatica, Campus Externo de la Universidad de Alcala, Alcala de Henares, Madrid, Spain

ARTICLE INFO

Keywords:
Crowd evacuation
Behavioral optimization
Distance-keeping
Behavioral controller
System identification
Model predictive control (MPC)

ABSTRACT

Behavioral modification using active instructions is a promising interventional method to optimize crowd evacuations. However, existing research efforts have been more focused on eliciting general principles of optimal behavior than providing explicit mechanisms to dynamically induce the desired behaviors, which could be claimed as a significant knowledge gap in crowd evacuation optimization. In particular, we propose using dynamic distance-keeping instructions to regulate pedestrian flows and improve safety and evacuation time. We investigate the viability of using Model Predictive Control (MPC) techniques to develop a behavioral controller that obtains the optimal distance-keeping instructions to modulate the pedestrian density at bottlenecks. System Identification is proposed as a general methodology to model crowd dynamics and build prediction models. Thus, for a testbed evacuation scenario and input–output data generated from designed microscopic simulations, we estimate a linear AutoRegressive eXogenous model (ARX), which is used as the prediction model in the MPC controller. A microscopic simulation framework is used to validate the proposal that embeds the designed MPC controller, tuned and refined in closed-loop using the ARX model as the Plant model. As a significant contribution, the proposed combination of MPC control and System Identification to model crowd dynamics appears ideally suited to develop realistic and practical control systems for controlling crowd motion. The flexibility of MPC control technology to impose constraints on control variables and include different disturbance models in the prediction model has confirmed its suitability in the design of behavioral controllers in crowd evacuations. We found that an adequate selection of output disturbance models in the predictor is critical in the type of responses given by the controller. Interestingly, it is expected that this proposal can be extended to different evacuation scenarios, control variables, control systems, and multiple-input multiple-output control structures.

1. Introduction

The management of congestion situations and operational efficiency in crowd evacuations and egress operations in facilities are relevant topics in transportation research (Bi and Gelenbe, 2019), which have been mainly focused on emergency scenarios (Helbing and Mukerji, 2012; Haghani and Sarvi, 2017a). However, how to improve comfort-related non-critical conditions, operational efficiency, or reducing contact between people are research topics gaining attention (Murakami et al., 2020). In any of these scenarios, crowd motion management systems aim will be to apply mechanisms intending to avoid dangerous situations and ensure reasonable egress times, although it is well known that these objectives may conflict in highly congested egress operations (Lopez-Carmona and Paricio Garcia, 2021).

E-mail address: miguelangel.lopez@uah.es.

https://doi.org/10.1016/j.trc.2022.103913
Received 31 May 2022; Received in revised form 13 September 2022; Accepted 28 September 2022

In the systematic review conducted by Haghani (2020b), three major methods are emphasized to optimize crowd evacuations using interventional approaches: architectural, mathematical, and behavioral, that seek evacuation efficiency rather than merely describe or predict evacuations. The study concludes that the reviewed literature is dominated by architectural solutions, which seek to facilitate pedestrian flows through strategical alterations to the physical space or the movement, while the evidence on their effectiveness is largely inconclusive and context-specific. Mathematical programming and optimization of path/departure-schedule planning solutions do not address microscopic aspects of individual behavior, raising the question of how could be implemented and enforced in the field. Finally, the behavioral modification (training and active instructions) method has been the least studied approach. However, it is gaining increasing attention as a promising method in terms of effectiveness and practicality, constituting a significant knowledge gap in evacuation dynamics literature. The "behavioral optimization" concept and its relevance in optimizing crowd evacuations were presented in Haghani (2020b), under the idea that adequate active instructions or training have the potential to make people be part of the evacuation solution effectively, considering that the performance of an evacuation process is the collective outcome of individual enacted strategies (Kinateder et al., 2018; Chen et al., 2018; Haghani and Sarvi, 2019d; Feliciani et al., 2020).

Implementing effective behavioral optimization methods using active instructions requires developing crowd motion control systems that monitor the environment and generate the appropriate instructions in real-time as a response to fulfill safety and egress time goals. Thus, an efficient coordination of evacuees (Bi and Gelenbe, 2019) using active instructions could be achieved by deploying guidance systems that dynamically inform to pedestrians about different decision-making elements such as exit choice (Lopez-Carmona and Paricio Garcia, 2021, 2022; Haghani and Sarvi, 2019d, 2017b), the path selection (Wang et al., 2015), or the evacuation start time (Abdelghany et al., 2014; Murakami et al., 2020). Regardless of whether the instructions are communicated by signaling systems directly or through leadership and guidance, in any of these scenarios, a real deployment will potentially have a clear connection with available technologies, such as positioning systems (Nguyen-Huu et al., 2017), modern crowd detection technologies (Jacques et al., 2010; Kaiser et al., 2018) and pedestrian tracking systems (Johansson et al., 2008; Brunetti et al., 2018). Our aim should be to take advantage of all the accumulated know-how about crowd dynamics (Helbing and Molnár, 1995; Zhang et al., 2012), leveraging existing technologies to develop crowd motion management systems on behavioral modification (Haghani, 2020a; Kok et al., 2016).

The existing efforts in behavioral optimization supported by crowd motion control systems that provide instructions to evacuees or leaders dynamically are significantly less than those devoted to studying optimal behaviors per se. There are several studies, mainly based on numerical simulations so far, on how the presence of leaders could potentially improve evacuation efficiency (Wang et al., 2015; Ma et al., 2016; Song et al., 2017; Yang et al., 2020; Ma et al., 2017). However, these papers do not address the practical application of mechanisms to inform leaders of their optimal behavior. An exception is in Lopez-Carmona and Paricio Garcia (2022), where we proposed using an adaptive cell-based crowd evacuation system that dynamically generates exit-choice recommendations to leaders, favoring a coordinated group dynamic that improves safety and evacuation time.

Behavioral conduct at the bottlenecks and its effect on flow efficiency and safety has also been the subject of research in many studies, discussing the two controversial conditions "faster is slower" and "faster is faster" (Haghani et al., 2019; Shi et al., 2019b; Sticco et al., 2017). Another body of studies has explored, exclusively via mathematical formulations and numerical simulation experiments, how modifying the level of selfish or cooperative behavior can influence evacuation (Zheng and Cheng, 2011; Haghani and Sarvi, 2019b; Song et al., 2016; Cheng and Zheng, 2018; Dossetti et al., 2017; Zou et al., 2020; Shi et al., 2019a). These studies suggest that intermediate cooperation results in the lowest evacuation times. Neither too many cooperators nor too many defectors are beneficial for the evacuation process. In this line, some studies have investigated whether the imitation tendency could positively affect evacuation performance. In Haghani and Sarvi (2019a) and Haghani and Sarvi (2019d), it is shown that peer imitation in exit choice is detrimental, whereas the crowd could benefit from intermediate degrees of peer imitation tendencies for exit choice changing.

How pedestrians' exit and route choice strategies can be improved to optimize evacuations has also been investigated in many works. In Zhou et al. (2019) five evacuation route choice strategies are compared using the social force model in large-scale public space. They observed that the strategy considering density and capacity factors had the best performance. Haghani and Sarvi (2019c) investigates the system-level optimality of the exit-choice behavior, and they observe that "the system could improve by modifying individuals' marginal valuations of exit attributes." We proposed in Lopez-Carmona and Paricio Garcia (2021) an adaptive guidance system that provides efficient exit-choice indications in a cell-based positioning infrastructure. The operational module that dynamically generates the optimal instructions is built upon a discrete-choice model used as a heuristic function trained through simulation–optimization.

Exciting studies in Yanagisawa et al. (2012) and Zeng et al. (2019) have explored the potential impact of the walking behavior in directional flows using rhythm. They observed that slow-rhythmic walking achieves a larger flow in the high-density regime. Finally, some studies have explored the behavioral modification methods concerning pre-evacuation delays, closely related to optimization of path/departure-schedule planning solutions. Haghani and Sarvi (2019c) investigated various degrees of waiting strategies and observed that the instant-response strategy creates denser and longer-lasting bottlenecks, but it does not translate to longer evacuation times. However, this could have significant implications for evacuation safety as studied in Murakami et al. (2020), where the optimization goal is to minimize evacuation time and density.

From the reviewed research, it can be observed that research efforts are more focused on eliciting general principles of optimal behavior than providing explicit mechanisms to dynamically induce the desired behaviors, which could be claimed as the most substantial knowledge gap that can be currently pointed out in the domain of crowd evacuation optimization. Thus, this paper's primary motivation is to contribute with a general proposal for behavioral optimization supported by a crowd motion control

system that provides instructions to evacuees dynamically. Our interest is in producing simple, readily-communicable yet practical instructions for people during evacuation. We adopt the general concepts described in Haghani and Sarvi (2019c), Murakami et al. (2020) and Fang et al. (2011) about the waiting strategies to mitigate congestion and focus our interventional goal on controlling the distance between pedestrians to mitigate crowding at bottlenecks (“distance-keeping” hereafter). Inspired by the buffer zone concept presented in Wang et al. (2019), which evaluates how waiting areas may play a significant positive role in congestion management at exits, we build our behavioral control system around a control area over which the active instructions are communicated.

Our central hypothesis is that by dynamically providing distance-keeping instructions to pedestrians in a predefined area upwards, we may regulate congestion to the desired values downwards to improve safety and evacuation time for directional pedestrian flows.

We are interested in evaluating the viability of using a closed-loop control system that provides the optimal control actions (active instructions in the form of distance-keeping values) dependent on the current evacuation state to fulfill desired objectives in terms of evacuation time and congestion at bottlenecks. Consequently, a central question of our study concerns which could be a valid control system technology? We are also interested in applying a consistent and general methodology to model crowd dynamics to facilitate the design of behavioral controllers.

With the purposes mentioned, a linear Model Predictive Control (MPC) (Camacho and Alba, 2013) design is proposed to generate optimal distance-keeping instructions in a control area dynamically. The input to the MPC controller is the pedestrian density measurement in a downwards sensing area, which is the variable to be regulated to control pedestrian flow. The MPC controller embeds a prediction model structured as a black-box linear AutoRegressive eXogenous model (ARX) (Ljung, 1999), which can reproduce the crowd dynamics in the evacuation scenario. The ARX model is estimated from input–output data obtained from simulation experiments launched in a microscopic pedestrian simulator, using the System Identification (Ljung et al., 2020) methodology. To validate our work, we conduct simulations and measure the evacuation efficiency in two dimensions, egress time and pedestrian density or crowd pressure.

Our proposal is expected to be valid in evacuation scenarios where the goal is to improve operational efficiency by trading-off evacuation time and safety (e.g., in music festivals where operational efficiency and safety are a must) and in emergency evacuation scenarios where the hazard does not disproportionately influence the evacuee’s urgency levels. In this work, by safety, we mainly refer to the intrinsic risk of the crowd due to excessive crowd pressure. However, the model and control mechanisms proposed are also valid for trading-off crowd pressure and egress time in scenarios where long evacuation times are detrimental, such as bomb threats. Overall, from this research, it can be reasonably deduced that the proposed methodology that combines system identification and behavioral control system design can be extended to a variety of evacuation scenarios, control system technologies, and control design variables.

This study contributes to the growing literature on interventional approaches that seek to improve evacuation performance rather than predict or describe evacuations. The paper makes several contributions to the literature on optimization approaches based on behavioral modification, which is gaining increasing attention as a promising method in both effectiveness and practicality (Haghani, 2020b). More specifically, the paper fills a gap in the existing literature about implementing control systems to elicit optimal active instructions to pedestrians during evacuations:

- It provides insights on the first use of MPC controllers to manage the evacuee’s motion adaptively.
- Our paper contributes with a comprehensive methodology based on System Identification to model crowd dynamics, ideally suited to designing behavioral control systems.
- Distinct from existing studies, this paper proposes the distance-keeping control variable to communicate simple and readily-communicable instructions to pedestrians during evacuation.
- It helps provide a better understanding of how different disturbance models within the prediction model in the MPC controller may contribute to rejecting output disturbances in the control system and balance evacuation time and congestion.
- This study provides reasonable evidence that the proposed mechanisms and methods can fit different evacuation scenarios, control design variables, and control system technologies.

The rest of the paper is organized as follows. Section 2 presents the evacuation scenario, including how microscopic crowd behavior has been modeled and the metrics used to measure evacuation performance. It also presents preliminary experiments that motivate using distance-keeping instructions to pedestrians as a mechanism to modulate crowd congestion and evacuation time. Section 3 outlines the critical points of system identification techniques and develops the data processing, linear model identification, and model validation tasks to identify a linear polynomial model representing the crowd dynamics of the evacuation scenario. The MPC controller design is outlined in Section 4. Section 5 presents the experimental evaluation, based on simulation analyses. The last two sections provide the discussion, concluding comments and possible research extensions.

2. Crowd dynamics, evacuation efficiency and distance-keeping

Most research on crowd behavior and evacuations have been based on microscopic simulations. In Bi and Gelenbe (2019), different simulation models are classified into cellular automata models (Pelechano and Malkawi, 2008; Feliciani and Nishinari, 2016), social force models (Helbing and Molnár, 1995), fluid-dynamics models (Henderson, 1971), lattice gas models (Takimoto and Nagatani, 2003), game-theoretic models (Hoogendoorn and Bovy, 2003) and computer agent-based models (Pan et al., 2007). In this paper, we adopted the Social Force Model (SFM) developed in Helbing et al. (2000) and Helbing et al. (2005), representing a widely accepted crowd behavior model that blends socio-psychological and physical forces. However, it does not preclude using

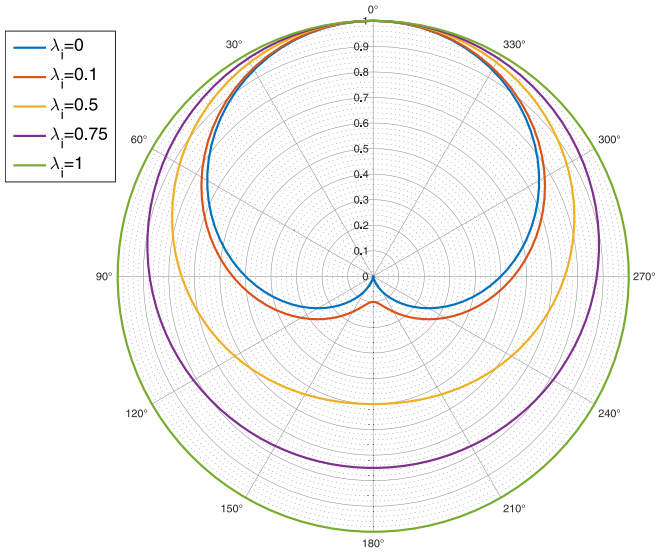

Fig. 1. Polar plot of the anisotropy function for different values of parameter λ_i. For lower values of λ_i the influence of pedestrians behind the motion direction is lower that for higher values. In the extreme, for $\lambda_i = 1$, pedestrians located at a given distance contribute with the same repulsion force, independently of their angle to the motion direction.

other models that include socio-psychological dimensions. On the simulation side, we opted for a multi-agent microscopic simulation framework built upon the software packages AnyLogic[1] and Matlab.[2] The framework core is AnyLogic, which implements the evacuation scenario and the social force model for simulating pedestrian motion. The pedestrians' behavior models and the system logic were implemented in Matlab. AnyLogic and Matlab interconnect in a master–slave configuration through external Java libraries provided by AnyLogic and the Matlab Java API engine. For the interested reader, in Lopez-Carmona and Paricio Garcia (2021), the details and motivations behind using this framework can be found. Also, in Lovreglio et al. (2020), an interesting online survey studies the main trends in using pedestrian evacuation simulators.

In the Helbing's SFM, each pedestrian i of mass m prefers to move at a speed $v_i^{preferred}$ into a direction $\mathbf{e}_i$, adapting the actual speed $\mathbf{v}_i$ with a certain characteristic time τ_i. Simultaneously, pedestrians try to keep a velocity-dependent distance to other pedestrians j and obstacles O. These effects are modeled by repulsive interaction forces $\mathbf{f}_{ij}$ and $\mathbf{f}_{iO}$, such that the change of velocity in time t is given by the acceleration equation

$$\mathbf{f}_i(t) = m_i \frac{d\mathbf{v}_i}{dt} = m_i \frac{v_i^{preferred}(t)\mathbf{e}_i^{preferred}(t) - \mathbf{v}_i(t)}{\tau_i} + \sum_{i \neq j} \mathbf{f}_{ij} + \sum_{O} \mathbf{f}_{iO} \,. \tag{1}$$

Pedestrian speed is derived from Eq. (1) at each time step Δt as follows:

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{\mathbf{f}_i(t)}{m_i} \Delta t \,.$$

The repulsive interaction force $\mathbf{f}_{ij}$ induced by the psychological and physical tendencies of pedestrians is modeled by

$$\mathbf{f}_{ij} = A_i exp\left[(r_{ij} - d_{ij})/b_i\right] \mathbf{n}_{ij} \cdot \Omega(\lambda_i, \varphi_{ij}) + kg(r_{ij} - d_{ij})\mathbf{n}_{ij} + \kappa g(r_{ij} - d_{ij}) \Delta v_{ji}^t \mathbf{t}_{ij} \,. \tag{2}$$

The first term in Eq. (2) describes the psychological tendency of two pedestrians i and j to stay away from each other, where $A_i[N]$ and $B_i[m]$ represent the strength and range of interaction, $d_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|$ denotes the distance between the pedestrians' centers of mass, $\mathbf{n}_{ij} = (n_{ij}^x, n_{ij}^y) = (\mathbf{r}_i - \mathbf{r}_j)/d_{ij}$ is the unit vector pointing from pedestrian j to i, and function $\Omega(\lambda_i, \varphi_{ij}) \Rightarrow [0, 1]$ takes into account the anisotropic character of pedestrian interactions (view area). Thus, the anisotropy function is used to weigh the psychological repulsive forces considering the position of other pedestrians or obstacles relative to the direction of motion. This function is defined as follows

$$\Omega(\lambda_i, \varphi_{ij}) = \lambda_i + (1 - \lambda_i)\frac{1 + cos(\varphi_{ij})}{2}$$

where φ_{ij} denotes the angle between the direction of motion $\mathbf{e}_i(t)$ and the direction of the pedestrian exerting the repulsion force $-\mathbf{n}_{ij}(t)$, and $0 \leq \lambda_i \leq 1$ shapes the influence area. Fig. 1 shows the shape of the anisotropy function for different values of λ_i.

[1] https://www.anylogic.com/ Accessed 3 June 2021.

[2] https://www.mathworks.com/ Accessed 3 June 2021.

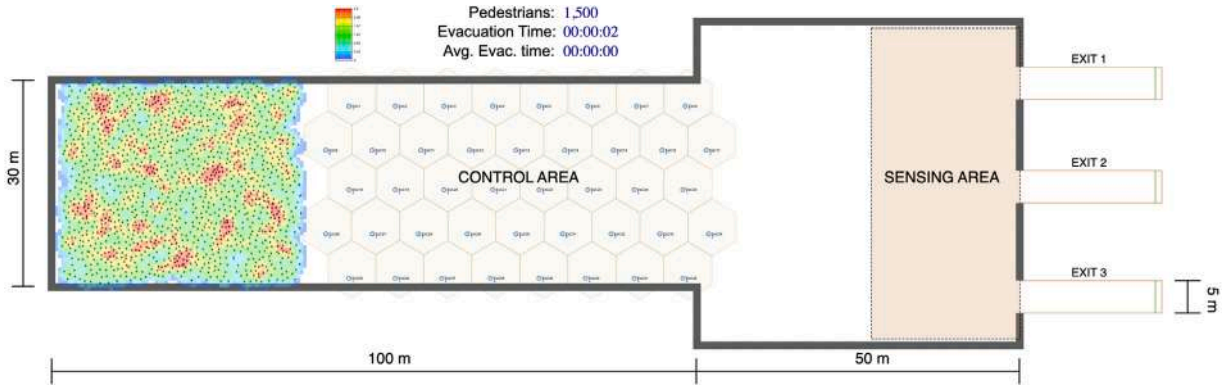

Fig. 2. Schematic for the evacuation scenario considered in this work. Pedestrians in the left area are evacuated through the three exit gates. The hexagonal cells structure represents the control area, while the sensing area is close to the exit gates providing density and crowd pressure measurements in real-time.

When distance d_{ij} is smaller than the sum of the pedestrians' radii $r_{ij} = (r_i + r_j)$, the pedestrians touch each other, and two additional physical forces are added (second and third terms in Eq. (2)):

- A body force $kg(r_{ij} - d_{ij})\mathbf{n}_{ij}$ counteracting body compression.
- A sliding friction force $\kappa g(r_{ij} - d_{ij}) \triangle v^t_{ji}\mathbf{t}_{ij}$ impeding relative tangential motion. The tangential direction is represented by the unit vector $\mathbf{t}_{ij} = (-n^y_{ij}, n^x_{ij})$, while $\triangle v^t_{ji} = (\mathbf{v}_j - \mathbf{v}_i) \cdot \mathbf{t}_{ij}$ denotes the tangential velocity difference.

The function $g(x)$ in the physical forces is zero if the pedestrians do not touch each other, otherwise $g(x) = x$.

The interaction with obstacles $\mathbf{f}_{iO}$ is treated analogously by substituting pedestrian j by the closest point to the obstacle, where $\mathbf{v}_O = 0$:

$$\mathbf{f}_{iO} = A_i exp\left[(r_i - d_{iO})/b_i\right]\mathbf{n}_{iO} + kg(r_i - d_{iO})\mathbf{n}_{iO} + \kappa g(r_i - d_{iO})(\mathbf{v}_i \cdot \mathbf{t}_{iO})\mathbf{t}_{ij}\,.$$

The model parameter values have been specified to reflect diversity in the pedestrian population. Thus, pedestrians have uniformly distributed masses in the interval [65 Kg – 85 Kg], the preferred speed is approximately Gaussian distributed with a mean value of 1.3 m/s and a standard deviation of 0.3 m/s representing an urgency situation, and the acceleration time τ_i is also Gaussian distributed with a mean of $1s$ and a standard deviation of 0.2 m/s. The physical interaction parameters take values $k = 1.2 \cdot 10^5$ kg s^{-2} and $\kappa = 2.4 \cdot 10^5$ kg m^{-1}s^{-1} as proposed in Helbing et al. (2000), while the interaction strength is fixed to $A_i = 2 \cdot 10^5$ N and the influence area to $\lambda_i = 0.1$.

In our research, we assume that some external mechanism operates during evacuation to psychologically influence the dynamic response of pedestrians to the presence of other pedestrians. This influence is captured through the psychological parameter b_i, which is defined in our work as a space–time dependent variable $b_i(s,t)$ subject to external control. It can be also interpreted as the induced safety distance to other pedestrians (distance-keeping).

A generic goal evacuation scenario has been considered, with a 100 m long and 30 m wide corridor giving access to three exit gates (Fig. 2). During an evacuation, pedestrian inflow in the left area traverse a control area where pedestrian behavior is modulated. Close to the exit gates, the sensing area provides real-time density-based measurement values to the behavioral control system.

2.1. Evacuation efficiency

Pedestrian density has been typically used to evaluate crowd motion and composition and judge critically intrinsic risk in pedestrian crowds, though this is not the only relevant property (Feliciani and Nishinari, 2018). Here, we have used the local density and "crowd pressure" metric formulated by Helbing et al. (2007) to estimate the risk of the pedestrian crowd from a macroscopic perspective, defined as the local variance of velocity multiplied by the corresponding density given by:

$$Cp = \rho \cdot Var(v)$$

where ρ is the local density and Var(v) is the local temporal variation in pedestrian speed. Using this quantity Helbing et al. identify regions responsible for "crowd turbulence" (Helbing et al., 2007; Helbing and Mukerji, 2012). To determine the local density we apply Voronoi diagrams (Steffen and Seyfried, 2010; Zhang and Seyfried, 2013) to the microscopic data from simulations. Fig. 3 shows a Voronoi diagram in our evacuation scenario, where each region represents a single pedestrian, including all locations closer to the pedestrian than to any other pedestrian. Each point in the diagram shows the location of a pedestrian i at time instant t_k. Being Ω_i the area available to the pedestrian, we modify the standard Voronoi diagram in such a way that $A = \cup_{i=1}^{n}\Omega_i$, where A is the walking area (Hoogendoorn et al., 2018). For a given diagram, the pedestrian specific density is defined as:

$$\rho_i(t_k) = \frac{1}{|\Omega_i(t_k)|}.$$

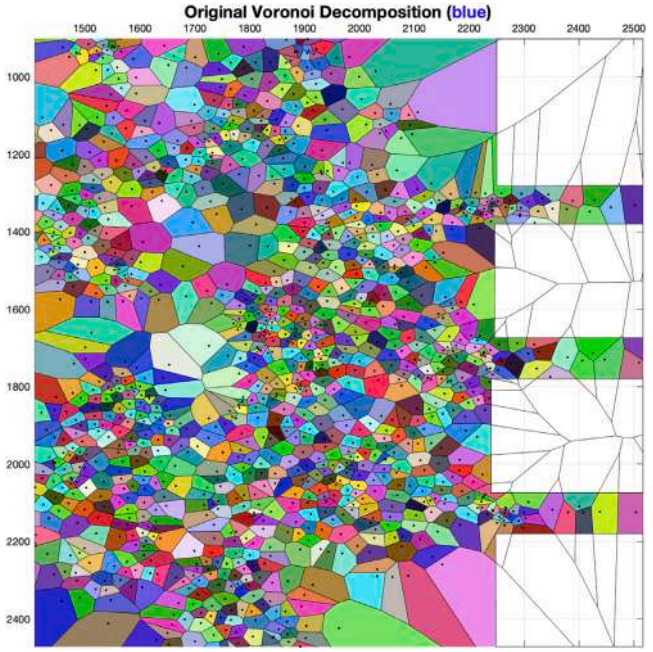

Fig. 3. Example of Voronoi diagrams.

The average density of pedestrians at time t_k is computed by averaging the pedestrian specific densities:

$$\rho(t_k) = \frac{1}{n}\sum_{i=1}^{n} \rho_i(t_k),$$

where n is the population of agents in the evacuation area. If our aim is to calculate the flow vs. density pedestrian Macroscopic Fundamental Diagram (pMFD) (Saberi and Mahmassani, 2014; Geroliminis and Daganzo, 2008; Geroliminis and Sun, 2011), the region-wide instantaneous mean speed is determined by averaging the speed of all pedestrians at time t_k. Intuitively, from a wide region perspective, an evacuation scenario will be more or less dangerous, dependent on the number of flow-density points above the threshold close to the capacity value.

2.2. Distance-keeping and crowd dynamics

We consider 5000 pedestrians entering the corridor at a rate of 1000 peds/min, where the distance-keeping value induced in the control area is set to a constant value. Let us first start by setting $b_i(s,t) = b_i(\text{ControlArea},t) = 0.08$ m, which means that the distance-keeping parameter for the entire evacuation scenario s is 0.08 m (i.e., the control area is off). Fig. 4 shows the formation of large queues at the exits as evacuation proceeds, with lower congestion levels upstream. The density and crowd pressure profile plots at the top of Fig. 6 show that the density increases linearly up to $3.5[\text{m}^{-2}]$ for 6 min and then decreases progressively at a slower rate. Crowd pressure follows a faster growth pattern, but once congestion reaches its maximum value, it exhibits a steep decline due to low spatial variance of velocity in the output queues. The fundamental diagram (upper right plot of Fig. 6) shows hysteresis due to the loading and unloading processes in the exit queues, which is an indication of nonlinear response.

Now, we activate the control area by setting $b_i(s \neq ControlArea,t) = 0.08$ m and $b_i(\text{ControlArea},t) = 0.5$ m. Fig. 5 shows the formation of upstream queues and low congestion levels in the sensing area (see Fig. 6). Thus, the density measurements exhibit a decreasing noisy pattern. As expected, the fundamental diagram is linear since the pedestrian flow at the exits does not reach the maximum capacity of the exits.

From these results, we conclude that a trade-off may exist between evacuation time and where we prefer congestion to be displaced. With the control area activated at a constant high value, we avoid congestion and potential dangerous situations associated with crowd pressure at the exits, shifting the congestion zone upward and significantly increasing the evacuation time.

Our objective is to obtain the optimal control actions $b_i(ControlArea,t)$, such that from the real-time measurements in the sensing area, we can induce the pedestrian behavior in the control area that allows an adequate balance between evacuation time, congestion, and safety. Therefore, it is assumed that the controller adaptively induces the b_i values that define the range of influence (distance-keeping) considered by each pedestrian. In this work, we have restricted ourselves to the homogeneous and direct mapping of b_i values. At each instant t all pedestrians in the control area receive the same control information that impacts the b_i values directly and homogeneously. Note, however, that in an actual control system deployment, the dependence between pedestrian perception, behavior, and the information provided by the control system may take more complex forms.

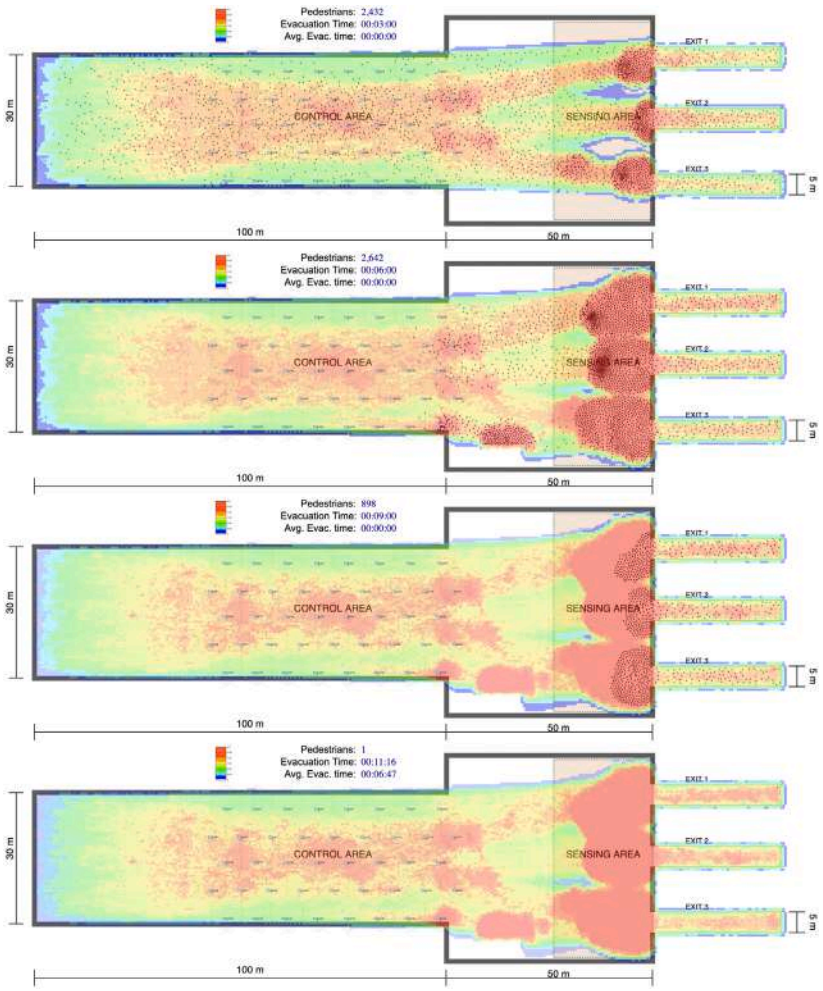

Fig. 4. Evacuation of 5000 pedestrians when the control area is off. The parameter $B_i(s,t)$ is set to 0.08 m for all the pedestrians during all the evacuation process. The heatmaps represent the cumulative density at each point, measured using a smoothing approach.

The proposal in this paper is to use the System Identification methodology to model crowd dynamics. We aim to exemplify how this methodology can be applied in a general context to design behavioral controllers for pedestrian evacuation.

3. System identification for modeling crowd dynamics

System Identification (Ljung, 1999; Ljung et al., 2020) "deals with the problem of building mathematical models of dynamical systems based on observed data from the system". It requires measuring input and output signals from the system in the time or frequency domain, selecting a model structure, estimating the adjustable parameters in the model structure, and evaluating the estimated model to see if it is acceptable for our application needs.

System identification is an iterative process, where models with different structures are identified from the data and compared. A system identification workflow might include the following stages[3]:

1. *Data processing* to examine time- and frequency-domain behavior; analyze delays, trends and offsets, feedback and signal excitation levels; and preprocess data by removing offsets and trends, interpolating, filtering or resampling.
2. *Linear or nonlinear model identification* to build adequate models: impulse-response, frequency-response, process models, state-space, transfer function, nonlinear black-box, or ordinary differential equations.
3. *Model validation* by comparing model response to measured response.

[3] https://www.mathworks.com/help/ident/gs/system-identification-workflow.html.

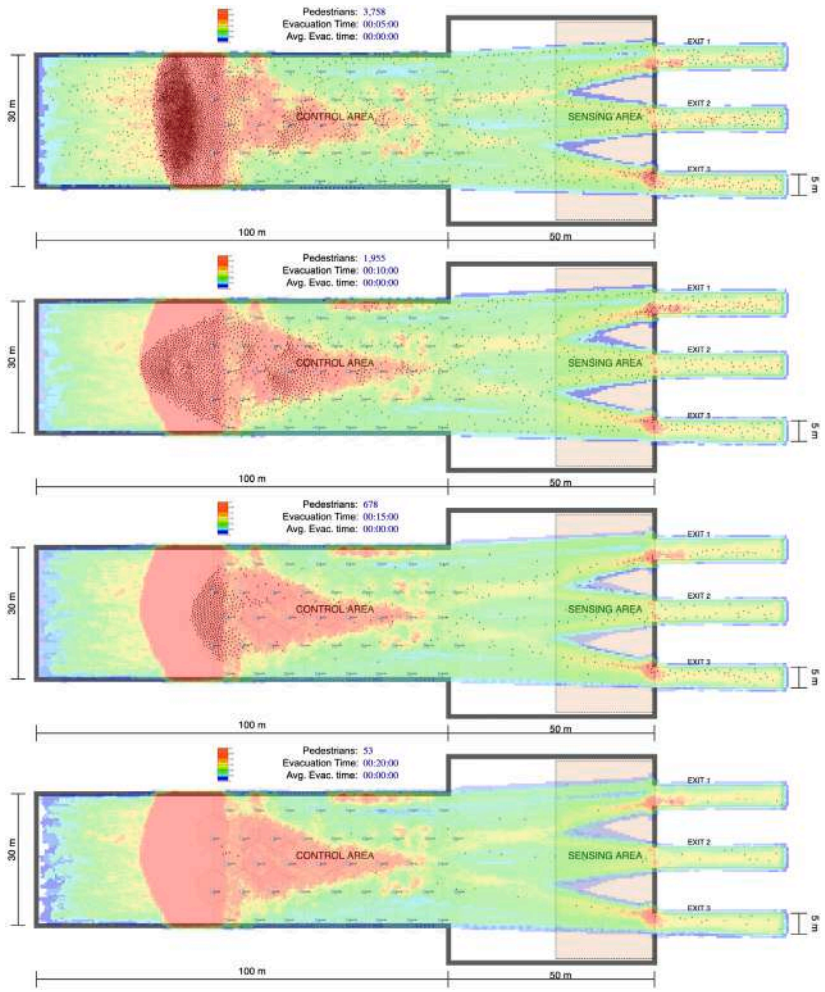

Fig. 5. Evacuation of 5000 pedestrians when the control area is on. The parameter $b_t(ControlArea, t)$ is set to 0.5 m in the Control Area, while it takes the value 0.08 m outside. The heatmaps represent the cumulative density at each point, measured using a smoothing approach.

4. *Postprocessing* to transform between continuous and discrete domains or model representations, concatenate and merge models, or linearize nonlinear models.
5. *Use identified models* for: control system design, online estimation, simulation and prediction.

The following sections describe in detail each of these phases of the system identification process, which are aimed at modeling the pedestrian evacuation dynamics for behavioral control system design.

3.1. Generating input–output data: experiment design

Our experimental design is built upon the evacuation simulation scenario described in Section 2, where two signals are identified: the input signal $b(ControlArea, t)$ (in short $b(t)$), and the output signal $d(t)$ that measures the pedestrian density in the sensing area. To obtain a good model of the crowd dynamics, we must measure data that accurately reflects the system's dynamic behavior. In general, the experimental design to obtain output data from input data must:

- Use inputs that excite the system dynamics adequately.
- Measure data long enough to capture essential time constants.
- Set up the data acquisition with a good signal to noise ratio (for data obtained from physical systems).
- Measure data at appropriate sampling intervals or frequency resolution.

Identification data can be obtained by measuring input and output signals from a physical system or generating data using simulation. In our experimental setup, which corresponds to a simulated Single-Input Single-Output (SISO) dynamic system, we generated the

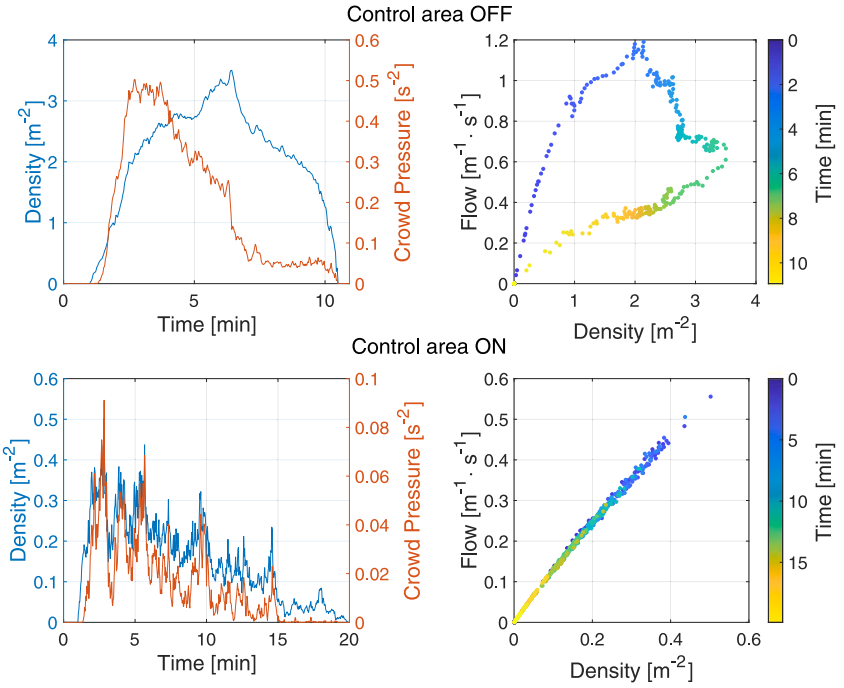

Fig. 6. Density profiles, crowd pressure profiles, and fundamental diagrams in the sensing area for evacuations with the control area OFF and ON. The two-Y-axes plots represent the density and crowd pressure profiles. The fundamental diagrams include color codes showing evacuation time.

input data $b(t)$ with the desired characteristics, and the evacuation simulation experiment generated the corresponding output density signal $d(t)$.

In constructing the input signal $b(t)$, we first determined the desired operating range in terms of amplitude and frequency. We thus configured an amplitude range between 0.08 and 1.5 m, and a minimum period of 40 s (0.025 Hz), which we considered reasonable in a behavioral control strategy that would avoid rapid changes. With this configuration, for the evacuation of a population of 5000 pedestrians, and injecting a random Gaussian noise signal $b(t)$, we obtained evacuation times of 30 minutes. Due to the bandwidth of the input signal $b(t)$ and the time constants of the system, it was observed that the variations we could induce in the output were limited for a single evacuation experiment. Thus, we opted for an identification data design based on multi-experiment data that would excite the system more representatively. Four batches of input–output data were generated from four sinusoidal $b(t)$ input signals used in four evacuation experiments.

Fig. 7 shows the input signals, and the corresponding outputs obtained $d(t)$. In all the experiments, the inputs correspond to sinusoidal signals shaped from three sinusoids with frequencies equally spread over the specified passband ($0 - 0.025$ Hz), for a sampling frequency of 2 s. The differences between the different input signals lie in the periodicity imposed and how the three sinusoids are phased to generate different excitation patterns. Two $b(t)$ signals are constructed from 125 samples repeated for eight periods (top row in Fig. 7). An iterative process was applied to generate the phase shifts that searched for the lowest signal spread, which is the difference between the minimum and the maximum value of the signal over all samples. In the bottom row, the signals $b(t)$ shown are generated from 250 samples with four periods (left) and 500 samples with two periods (right). The lowest signal spread algorithm was not used in the latter two experiments.

When dealing with multi-experiment datasets for model estimation, we may group them in a single dataset, or it is also possible to merge models after estimation by averaging out independently estimated models. We opted by merging models after estimation for two reasons. First, the evacuation dynamics are slightly different at the beginning, intermediate, and final stages, making the system not strictly time-invariant. Integrating the different input datasets would strictly require focusing on each stage and removing the non-related samples. Thus, this strategy would imply obtaining three separate models. Second, we assume that the signal-to-noise ratios for the experiments are different, which means merging models after estimation works better than merging the datasets. For merging different models, all must be of the same structure, just differing in parameter values and covariance matrices obtained in its estimation. The parameter vector is a statistically weighted mean of the parameters of the different models (using the covariance matrices to determine the weights).

To model a dynamic system with a small bias, it is necessary to use a large model parameter set, while a small parameter set will show a bad fit, with a smaller variance of the identified parameters. The “best” model will be somewhere in the middle and will depend on the intended use of the model. It may not be necessary for control systems’ design to obtain high-order models that capture all behavior in all detail. The solution to avoid overfitting is cross-validation, which means that different datasets are used to identify and validate. Thus, posterior comparisons of the identified models will be made over validation datasets. Our validation

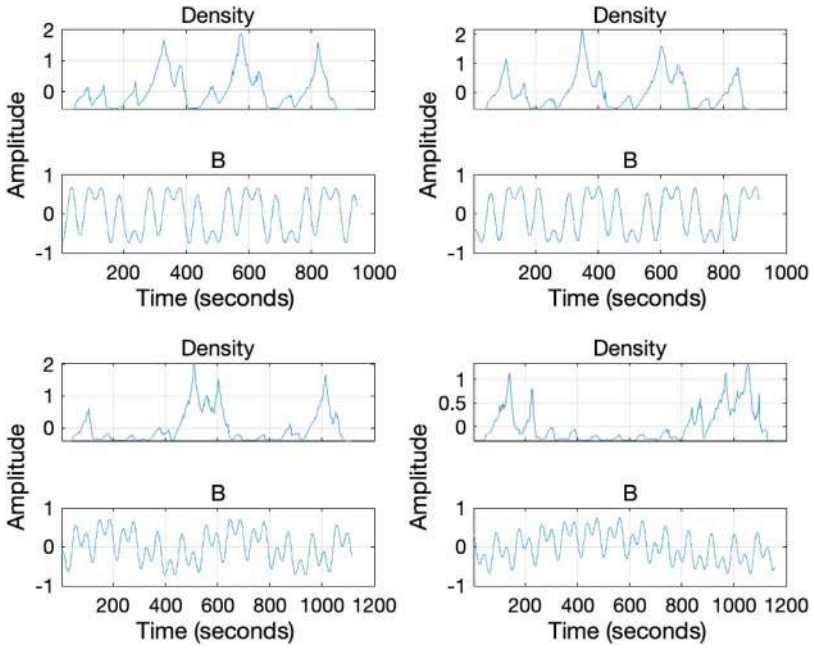

Fig. 7. Input–Output ($b(t) - d(t)$) signals for estimation. Each subplot shows the $b(t)$ input signal and the corresponding $d(t)$ output signal obtained for an evacuation experiment. Mean values have been removed.

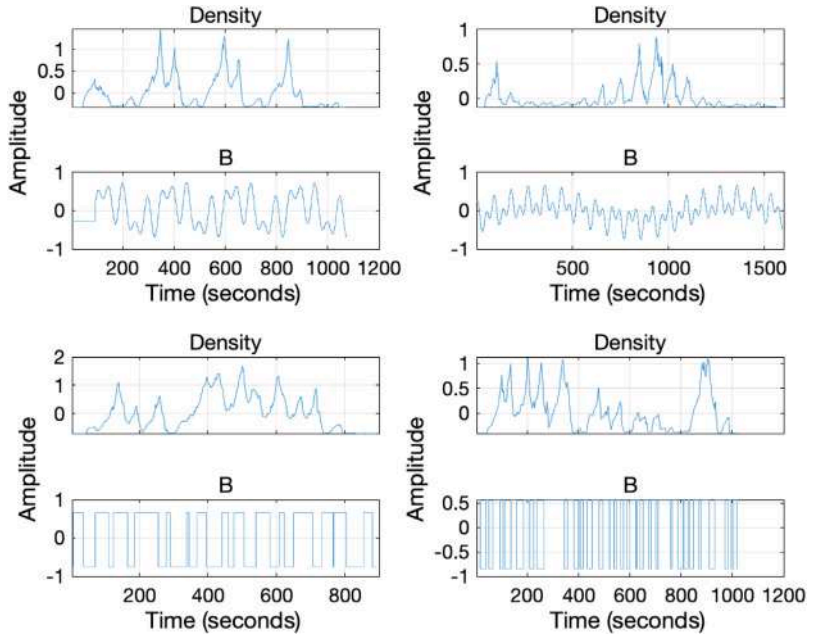

Fig. 8. Input–Output ($b(t) - d(t)$) signals for validation. Each subplot shows the $b(t)$ input signal and the corresponding $d(t)$ output signal obtained for an evacuation experiment. Mean values have been removed.

datasets are shown in Fig. 8. First row shows two input sinusoidal signals with eight and four periods, with and without using the lowest signal spread algorithm. The second row defines two random binary input signals generated with an eight-order Butterworth lowpass filter in the frequency bands $0 - 0.025$ Hz and $0 - 0.05$ Hz.

3.2. Non-parametric system identification

The first step in the whole identification procedure is typically non-parametric identification. The non-parametric models, based on correlation and spectral analysis, may give insight into the specific dynamic behavior of a system, helping detect erroneous

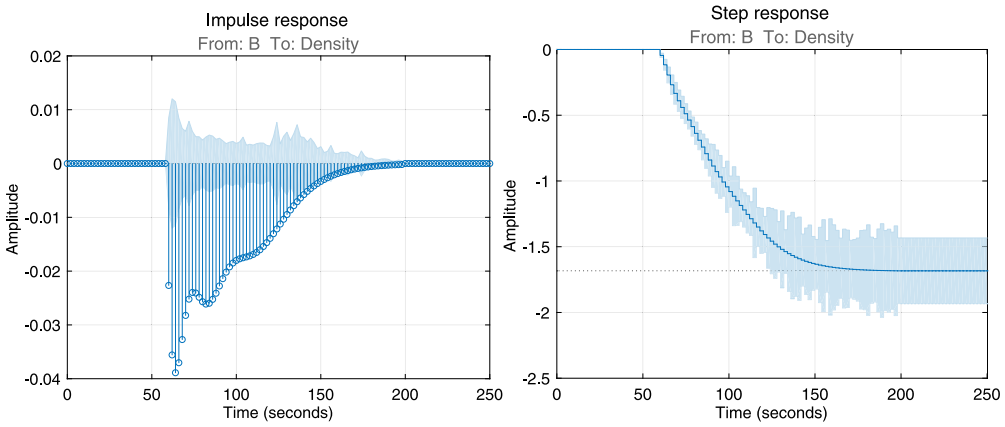

Fig. 9. Impulse and step responses from estimation datasets, with confidence regions of 3 standard deviations.

results in a parametric identification. Note, however, that our goal is to obtain and compare parametric models, better suited for the design of closed-loop controllers, providing compact models with fewer parameters than "non-parametric" models.

3.2.1. Correlation analysis

Correlation analysis estimates the impulse response of dynamic systems as a finite impulse response model from the data (FIR). Impulse response is the output that results when the input $b(t)$ is an impulse $\delta(t)$, such that $\delta(t) = 0, \forall t > 0$, and $\delta(0) = 1$. Before estimating the impulse response model, we took into account the transport delay between the input and output visible in the data. Thus, our first task was to estimate the transport delay in samples from $b(t)$ to $d(t)$. This estimate is based on iteratively comparing the fitness of AutoRegressive eXogenous models (ARX) with different transport delays nk and orders $na = nb = 2$:

$$d(t) + a_1 d(t - T) + \cdots + a_{na} d(t - naT) = c_1 b(t - nkT) + \cdots + c_{nb} b(t - nbT - nkT + T)$$

The best fitness was obtained for a transport delay of $nk = 30$ samples, which corresponds to 60 s for our $T = 2$ s sampling period. This result is consistent with the evacuation simulations, in which it can be observed that pedestrians reach the sensing area in 60 s on average.

Next step was to estimate the coefficients of the FIR model of order n, assuming the computed transport delay. To determine the value n, we used persistence of excitation analysis on the input data (Ljung, 1999). The estimation algorithm uses the least-squares method with regularization, using tuned and correlated kernel complemented with an autoregressive model of order 5, without prewhitening filter. Regularization is essential when the order is large to reduce the variance of the estimate. Fig. 9 shows the impulse and step responses for the identified nonparametric model of order $n = 70$ obtained, exhibiting a rise time around 60 s.

Typically, the correlation analysis includes computing the empirical impulse response for negative lags to verify feedback effects, i.e., the output may partially determine input data. This analysis is critical when the input data has been obtained through direct measurements. However, $b(t)$ is based on synthetic data. We also assume that pedestrian behavior is not influenced by the density measurements at the exits.

3.2.2. Frequency-response model

Assuming our system is linear time-invariant (LTI), non-parametric identification in the frequency domain aims to obtain the frequency response, so the Bode plot. The Bode diagram obtained for 128 frequency points is shown in Fig. 10, using the Blackman–Tukey spectral analysis method. This method computes the Fourier transforms of the covariances and cross-covariances from $b(t)$ and $d(t)$. The Bode plot shows a uniform response up to $0.05 - 0.08$ rad/s, while the first resonance frequency appears around 0.2 rad/s. It appears that behavioral variations with periods shorter than 90 s induce a decreasing density at the output, at the cost of an increase in the upstream crowd pressure. Our interpretation is that the rapid change in $b(t)$ favors velocity variations and shockwaves that slow down the upstream pedestrian flow.

3.2.3. Detecting nonlinearity

The evacuation dynamics can be significantly different depending on the time elapsed since its onset. Mainly, three phases could be distinguished corresponding to the start of the evacuation, a steady phase, and the final discharge, which could indicate, as mentioned before, a violation of the time invariance assumption. However, to design a controller, we found it reasonable to integrate the three phases in the modeling, considering that the duration and contribution to congestion of the initial and final stages are lower than in the steady stage. On the other hand, hysteresis in density measurements for low values of $b(t)$, which induce congestion at the output, is an indicator of nonlinearity, at least for specific frequency and amplitude values.

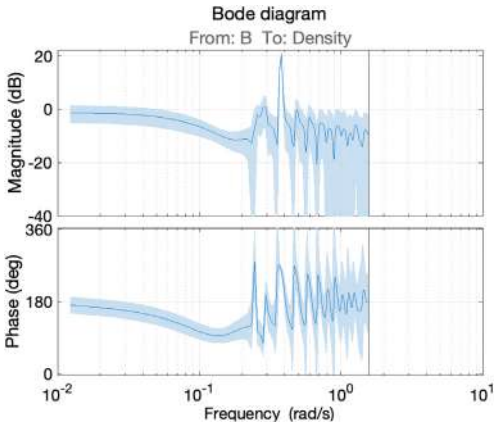

Fig. 10. Bode diagram with confidence regions of 3 standard deviations.

Experimentally, we can detect indications of nonlinearity from the estimation data by testing whether a nonlinear ARX model produces a better estimate than its linear version. A nonlinear ARX model, instead of the weighted sum of regressors representing a linear mapping, employs a more flexible form of mapping function F, such that the inputs to F are regressors of the model. These regressors r_n can be lagged input and output variables or more complex nonlinear expressions of these delayed variables. The general structure of a nonlinear ARX model is as follows:

$$\hat{d}(t) = F(r_1, r_2, r_3, \ldots) = L^T(r(t)) + d + g(Q(r(t))) + E(t)$$

where r_n generally represents any nonlinear regressor that can combine delayed input and output variables (e.g., $r_1 = d(t-1)^2 * b(t)$), $L^T(r)+d$ is the output of a linear function block, $g(Q(r))$ is the output of the nonlinear function block, where Q is a projection matrix that makes the calculations well-conditioned, and $E(t)$ is the remaining error that is unexplained by the model and is typically white noise. The exact form of $F()$ depends on the nonlinearity estimator used: wavelet networks, sigmoid networks, treepartition, neural networks, or a simple linear mapping.

Using the nonlinear ARX structure with treepartition, we estimated the portion of the data explained by the linear function L, the nonlinear function g, and the remaining error. In the identification of the nonlinear ARX model, a line combination of least squares search algorithms (subspace Gauss–Newton, Levenberg–Marquardt, Adaptive subspace Gauss–Newton and steepest descent) were used to minimize the error between the predicted model output and the measured output. For a delay of 30 samples and orders $na = 5$ and $nb = 5$, feeding the nonlinear block with all the regressors, we found an estimated noise standard deviation of 0.038219, while the estimated standard deviation of the data explained by the nonlinearity was smaller and equal to 0.033783, which is an indication of a weak nonlinearity. In this case, a good strategy is to try fitting linear parametric models first, and if the models do not adequately reproduce the measured output, use a nonlinear model.

3.3. Parametric system identification

Parametric system identification is usually used to capture the essential dynamics of a system with a finite (and limited) number of parameters. The application of these models includes process simulation, prediction, or controller design. For our purposes, a good model should focus on simulating the most crucial dynamic behavior to design a stable controller accurately.

The starting point for linear system identification is to assume that the output $y(t)$ of a real system can be described as a linear combination of input $u(t)$ and system $G_0(z)$, plus a noise signal $v(t)$:

$$y(t) = G_O(z)u(t) + v(t)$$

Noise signal includes the output from non-measured inputs, measurement noise, and non-linearities, such that such sources are independent from $u(t)$. It is assumed that the disturbance signal $v(t)$ is the output of a linear system whose input is a white noise source $e(t)$ with variance σ_e^2, and thus, the structure of the system is:

$$y(t) = G_O(z)u(t) + H_O(z)e(t). \tag{3}$$

where, it should be verified that the input signal and disturbance are uncorrelated.

The identification process requires choosing a model structure. Here we can distinguish between gray-box modeling, applicable when the structure is derived from known physical principles, or black-box modeling, when the objective is to fit the data independently of a particular mathematical structure. In our evacuation scenario, the dynamics of the system was not known a priori. Therefore, black-box modeling was used, in which, by trial and error, we estimated the parameters of various structures and compared the results. The objective was to choose the simplest structure that would provide a good fit.

Table 1
Overview of polynomial structures.

Model	Equation	$G(z)$	$H(z)$
ARX	$A(z)y(t) = B(z)u(t-nk) + e(t)$	$\frac{B(z)}{A(z)}$	$\frac{1}{A(z)}$
ARIX	$A(z)y(t) = B(z)u(t-nk) + \frac{1}{1-z^{-1}}e(t)$	$\frac{B(z)}{A(z)}$	$\frac{1}{A(z)}\frac{1}{1-z^{-1}}$
ARMAX	$A(z)y(t) = B(z)u(t-nk) + C(z)e(t)$	$\frac{B(z)}{A(z)}$	$\frac{C(z)}{A(z)}$
ARIMAX	$A(z)y(t) = B(z)u(t-nk) + \frac{C(z)}{1-z^{-1}}e(t)$	$\frac{B(z)}{A(z)}$	$\frac{C(z)}{A(z)}\frac{1}{1-z^{-1}}$
Box–Jenkins (BJ)	$y(t) = \frac{B(z)}{F(z)}u(t-nk) + \frac{C(z)}{D(z)}e(t)$	$\frac{B(z)}{F(z)}$	$\frac{C(z)}{D(z)}$
Output-error (OE)	$y(t) = \frac{B(z)}{F(z)}u(t-nk) + e(t)$	$\frac{B(z)}{F(z)}$	1

Typically, the identification process starts by testing all or some of the simplest linear black-box structures: transfer functions, linear ARX models, and state-space models, which can be estimated using noniterative algorithms due to the linearity in the parameters. If these simple model structures do not produce good results, typical actions include: increasing the order of the same linear models, explicitly modeling the noise treating the disturbance as the output of a linear system driven by white noise source, using different linear model structures, or using a nonlinear model structure. Interestingly, explicitly modeling the additive disturbance may help to improve the accuracy of the measured component and improve the models when used for predicting future response values.

Our research started with the parametric identification process of linear structures. We focused on input–output polynomial models, starting with the simplest input–output polynomial model, the linear ARX model. The linear ARX model orders with the best performance can be used as an initial guess for further modeling.

3.3.1. Identification of input–output polynomial models

A polynomial linear model for a SISO system has the following general form:

$$A(z)y(t) = \frac{B(z)}{F(z)}u(t-nk) + \frac{C(z)}{D(z)}e(t),$$

where $A(z)$, $B(z)$,... are polynomials in z^{-1} (time-shift z operator) with orders na, nb, ..., and nk is the input transport delay. We may opt for different configurations of polynomial models, which essentially differ in how the polynomials are included and build the estimated $G(z)$ and $H(z)$ transfer functions. Table 1 gives an overview of the most frequently used models. In ARX, the noise and the dynamics model are coupled, being the large advantage the linearity in the parameters, which allows fast execution of prediction error identification methods. ARIX extends ARX including an integrator, which is useful when the disturbance is not stationary. ARMAX introduces $C(z)$ polynomial, which computes a moving average of white noise. This model is useful when the disturbances are at the input. ARIMAX extends ARMAX including an integrator, useful as in ARIX when the disturbance is not stationary. BJ model structure provides the most general structure, making no assumptions about common parts in the system and noise components. Finally, the OE model assumes that the disturbance is a white noise signal, so the noise model is not estimated.

To estimate the parameters θ of different structures using the input–output measured data $u_{meas}(t)$ and $y_{meas}(t)$, the Prediction Error identification Method was used (PEM). In PEM, the prediction error $\varepsilon(t)$ can be defined as:

$$\varepsilon(t,\theta) = H^{-1}(z,\theta)[y_{meas}(t) - G(z,\theta)u_{meas}(t)],$$

which is derived from Eq. (3) under the assumption that the disturbance is generated from white noise. Thus, identifying a model from the measured data $y_{meas}(t)$ and $u_{meas}(t)$ requires defining the model structure and the identification criterion. In practice, the following two criterions are mostly used: least-squares criterion trying to minimize the cost function $V_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}\varepsilon(t,\theta)^2$, or the so-called Instrumental Variable (IV) techniques, trying to make the prediction error an uncorrelated signal. These 'prediction' focused methods provide optimal statistical properties, optimized for output prediction applications, favoring a model fit at high frequencies. Alternatively, 'simulation' focused identification approaches approximate the model dynamics $G(z)$ in a norm given by the input spectrum, with the noise model $H(z)$ estimated using a prediction error method while the dynamics model is kept constant. 'Simulation' focused estimation is optimized for output simulation applications.

From an application perspective, in 'prediction' oriented applications, output measurements are available, which are used to estimate step-ahead output values. Some or all of the $y(t-n)$ values used to obtain $y(t)$ are measured values. In 'simulation' applications, output measurements of the modeled system are not needed or available, being the model purpose to simulate the system. Since we had actual density measurements at the system output, our primary focus was on prediction.

3.3.2. Identification of linear ARX model

Our first step in the identification process was to estimate several linear ARX models with a range of orders $[1-10]$ for na and nb, and compare the performance of these models. The delay nk was varied from 1 to 5, and the input delay was fixed to 30 samples, for a sampling period of 2 seconds. The Akaike's Information (AIC) and Rissanen's Minimum Description Length criteria (MDL) were used for choosing the best model.

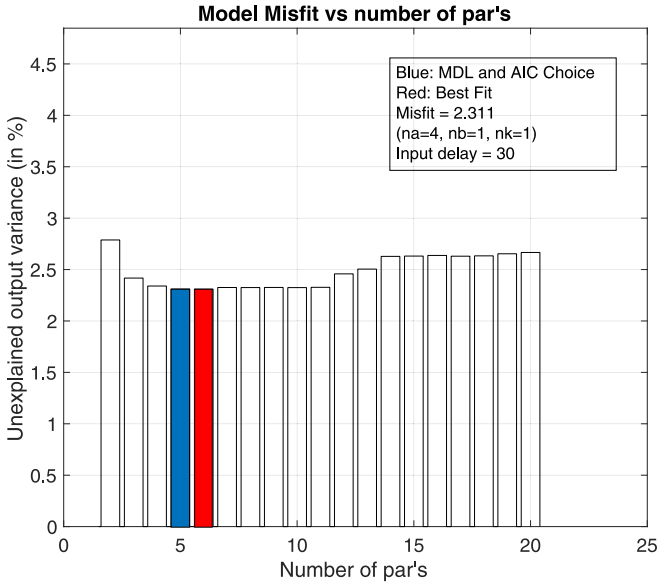

Fig. 11. ARX model structure selection. The horizontal axis shows the total number of parameters $na + nb$, while the vertical axis represents the proportion of the output no explained by the ARX model.

Fig. 11 shows the performance of the different estimated models in terms of prediction error (the sum of squares of the differences between the validation data output and the model one-step ahead predicted output). The ARX model with five parameters ($na = 4$, $nb = 1$, $nk = 1$) exhibited the best-fit using the AIC and MDL criteria (blue color), while the model with six parameters (red color) minimized the prediction error. Therefore, the chosen identified model has the following structure:

$$A(z) = 1 - 1.211z^{-1} + 0.09948z^{-2} + 0.02895z^{-3} + 0.1128z^{-4}$$
$$B(z) = -0.04142z^{-31}$$

The fit of the model to estimation datasets are (see datasets in Fig. 7): $[88.84\%, 88.58\%, 88.8\%, 83.87\%]$, with an Akaike's Final Prediction Error (FPE) of 0.003784.

3.3.3. Validation of the identified linear ARX model

The next step was to validate the estimated model by comparing the simulated and predicted model output to measured validation data (see validation datasets in Fig. 8). Fig. 12 compares the simulated response to validation data and the calculated NRMSE fitness value (Normalized Root Mean Square Error). It can be seen how the best fitness was found for the binary input signals (third and fourth validation datasets), while for the second validation signal, the fitness is significantly lower of (0.3035%), which may be an indication of non-linearity.

Fig. 13 compares the predicted response to validation data for 1, 5, and 10 steps ahead. Because the identification process focused on "prediction", and the predictions account for the nature of the perturbations, as expected, the results were significantly better than for the simulated response. Thus, 1-step predicted response fitness values are around 84%, while for 10-steps, the fitness ranges between 11.52% to 48.39%. Again, as for the simulation response, the fitness to the second validation signal is very poor, which suggests the weak non-linearity for some signal patterns.

After comparing the simulated and predicted ARX model output to validation measured data, we performed a residual test, which verifies whether the identified model can represent the data within the specified confidence levels or not. Residuals are the differences between the one-step predicted output from the model and the measured output from the validation dataset. Residual analysis checks if the prediction error $\varepsilon(t, \theta)$ is a realization of white noise and a stochastic process. First requirement can be verified by evaluating the autocovariance of $\varepsilon(t, \theta)$, while the second requirement can be verified by evaluating the cross-covariance between $\varepsilon(t, \theta)$ and the input signal $b(t)$. A good ARX model has the residual autocovariance function inside the confidence interval of the corresponding estimates, which means that the residuals are uncorrelated and represent a white noise signal. Also, a good model has residuals uncorrelated with past inputs, and so, cross-covariance function will be within the confidence interval.

Fig. 14 shows that the covariances calculated are within the 99% confidence region for all lags, confirming that the identified ARX model is correct. It can also be seen how the step response, impulse response, and Bode diagram for the estimated ARX model resemblance reasonably well the non-parametric models obtained in Section 3.2 (see Fig. 15).

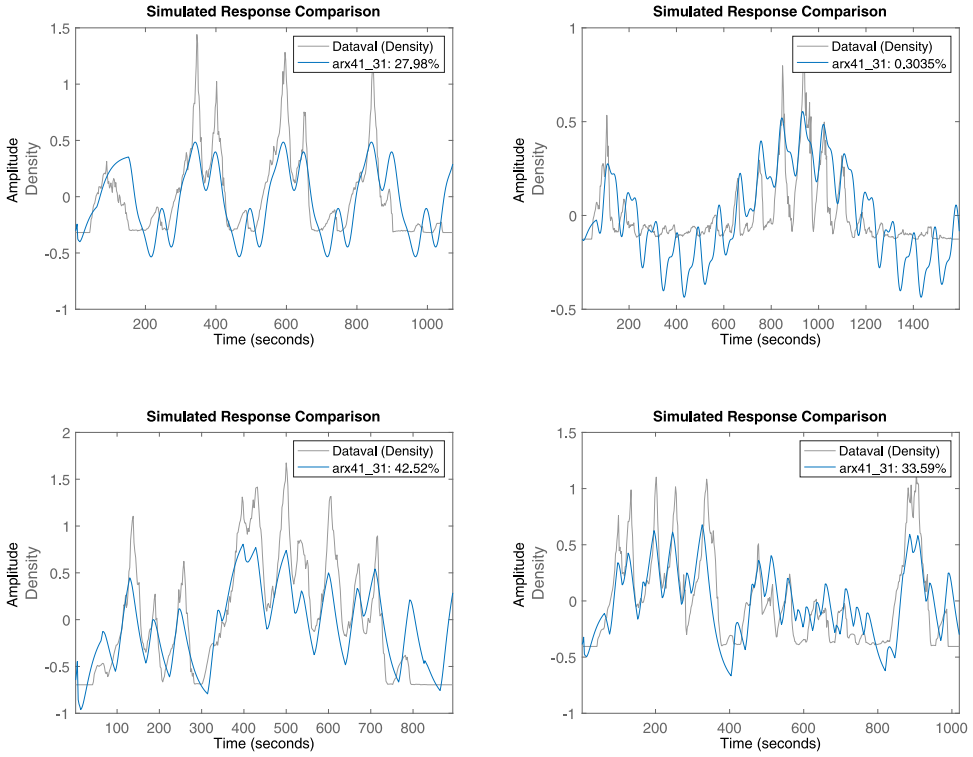

Fig. 12. Comparison of the simulated response of the identified linear ARX model to validation measured data. The percentages represent the NRMSE fitness values.

3.3.4. Identification with ARMAX, OE and BJ polynomial models

These polynomial models provide more complex structures, which have to be found with nonlinear optimization techniques due to the non-linearity in the parameters. We tested the following structures, taking as a basis the orders found for the ARX model:

$$ARMAX = [na = 4, nb = 1, nc = 2, nk = 1]$$
$$OE = [nb = 4, nf = 4, nk = 1]$$
$$BJ = [nb = 4, nf = 4, nc = 4, nd = 4, nk = 1]$$

The fitness obtained for the OE model was significantly worse than for the ARX model, while practically identical results were obtained for ARMAX and BJ. Therefore, it was decided to use the simpler linear ARX model in the controller design.

4. MPC controller design

Three possible alternatives were considered in the design of the control system: PID Control, Reinforcement Learning Control, and Model Predictive Control (MPC) (Camacho and Alba, 2013). We opted for linear MPC, which provides inherent flexibility and simplicity in the specification of disturbances, control and output variable constraints, and multiples control objectives. We hypothesized that the simple configuration of control variable constraints and perturbations makes MPC ideal for designing and deploying human behavior controllers, where particular performance and control action policies may be necessary.

A linear MPC controller uses linear plant (identified linear model of crowd dynamics for our evacuation scenario), disturbance, and noise models to estimate the controller state and predict p future plant outputs (future density values) at each control interval k. The controller searches for optimal control actions $b(k), b(k+1), \ldots, b(k+p-1)$ by using predicted densities and solving a quadratic programming (QP) optimization problem.

In the design of our MPC controller, we used the identified ARX model as Prediction Model and Plant Model in the Control Loop (see Fig. 16). Once the controller's performance has been evaluated by simulation, the Plant Model (Identified ARX model for modeling crowd dynamics) is replaced by the Pedestrian Simulator in the Control Loop to measure its performance in the "real" evacuation scenario. This process is iterative, so the controller can be refined by tuning the gains of the Kalman state estimator, modifying constraints or weights, control period, prediction horizon, control horizon, nominal values, scale factors, reference, and disturbance and noise models.

More in detail, the MPC controller in the evacuation scenario attempts to predict how known and unknown events affect the pedestrian density values in the sensing area. Our prediction model (see Fig. 17) uses the identified ARX model to estimate the crowd

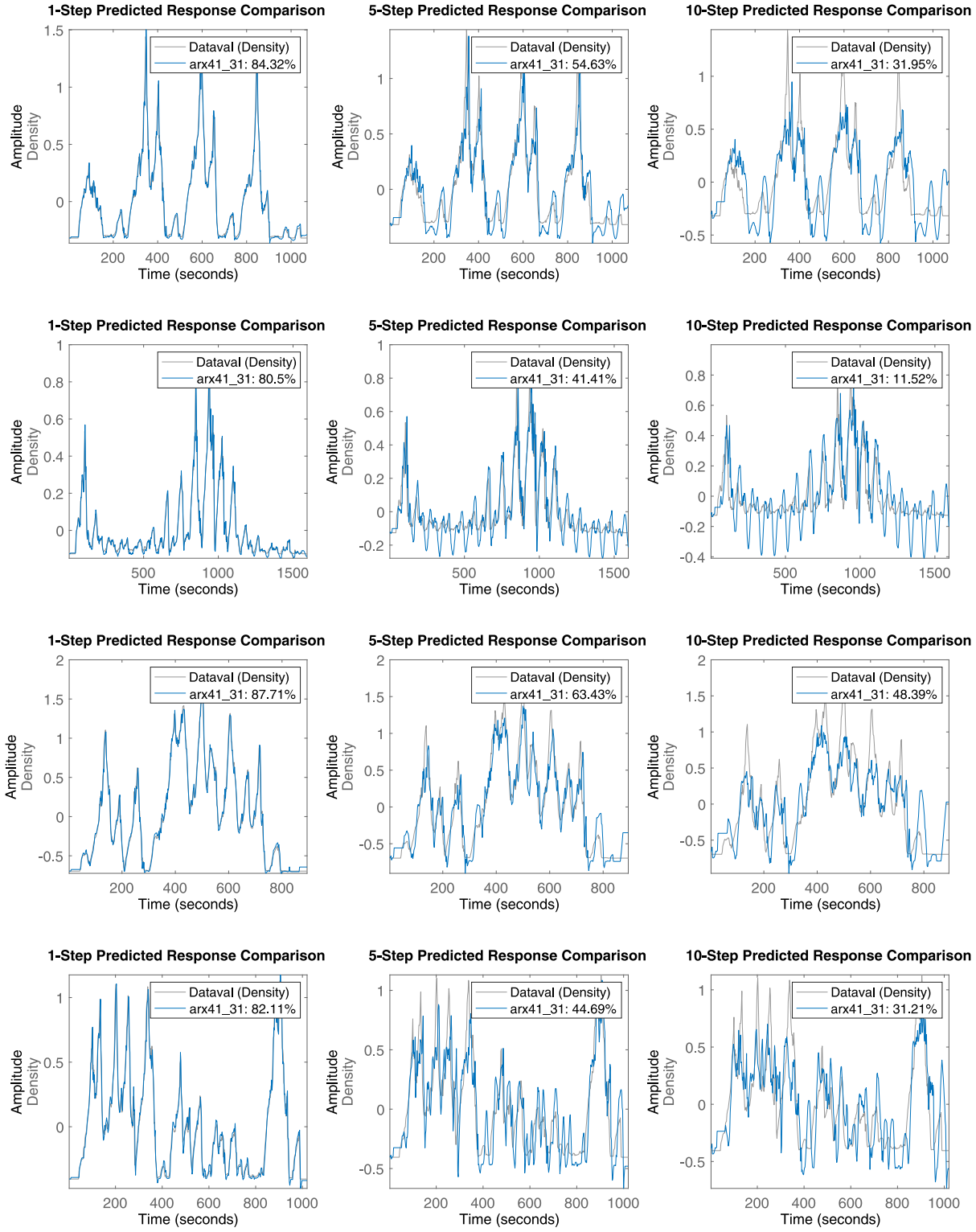

Fig. 13. Comparison of the predicted response of the identified linear ARX model to validation measured data. Different columns show 1, 5, and 10 steps predictions, respectively. The percentages represent the NRMSE fitness values.

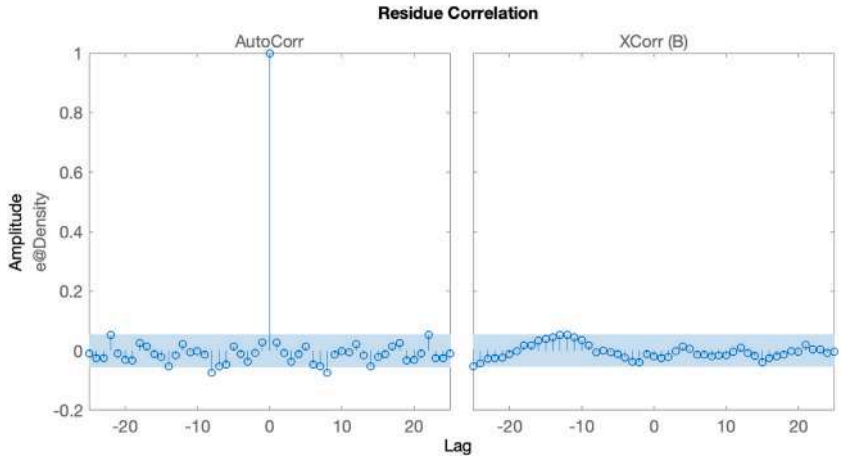

Fig. 14. Residual tests for the identified ARX model.

dynamics in the sensing area, whose two inputs are the manipulated variable b and white noise $w_{id}(k)$ to account for unmeasured input disturbances. Unmeasured input disturbances are independent inputs of which the controller has no direct knowledge and which it must compensate. The pedestrian crowd's particular characteristics at the entrance would be encompassed within the unmeasured input disturbances.

To the output $d_p(k)$ generated by the ARX model at control interval k, we add expected unmeasured output disturbances $d_{od}(k)$ and measurement noise $d_n(k)$, which can be modeled with different Output disturbance and Measurement noise models. The goal of these models is to anticipate deviations in pedestrian density measurements from the near-steady condition and improve disturbance and noise rejection. In our approach, the Measurement noise model is a static gain $N(z) = 1$, and so we directly add white noise $w_n(k)$, which generally provides adequate performance. For the Output disturbance model, we tested three different models whose input $w_{od}(k)$ is zero mean and unit variance white noise (see the Experimental results Section):

- A static gain for white noise disturbance. It is assumed that the impact of the disturbance is short-lived and requires a short-term controller response.
- A static gain in series with an integrator for a step-like disturbance. The impact of the disturbances is expected to last indefinitely, maintaining a roughly constant magnitude, requiring a more aggressive and sustained controller response.
- A static gain in series with two integrators for a ramp-like disturbance. The disturbances are expected to grow with time and require an even more aggressive controller response.

The QP optimization at each control interval, converts the linear MPC optimization problem to the general form QP problem $\underset{b_k}{Min}(\frac{1}{2}b_k^T H b_k + f^T b_k)$, subject to the linear inequality constraints $A\bar{b}_k \le c$. The QP solver used was the KWIK algorithm from Schmid and Biegler (1994), which determines the optimal set of moves $\bar{b}_k^* = \{b(k), b(k+1), \ldots, b(k+p-1), \varepsilon_k\}$ at each control interval k, from which the $b(k)$ value is extracted as the input to the Plant Model or Pedestrian Simulator until the next interval. For a predefined control horizon m, values from $b(k+m-1)$ to $b(k+p-1)$ are always equal in the optimization process.

The cost function J to optimize was defined as a multiobjective cost function, where the input is the candidate set of control actions $\bar{b}_k$, including the slack variable ε_k which quantifies violation of existing variable constraints:

$$J(\bar{b}_k) = J_d(\bar{b}_k) + J_{\Delta b}(\bar{b}_k) + J_\varepsilon(\bar{b}_k). \tag{4}$$

$J_d(\bar{b}_k)$ measures the performance of the pedestrian density reference tracking:

$$J_d(\bar{b}_k) = \sum_{i=1}^{p} \left\{ \frac{w_i^d}{s^d}[r(k+i|k) - d(k+i|k)] \right\}^2,$$

where $r(k+i|k)$ is the reference value for density at ith prediction horizon step, $d(k+i|k)$ is the predicted value of density at ith prediction horizon step, w_i^d is a tuning weight for density reference tracking performance at ith prediction horizon step, and s^d is a scale factor. The values of s^d and w_i^d are constant controller specifications.

$J_{\Delta b}(\bar{b}_k)$ measures b variations within the candidate set $\bar{b}_k$, favoring small adjustments:

$$J_{\Delta b}(\bar{b}_k) = \sum_{i=0}^{p-1} \left\{ \frac{w_i^{\Delta b}}{s^b}[b(k+i|k) - b(k+i-1|k)] \right\}^2,$$

where $b(k+i|k)$ and $b(k+i-1|k)$ represent control actions in adjacent control intervals, $w_i^{\Delta b}$ is a tuning weight for b movement performance at ith prediction horizon step, and s^b is a scale factor. Note that for $i = 0$, $b(k-1)$ represents the known control action from the previous control interval, and that for $i > m$, variations on b are zero due to the control horizon m imposed.

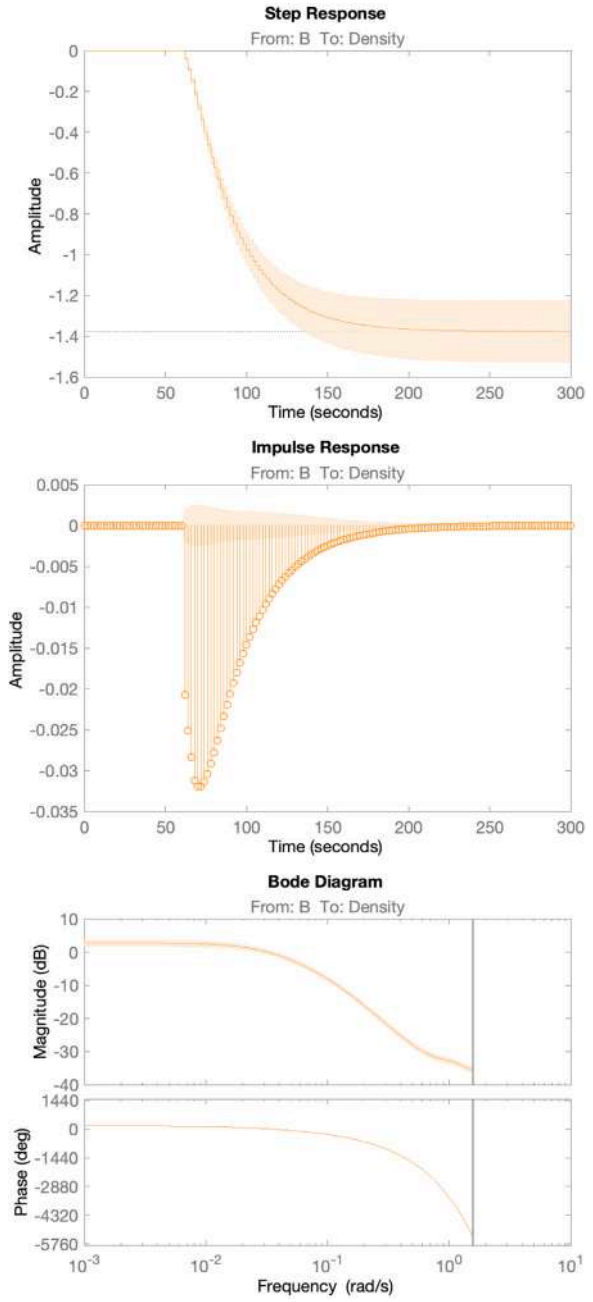

Fig. 15. Step response, impulse response and bode diagram for the estimated ARX model.

Finally,

$$J_{\epsilon}(\bar{b}_k) = \rho_k \epsilon_k^2$$

quantifies worst-case constraint violation, in which ϵ_k is the slack variable and ρ_k is a constraint violation penalty weight. Thus, the constraints on bounds for b and b increments were defined as follows:

$$\frac{b_{min}(i)}{s^b} - \varepsilon_k V_{min}^b(i) \leq \frac{b(k+i-1|k)}{s^b} \leq \frac{b_{max}(i)}{s^b} + \varepsilon_k V_{max}^b(i),\ i = 1...p$$

$$\frac{\Delta\ b_{min}(i)}{s^b} - \varepsilon_k V_{min}^{\Delta b}(i) \leq \frac{\Delta\ b(k+i-1|k)}{s^b} \leq \frac{\Delta\ b_{max}(i)}{s^b} + \varepsilon_k V_{max}^{\Delta b}(i),\ i = 1...p$$

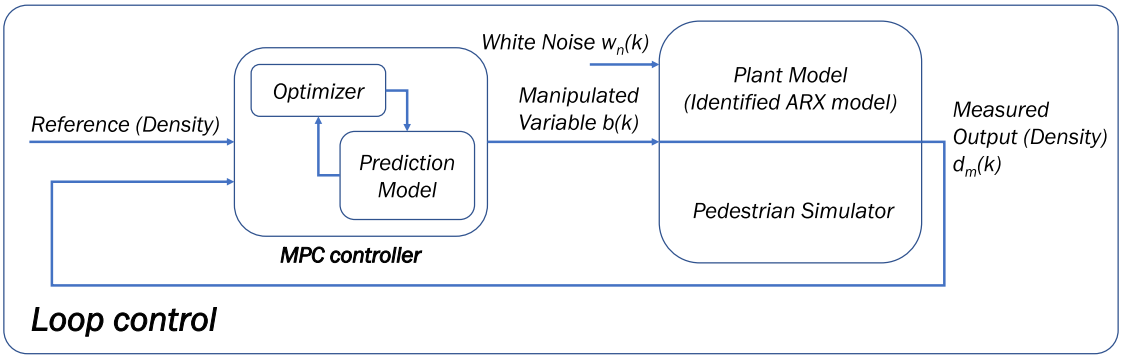

Fig. 16. MPC control loop for simulation and deployment.

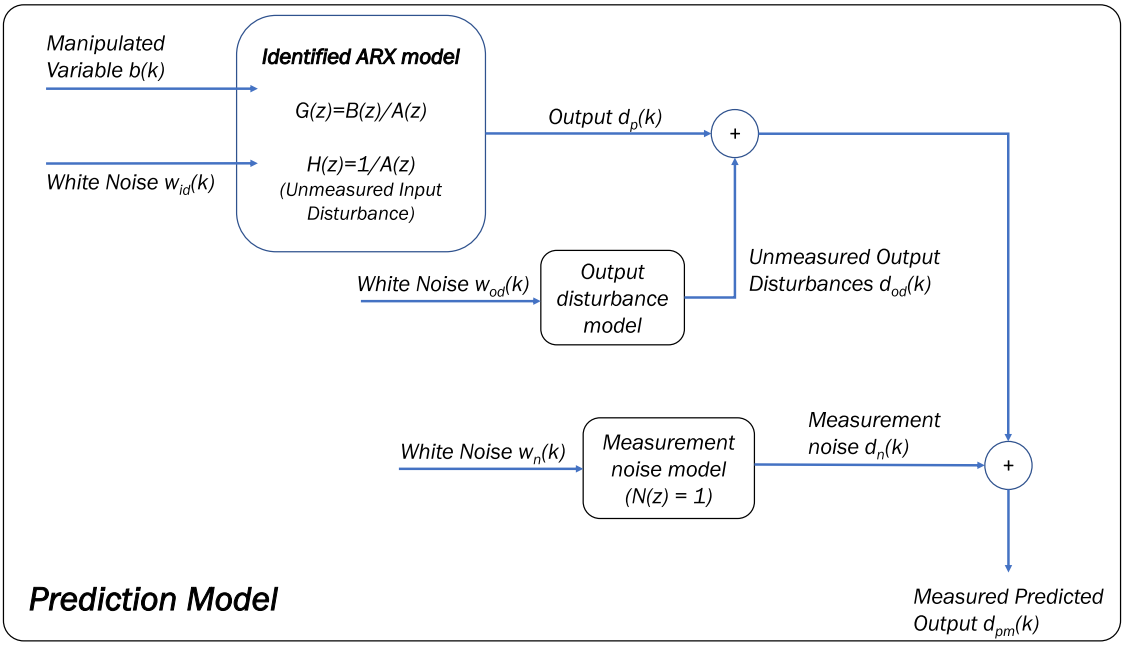

Fig. 17. Prediction model used in the MPC control loop.

The parameters $V^{b,\Delta b}_{min,max}$ are weights used for constraint softening, indicating the relative magnitude of a tolerable violation. $V = 0$ indicates that no violation is allowed (hard constraint), while values 0.05, 0.2, 1, 5 or 20 can be used as a guideline to apply constraint softening progressively. The scale factors s^b and s^d were used to generate dimensionless values for variables d, b and Δb, which improves controller tuning of weights and numerical conditioning. Typically, the scale factors are made equal to the span of the variables.

We did not include any performance measure on target $b(k)$ values in the cost function. However, in a more general form of a cost function $J(\bar{b}_k)$, it would be possible to measure the deviation of the input from a desired target value.

4.1. MPC parameterization

The first task in MPC parameterization was to choose prediction and control horizons p and m. For each control interval k, the MPC controller iteratively predicts p future density values for different combinations of p future control actions $b_k, \ldots, b_{k+p-1}$ until the optimized set of moves is found. Therefore, the optimized b value found at the beginning of the control horizon is used, and the others are discarded. Given that the system imposes a delay of 60 s, that in open-loop, the response time is around 40 s, and the control period is $T_s = 2$ s, we set the prediction horizon $p = 50$ ($pT_s \simeq 100$ s). Finally, we set the control horizon $m = 2$ to avoid b moves that might not affect the density outputs before the end of the prediction horizon. It is essential to avoid this effect when the dynamic system includes delays, which may lead to a singular QP Hessian matrix in the quadratic programming optimization.

The next step was to specify constraints on values and variations of $b(k)$. In our evacuation scenario, we found it reasonable to use the interval $0.08-2$ m, and a maximum change rate of $\pm 0.2m/T_s$, which approaches the bandwidth of the input data b used in the identification process (0.025 Hz). An interesting property of MPC controllers, as explained above, is that we can soften constraints, being a good practice not to include both hard bounds on values and increments. Thus, bounds on b increments were defined as soft constraints for which very small violations are allowed ($V^{\Delta b}_{min}(i) = V^{\Delta b}_{max}(i) = 0.05, \forall i$), reflecting physical limit on the rate of change induced over pedestrians, while the upper bound of b is a soft constraint $V^{b}_{max}(i) = 5, \forall i$, for which relatively large violations are admissible. The weight on the slack variable was set to $\rho_k = 100.000$ to measure constraint violation within the cost function (Eq. (4)). The scale factors were set accordingly to the span of the input and output variables to $s^b = 2$ m and $s^d = 1.5$ peds/m^{-2}.

We used a closed-loop cumulative performance index to evaluate the controller's performance, defined as the weighted Integral of the Square Error (ISE) between the density and its reference and the input b increments:

$$J_{ISE} = \sum_{k=1}^{N_s}((w^{pd} e_k^d)^2 + (w^{p\Delta b} \Delta b_k)^2),$$

where $N_s = 500$ is the number of control periods (simulation steps) used for measuring the performance of the controller, w^{pd} is the performance weight for the density reference tracking, e_k^d is the difference between density and its reference at each control period k, $w^{p\Delta b}$ is the performance weight for the b increments, and Δb_k represent the b increments at control period k. We gave more importance to density tracking than keeping the b increments low by making $w^{pd} = 1$ and $w^{p\Delta b} = 0.5$. Note that ISE performance weights are not related to the weights specified inside the MPC controller for the quadratic cost function to minimize.

For tuning the weights w_i^d and $w_i^{\Delta b}$ in the quadratic cost function $J(\bar{b}_k)$, we considered the typical approach that makes the weights constant over the whole prediction horizon and calculated sensitivities of the ISE controller performance metric to w_i^d and $w_i^{\Delta b}$. By trial and error, we found for $w_i^d = 1$ and $w_i^{\Delta b} = 0.01$ an optimal average ISE performance of 135.17 with typical deviation of 33.49, and sensitivities close to 0. In calculating the ISE, we injected white noise into the unmeasured input disturbance model of the plant.

Finally, we reviewed the design for potential stability and robustness problems for the three Output disturbance models considered in the predictor (see Section 4): static gain for white noise disturbance, static gain in series with an integrator for step-like disturbance, and static gain in series with two integrators for ramp-like disturbance rejection.

The first analysis was checking QP Hessian matrix validity such that the matrix is positive-definite, which verifies the optimization has a unique solution. The second analysis was to extract the A matrix from the state-space realization of the unconstrained MPC controller and calculate the eigenvalues. Here, the absolute values of eigenvalues are less than 1 except for the ramp-like disturbance model, which means that the feedback system is internally stable only for the first two output disturbance models. Next, closed-loop nominal stability was tested, extracting the A matrix from the feedback configuration. The absolute values of the eigenvalues were less than 1, which proves the stability of the unconstrained system. Finally, the closed-loop steady-state gains test verified that the controller enforces the density variable to the reference value at a steady-state without constraints.

In conclusion, according to the stability analysis, the control system is not stable when rejecting ramp-like disturbances at the output. Thus, we restricted ourselves to evacuation scenarios where the output disturbances were modeled with white noise or step-like signals. A step-like signal can be interpreted in our evacuation scenario as a situation where the density increases rapidly and remains constant for a period of time. In contrast, a ramp-like signal would be the case where the density increases indefinitely.

5. Experimental results

This section presents the experimental results for the two MPC control loop simulation scenarios (see Fig. 16):

1. Plant modeled with the identified ARX model (Simulation environment, SimulE).
2. Plant modeled with the pedestrian evacuation simulator (Deployment environment, DeployE).

5.1. Plant modeled with the identified ARX model (SimulE)

Here, the identified ARX model is used within the MPC controller to make predictions and as Plant Model. This configuration allows creating, configuring, and refining the MPC controller before deployment in a real environment. In our research, the "real" environment would be the MPC control loop in which the crowd dynamics (Plant) are modeled with the pedestrian evacuation simulator (DeployE).

We conducted experiments simulating evacuation processes of 1000 s and a reference density of 1 ped/m^2 at the sensing area, which is below the capacity level. We evaluated the system's response for the two output disturbance models for which the MPC controller was stable: A static gain for rejecting white noise disturbances and a static gain in series with an integrator for rejecting step-like disturbances.

Fig. 18 presents a single-shot experiment that shows the control actions b generated by the MPC controller, applied over the Plant model, and the corresponding pedestrian density response. In the absence of output perturbations, the static gain model provides better reference tracking than the integrator model, with significantly less variation in control actions.

For statistical significance, we performed 50 simulations for each output disturbance model and measured the Integral Square Error to evaluate their performance in terms of reference tracking and control variations (see Fig. 19). It can be observed significantly better performance in reference tracking and control variations using the static gain model. The span of b with the static gain model

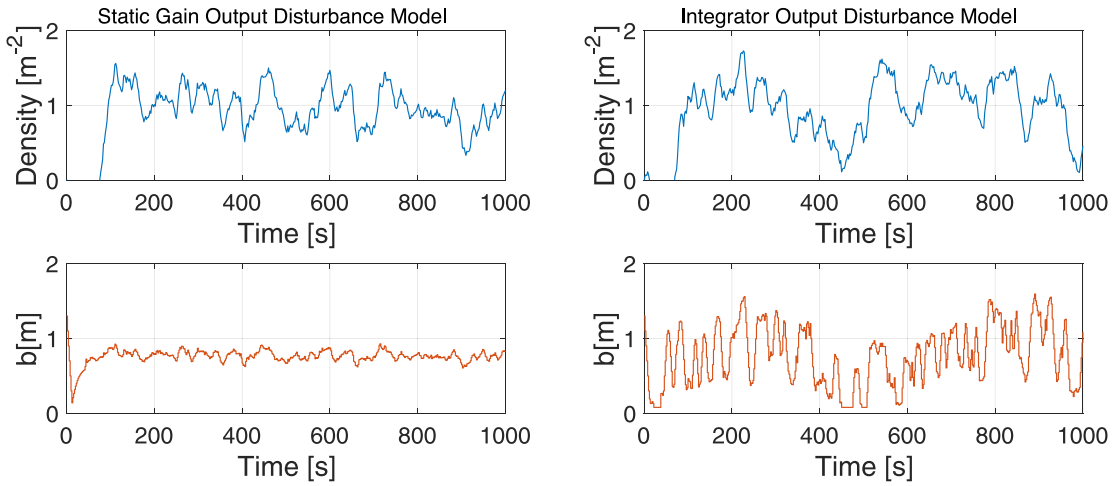

Fig. 18. Control actions b generated by the MPC controller against the plant model input and pedestrian density response at plant model output for a single-shot simulation experiment. The first column plots show the signals when the predictor's output disturbance model used is a static gain, while in the second column, the model used is an integrator.

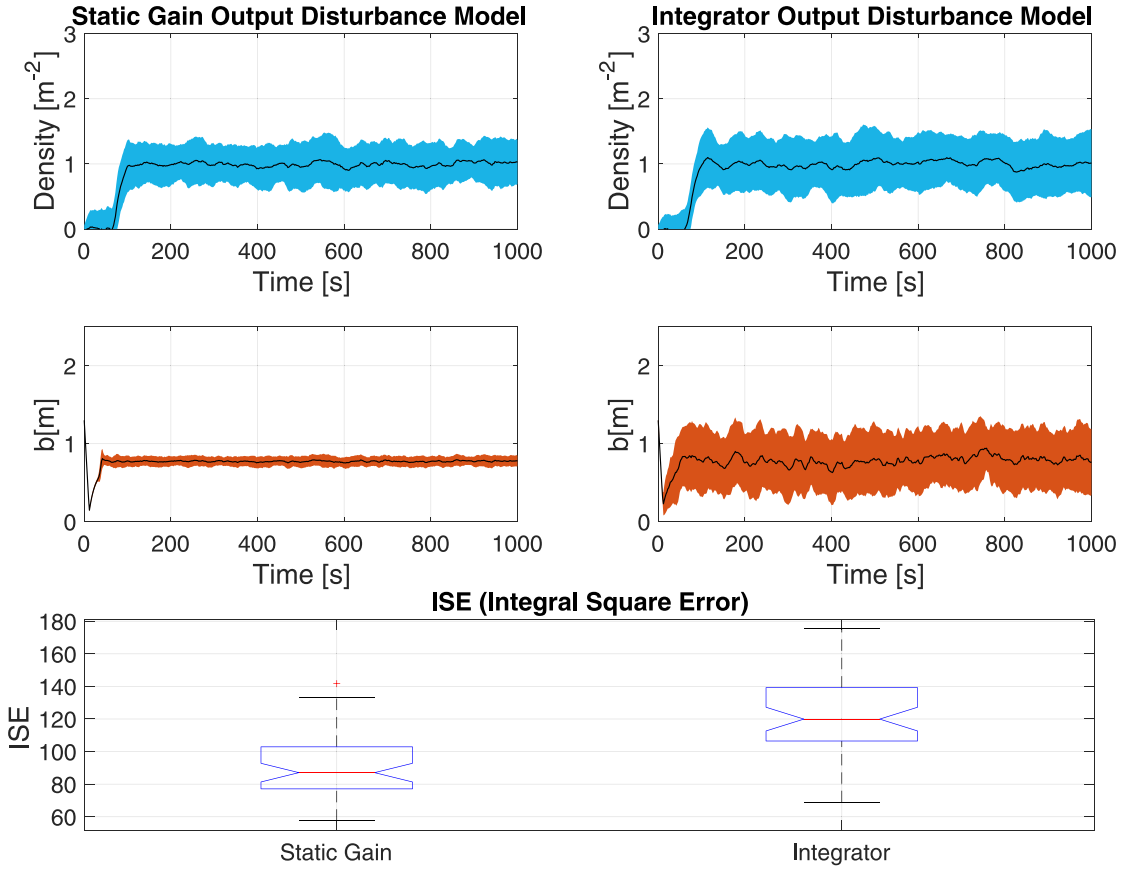

Fig. 19. Control actions b generated by the MPC controller against the plant model input and pedestrian density response at plant model output for 50 simulations. The plots represent the average values and confidence regions of 1 standard deviation. The first column plots show the signals when the predictor's output disturbance model is a static gain, while in the second column, the model used is an integrator. The last row presents the median boxplot of the Integral Square Error for the two configurations.

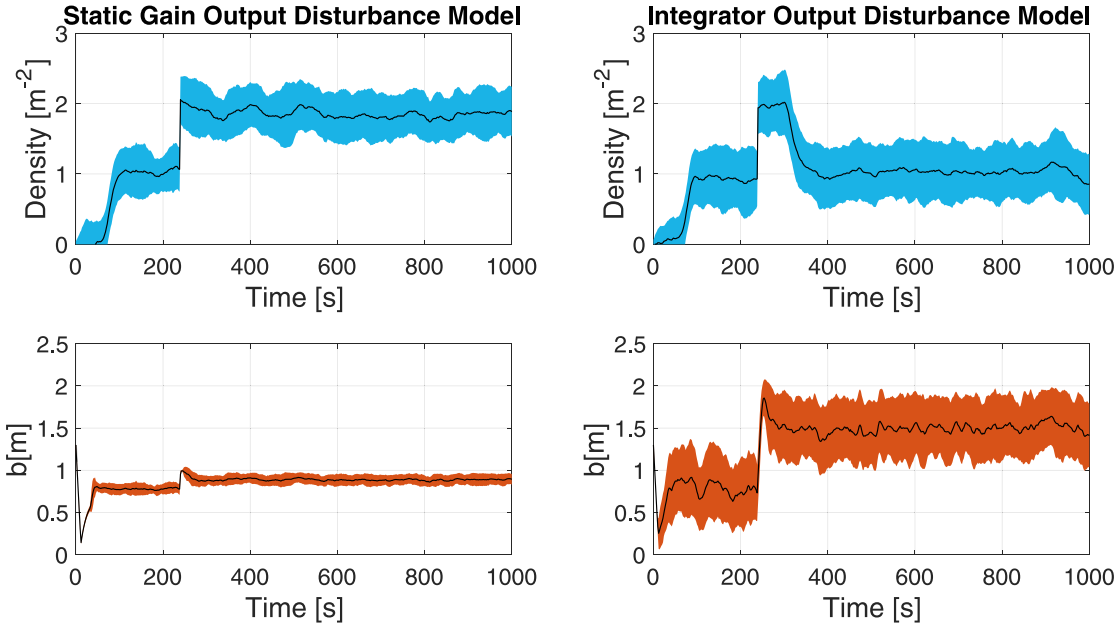

Fig. 20. Rejection of the MPC controller to step-like density disturbances at the output. Perturbation is generated as a density constant of amplitude 1, which is added to the density signal at time 240 s. The plots show the control actions b generated by the MPC controller against the plant model input and pedestrian density response at plant model output for 50 simulations. The plots represent the average values and confidence regions of 1 standard deviation. The first column plots show the signals when the predictor's output disturbance model is a static gain, while in the second column, the model used is an integrator.

ranges from 0.5 to 1 m, while with the integrator model, the interval is $0.08 - 2$ m. Interestingly, with soft control actions, the pedestrian density control at the output is more stable, and reference tracking improves. However, it comes at a cost if the output is perturbed with step-like signals.

Next, we evaluated the rejection of the MPC controller to step-like disturbances at the output. Thus, at time 240 s, we added to the density output a constant value of 1 ped/m^2 to simulate a step-like perturbation. It can be observed (Fig. 20) how the static gain model cannot reject the disturbance while the integrator model stabilizes the output in 100 s. The MPC controller dynamically adapts the control actions to the output disturbance by increasing the distance value b, which repeatedly reaches the limit of 2 m imposed by the soft constraint defined.

From the results, we suggest that the adequacy of using the static gain or integrator models will depend mainly on the expected behavior of the evacuation environment. The integrator model would be more suitable in conservative environments to reject external disturbances that extend over time. However, the static gain model would be preferred in egress situations where the operational performance is paramount or external disturbances are not expected or expected to be very short in time.

5.2. Plant modeled with the pedestrian evacuation simulator (DeployE)

This experimental scenario represents the "real" environment in which the crowd dynamics (Plant) are modeled with the pedestrian evacuation simulator. The simulator periodically receives the control actions from the MPC controller, applied to the pedestrians in the control area. Also, the simulator sends to the MPC controller the density measurements every $2s$.

As in the SimE experiments, we conducted experiments simulating evacuation processes with a reference density of 1 ped/m^2, for the two output disturbance models for which the MPC controller was stable. Fig. 21 presents a single-shot experiment that shows the control actions b generated by the MPC controller, the corresponding pedestrian density response, and the pedestrian macroscopic fundamental diagrams. In the absence of output perturbations, the static gain model provides a much better reference tracking than the integrator model, with significantly less variation in control actions. For the integrator output disturbance model, we observed a reference tracking with large oscillations up to 2 peds/m and a very different response from that observed in the SimE scenario, which suggests a mismatch between the identified ARX model and the pedestrian simulator. It can be observed how the fundamental diagram for the static gain model is always in a free-flow regime, while for the integrator model, there are flow drops due to the reach of the capacity zone.

To adjust the integrator model's sensitivity and achieve a less aggressive response, we adjusted the gain in series manually to 0.02. Thus, we obtained a density response with fewer oscillations and variations in control actions (Fig. 22). Also, the fundamental diagram exhibits a linear shape.

For statistical significance, we performed 50 simulations for each output disturbance model (see Fig. 23). The results were similar when compared to the SimulE scenario, where it can be observed significantly better performance in reference tracking and control

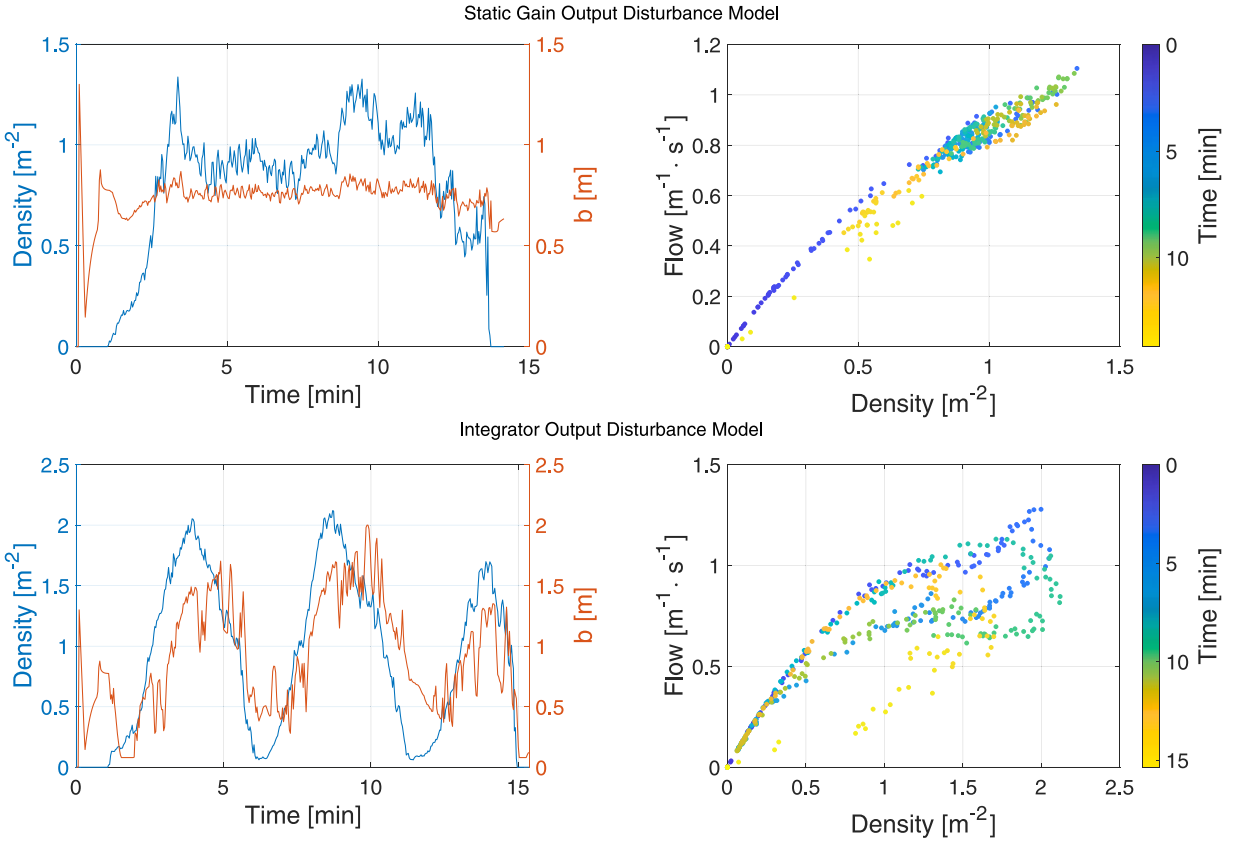

Fig. 21. Density output, b control actions, and fundamental diagrams in the sensing area for evacuations with the MPC controller configured with the static gain and integrator output disturbance models. Gains in both models are set to 1. The two-Y-axes plots represent the density and b control actions. The fundamental diagrams include color codes showing evacuation time.

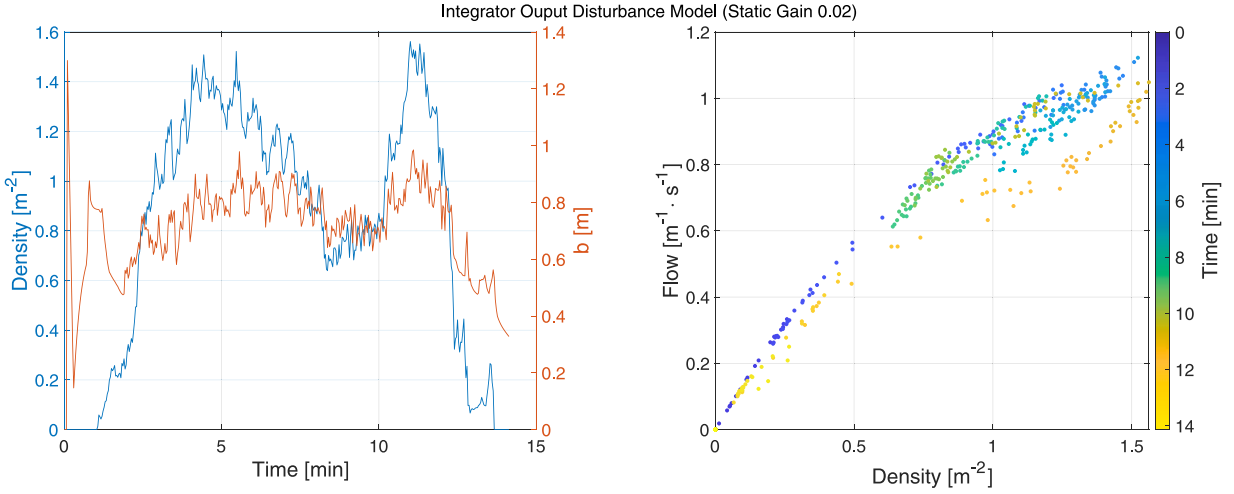

Fig. 22. Density output, b control actions, and fundamental diagram in the sensing area for evacuation with the MPC controller configured with the integrator output disturbance model. Gain of the integrator is set to 0.02. The two-Y-axes plots represent the density and b control actions. The fundamental diagrams include color codes showing evacuation time.

variations using the static gain model. The span of b with the static gain model ranges from 0.7 to 1 m, while with the integrator model, the interval is $0.2-1$ m. As expected, with soft control actions, the pedestrian density control at the output is more stable, and reference tracking improves. However, the more aggressive behavior of the controller with the integrator model resulted in shorter

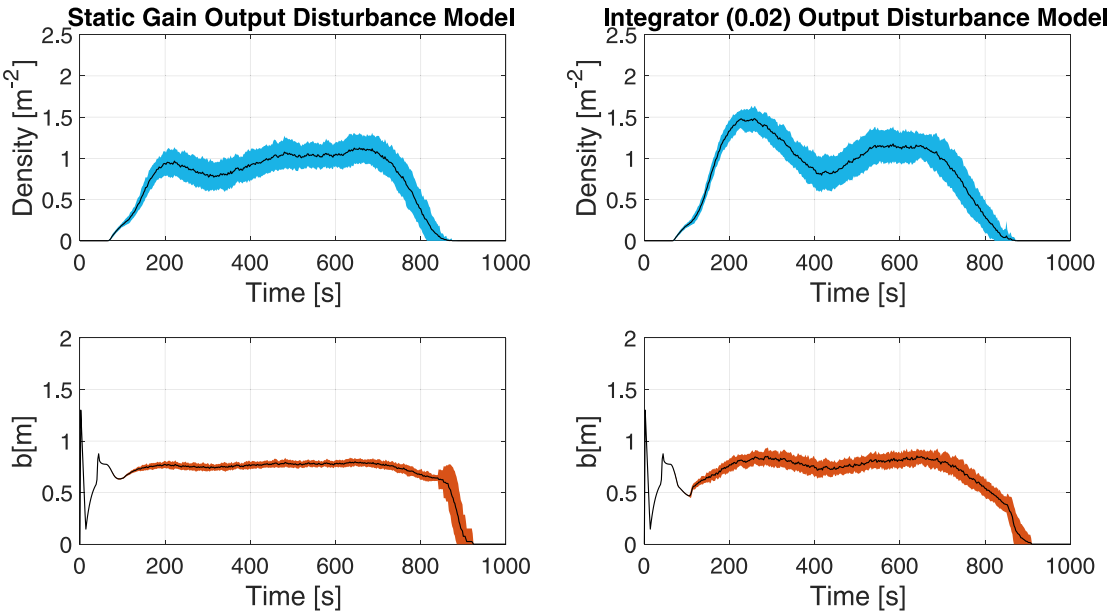

Fig. 23. Control actions b generated by the MPC controller against the pedestrian simulator and pedestrian density response at the sensing area for 50 simulations. The first column plots show the signals when the predictor's output disturbance model is a static gain, while in the second column, the model used is an integrator with gain 0.02.

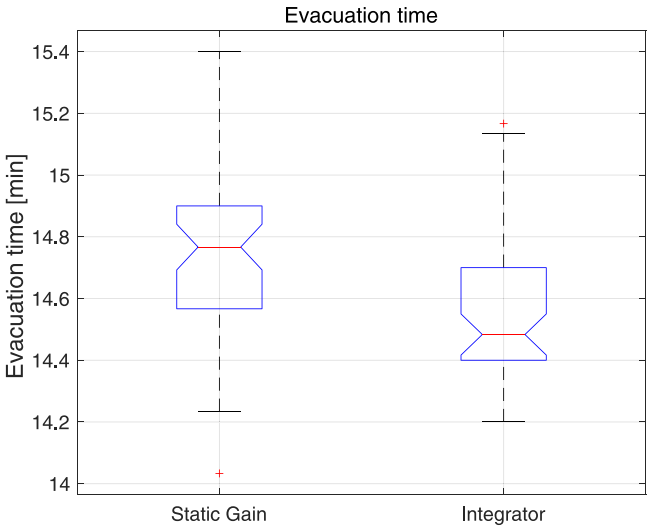

Fig. 24. Evacuation time boxplot for the static gain and integrator (gain 0.02) output disturbance models.

evacuation times (Fig. 24). For illustrative purposes, Fig. 25 shows snapshots of the evacuation process at intervals of 3 minutes. It can be observed how the MPC controller balances the formation of queues at the exits and congestion levels upstream.

Finally, the next step was to test the performance of both models to the rejection of step-like output disturbances. To simulate the step-like output disturbance, at time 240 s and for 10 minutes, we injected to the sensing area of the evacuation scenario a constant pedestrian flow of 100 peds/min. As in the SimulE scenario, we observed (Fig. 26) how the static gain model could not reject the disturbance. However, with the integrator model, the MPC controller dynamically adapted the control actions to the output disturbance by progressively increasing the distance value b. It can be seen how, with the static gain model, density grows to 2 peds/m^2, which only decreases at the end of the evacuation process. With the integrator model, the controller responds with a progressive increase in the b value once the output disturbance appears, which contributes to a decreasing density at the output. Interestingly, because of the static gain of 0.02 in series with the integrator, the span of the control actions is significantly lower than in the SimulE scenario (Fig. 20), which repeatedly reaches the limit of 2 m imposed by the soft constraint defined.

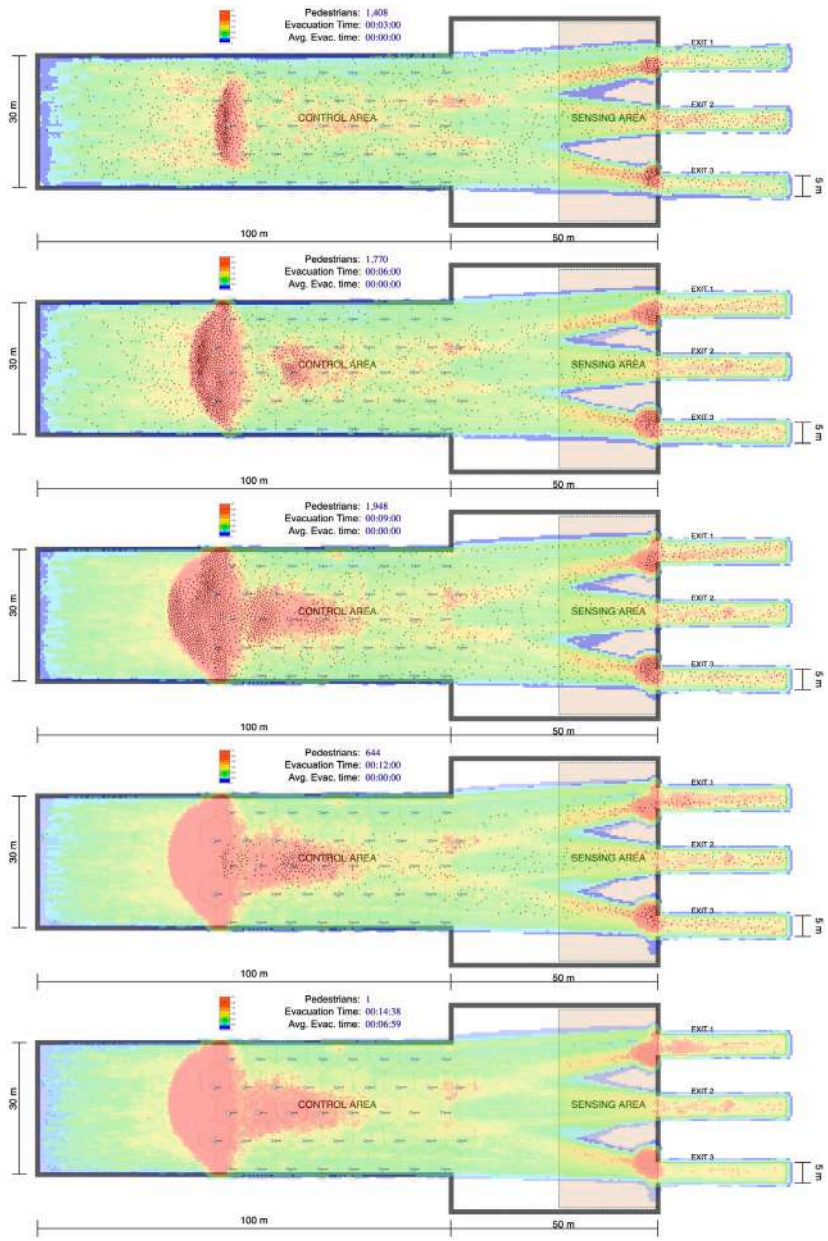

Fig. 25. Snapshots of an evacuation of 5000 pedestrians with the MPC controller set with the integrator output disturbance model and gain 0.02.

6. Discussion

The system identification methodology, applied to the mathematical modeling of pedestrian dynamic behavior, has proven its potential in designing pedestrian motion control mechanisms, exemplified by the construction of an MPC controller. In particular, we have modeled crowd dynamics as a single-input, single-output (SISO) system using a polynomial ARX model. However, it is essential to note that it would be possible to either use a state-space structure directly and do the parametric estimation or convert the polynomial model to a state-space model through a transformation. In both cases, our previous laboratory tests exhibit similar results.

In this proposal, the system identification mechanism lumps together all the loading, unloading, and intermediate phases of the evacuation process, without explicitly considering which population is pending evacuation and what state that population is, restricting the state modeling to the density measurement at the exit gates. However, one of the advantages of our proposed dynamic model structure lies in the possibility of its extension to a multiple-input multiple-output (MIMO) structure in a straightforward

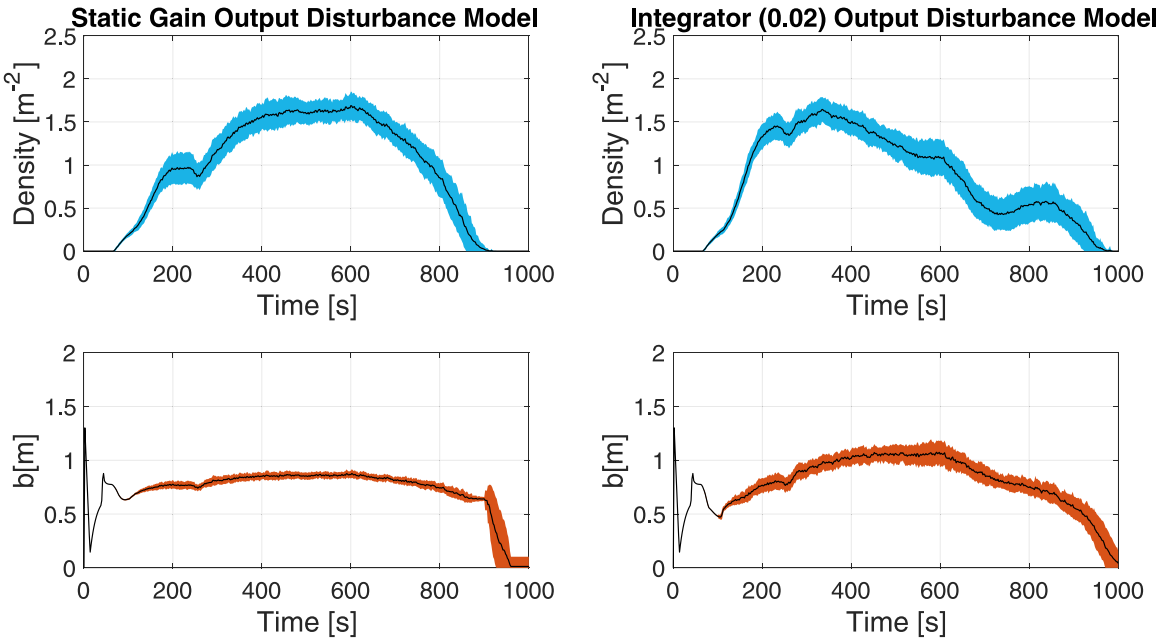

Fig. 26. Rejection performance of the deployed MPC controller to step-like density disturbances. The disturbance is generated as a constant pedestrian flow of 100 peds/min injected at the sensing area starting at 240 s. Plots show the control actions b generated by the MPC controller against the pedestrian simulator and pedestrian density response at the sensing area for 50 simulations.

manner. Incorporating the model of more input signals related to population metrics in the form of measured input disturbances can be very interesting in the specific case of MPC controller design. Similarly, the work has considered a control objective exclusively based on density tracking at the output. However, using both polynomial and state-space models, it is possible to naturally incorporate more output signals in the form of metrics related to pedestrian dynamics. These signals can be used directly in MPC controllers as measured or unmeasured outputs to apply multiple-objective tracking.

Another relevant aspect of the research is on using MPC control, which has proven to be suitable for behavioral control applications in evacuation scenarios. It has been shown how it is possible to define input constraints explicitly and simply so that the control actions in terms of their magnitude and variation are within reasonable margins. This aspect is critical in behavioral control, where, for example, rapid variations in instructions may not make sense in a real scenario. In turn, we have verified how, for a predefined performance metric (ISE) as a function of the fit to the density reference and distance control variation b, we have been able to adjust by sensitivity analysis the weighting factors of the optimization goal function of the MPC controller.

Interestingly, an aspect that we consider essential to our proposed MPC control is the possibility of adjusting the controller concerning the rejection capability to the expected output disturbances. This aspect is of particular interest in evacuation scenarios, where it is necessary to act against external disturbances like added flows, blockages, or capacity decrease. We have shown how adjusting the output disturbance model in the MPC controller will imply ranging between a static gain model or an integrator model with static gain in series. Our experiments suggest a broad-spectrum strategy to apply an integrator model and vary the gain manually until the desired performance with and without perturbations is achieved. However, this does not exclude the possibility of applying dynamic gain policies, which would be part of a more general adaptive control strategy.

The control variable b proposed presents two determining characteristics to consider when applied to real scenarios. First, the behavior of individuals when periodically receiving distance-keeping b values will have to be contextualized in terms of how they perceive the instructions and how they act accordingly. In this paper, we have assumed that the mapping between instructions or control actions, perception, and intention is direct, but in the future, how the processes of transferring control actions to an individual's response influence system behavior should be studied. Second, the domain of variable b is continuous, which means that although we can restrict the range of values, the degree of sensitivity of pedestrians to different perceived b values should be taken into account. In this sense, we believe that an exciting approach that should be studied is to discretize the domain of the control variable, which implies at least modifying the controller optimizer from a QP programming to a mixed-integer programming optimization problem.

We believe that our control proposal, based on distance-keeping, is reasonable in a real evacuation context because distance references may be easier to maintain than other types of speed-based variables. However, we are aware from a behavioral perspective that this requires experimental validation in real scenarios. In this sense, beforehand, it is interesting to evaluate whether speed-based control actions result in higher system efficiency than distance control.

Finally, and related to the previous point regarding the type of control variable, we believe that the typology of output variables can play a fundamental role in the efficiency of evacuation control systems, regardless of their single or multiple dimensions, as

mentioned above. For example, we have considered the density values calculated employing Voronoi diagrams, and when defining the reference tracking, we have pre-calculated the fundamental diagrams to set this reference at a value relatively far from its saturation point. We believe that this approach covers an enormous range of possible evacuation scenarios, constituting a solid metric of congestion state, widely used by the scientific community. However, it would be interesting to study how other congestion and performance metrics based, for example, on pedestrian counts, pedestrian flow, or densities calculated by other techniques, influence the control system performance.

7. Conclusions

An essential contribution of this study has been using the system identification methodology to model crowd dynamics in evacuation scenarios. The sequence of tasks needed to build a parametric model from input–output data has been formulated using a simulated evacuation scenario. Input–output datasets in the form of control actions based on distance-keeping and density measurement values have been generated strategically to improve the quality of the model. Not excluding other parametric model structures, the model obtained using the Prediction Error identification Method takes the form of a polynomial ARX model, which includes a manipulated variable (distance-keeping b), an unmeasured input disturbance, and measured output (pedestrian density).

It is demonstrated how a parametric model can be used in prediction tasks or simulations, replacing a real evacuation environment or a pedestrian motion simulator. Having a robust parametric model of the crowd dynamics has been shown critical in the design of behavioral controllers for improving crowd congestion. Compared to using a pedestrian simulator directly, the main advantage of using a parametric model in controller design is computation speed. Whether in the case of PID, reinforcement learning, or MPC controller tuning processes, it is necessary to iterate through multiple evacuation simulations. Using microscopic simulations with thousands of pedestrians can lead to unaffordable tuning or optimization times. In the specific case of MPC control, the prediction model has to operate fast enough for the optimizer to obtain the optimal control sequence within the control interval.

Another main contribution has been using MPC control technology for behavioral control, which has proven to be very efficient in reference tracking and minimization of control actions variation, demonstrating its inherent flexibility in the specification of constraints and capability of rejection to external disturbances. It has been investigated how varying the output disturbance model used in the prediction model of the MPC controller, the controller response to output disturbances typically present in crowd evacuations can be modulated.

Experimental results confirm that system identification methods and MPC control allow building pedestrian behavior controllers systematically and effectively to manage crowd congestion. The MPC controller can keep the density values within a reasonable range around the reference density while minimizing the distance-keeping control actions variations. Moreover, in the presence of external disturbances at the output and using a calibrated static gain in series with an integrator as an output disturbance model in the predictor, the controller performs reasonably well in their rejection.

The evidence from this study suggests the following:

- System identification techniques are ideally suited for constructing mathematical models of crowd dynamic systems from measured input–output data.
- Linear polynomial ARX models reasonably predict crowd congestion values from distance-keeping recommendations to pedestrians in a control area.
- Control actions based on the distance-keeping variable are an idoneous candidate to manage crowd congestion.
- From a methodological perspective, MPC control technology is ideally suited to design behavioral controllers in crowd evacuation scenarios.
- MPC control allows keeping the control actions within an acceptable range of values, which is essential when controlling pedestrian behavior, and also provides intuitive mechanisms to pursue different control objectives in terms of reference tracking and control variations.
- The response to expected output disturbances in crowd dynamics can be managed by strategically calibrating the output disturbance model used in the predictor module of the MPC controller.
- By varying the density reference value in the MPC controller, it is possible to balance evacuation time (efficiency) against the intrinsic risk of the crowd (safety).
- The proposed application of system identification and controller design techniques can be used as a general framework in which different evacuation scenarios and crowd management objectives could fit.

Several extensions to this research are being considered. We are investigating the discretization of the domain of the control variable, which implies at least modifying the controller optimizer from a QP programming to a mixed-integer programming optimization problem. We are also studying the influence of different forms of control and output variables, such as speed-based and different alternatives to density values for congestion measurement. Finally, we are exploring the extension of the studied SISO model to a MIMO configuration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Spanish Ministry of Economy and Competitiveness under Grant TIN2016-80622-P. The author is most grateful to the reviewers for their comments and recommendations on the text.

References

Abdelghany, A., Abdelghany, K., Mahmassani, H., Alhalabi, W., 2014. Modeling framework for optimal evacuation of large-scale crowded pedestrian facilities. European J. Oper. Res. 237, 1105–1118. http://dx.doi.org/10.1016/j.ejor.2014.02.054.

Bi, H., Gelenbe, E., 2019. A survey of algorithms and systems for evacuating people in confined spaces. Electronics 8, 711. http://dx.doi.org/10.3390/electronics8060711.

Brunetti, A., Buongiorno, D., Trotta, G.F., Bevilacqua, V., 2018. Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. Neurocomputing 300, 17–33. http://dx.doi.org/10.1016/j.neucom.2018.01.092.

Camacho, E.F., Alba, C.B., 2013. Model Predictive Control. Springer science & business media.

Chen, L., Tang, T.-Q., Huang, H.-J., Song, Z., 2018. Elementary students' evacuation route choice in a classroom: A questionnaire-based method. Physica A 492, 1066–1074. http://dx.doi.org/10.1016/j.physa.2017.11.036.

Cheng, Y., Zheng, X., 2018. Can cooperative behaviors promote evacuation efficiency? Physica A 492, 2069–2078. http://dx.doi.org/10.1016/j.physa.2017.11.124.

Dossetti, V., Bouzat, S., Kuperman, M.N., 2017. Behavioral effects in room evacuation models. Physica A 479, 193–202. http://dx.doi.org/10.1016/j.physa.2017.03.021.

Fang, Z., Li, Q., Li, Q., Han, L.D., Wang, D., 2011. A proposed pedestrian waiting-time model for improving space–time use efficiency in stadium evacuation scenarios. Build. Environ. 46, 1774–1784. http://dx.doi.org/10.1016/j.buildenv.2011.02.005.

Feliciani, C., Murakami, H., Shimura, K., Nishinari, K., 2020. Efficiently informing crowds – Experiments and simulations on route choice and decision making in pedestrian crowds with wheelchair users. Transp. Res. C 114, 484–503. http://dx.doi.org/10.1016/j.trc.2020.02.019.

Feliciani, C., Nishinari, K., 2016. An improved Cellular Automata model to simulate the behavior of high density crowd and validation by experimental data. Physica A 451, 135–148. http://dx.doi.org/10.1016/j.physa.2016.01.057.

Feliciani, C., Nishinari, K., 2018. Measurement of congestion and intrinsic risk in pedestrian crowds. Transp. Res. C 91, 124–155. http://dx.doi.org/10.1016/j.trc.2018.03.027.

Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. Transp. Res. B 42, 759–770.

Geroliminis, N., Sun, J., 2011. Properties of a well-defined macroscopic fundamental diagram for urban traffic. Transp. Res. B 45, 605–617.

Haghani, M., 2020a. Empirical methods in pedestrian, crowd and evacuation dynamics: Part II. Field methods and controversial topics. Saf. Sci. 129, 104760. http://dx.doi.org/10.1016/j.ssci.2020.104760.

Haghani, M., 2020b. Optimising crowd evacuations: Mathematical, architectural and behavioural approaches. Saf. Sci. 128, 104745. http://dx.doi.org/10.1016/j.ssci.2020.104745.

Haghani, M., Sarvi, M., 2017a. Social dynamics in emergency evacuations: Disentangling crowd's attraction and repulsion effects. Physica A 475, 24–34.

Haghani, M., Sarvi, M., 2017b. Stated and revealed exit choices of pedestrian crowd evacuees. Transp. Res. B 95, 238–259.

Haghani, M., Sarvi, M., 2019a. Heterogeneity of decision strategy in collective escape of human crowds: On identifying the optimum composition. Int. J. Disaster Risk Reduct. 35, 101064. http://dx.doi.org/10.1016/j.ijdrr.2019.101064.

Haghani, M., Sarvi, M., 2019b. Imitative (herd) behaviour in direction decision-making hinders efficiency of crowd evacuation processes. Saf. Sci. 114, 49–60. http://dx.doi.org/10.1016/j.ssci.2018.12.026.

Haghani, M., Sarvi, M., 2019c. 'Rationality' in collective escape behaviour: Identifying reference points of measurement at micro and macro levels. J. Adv. Transp. 2019, e2380348. http://dx.doi.org/10.1155/2019/2380348.

Haghani, M., Sarvi, M., 2019d. Simulating dynamics of adaptive exit-choice changing in crowd evacuations: Model implementation and behavioural interpretations. Transp. Res. C 103, 56–82. http://dx.doi.org/10.1016/j.trc.2019.04.009.

Haghani, M., Sarvi, M., Shahhoseini, Z., 2019. When 'push' does not come to 'shove': Revisiting 'faster is slower' in collective egress of human crowds. Transp. Res. A 122, 51–69. http://dx.doi.org/10.1016/j.tra.2019.02.007.

Helbing, D., Buzna, L., Johansson, A., Werner, T., 2005. Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. Transp. Sci. 39, 1–24.

Helbing, D., Farkas, I., Vicsek, T., 2000. Simulating dynamical features of escape panic. Nature 407, 487–490. http://dx.doi.org/10.1038/35035023.

Helbing, D., Johansson, A., Al-Abideen, H.Z., 2007. Dynamics of crowd disasters: An empirical study. Phys. Rev. E 75.

Helbing, D., Molnár, P., 1995. Social force model for pedestrian dynamics. Phys. Rev. E 51, 4282–4286. http://dx.doi.org/10.1103/PhysRevE.51.4282.

Helbing, D., Mukerji, P., 2012. Crowd disasters as systemic failures: Analysis of the love parade disaster. EPJ Data Sci. 1, 1–40.

Henderson, L.F., 1971. The statistics of crowd fluids. Nature 229, 381–383. http://dx.doi.org/10.1038/229381a0.

Hoogendoorn, S.P., Bovy, P.H.L., 2003. Simulation of pedestrian flows by optimal control and differential games. http://dx.doi.org/10.1002/oca.727.

Hoogendoorn, S.P., Daamen, W., Knoop, V.L., Steenbakkers, J., Sarvi, M., 2018. Macroscopic Fundamental Diagram for pedestrian networks: Theory and applications. Transp. Res. C 94, 172–184.

Jacques, J.C.S., Mussef, S.R., Jung, C.R., 2010. Crowd analysis using computer vision techniques. IEEE Signal Process. Mag. 27, 66–77.

Johansson, A., Helbing, D., Al-Abideen, H.Z., Al-Bosta, S., 2008. From crowd dynamics to crowd safety: A video-based analysis. Adv. Complex Syst. 11, 497–527.

Kaiser, M.S., Lwin, K.T., Mahmud, M., Hajializadeh, D., Chaipimonplin, T., Sarhan, A., Hossain, M.A., 2018. Advances in crowd analysis for urban applications through urban event detection. IEEE Trans. Intell. Transp. Syst. 19, 3092–3112. http://dx.doi.org/10.1109/TITS.2017.2771746.

Kinateder, M., Comunale, B., Warren, W.H., 2018. Exit choice in an emergency evacuation scenario is influenced by exit familiarity and neighbor behavior. Saf. Sci. 106, 170–175. http://dx.doi.org/10.1016/j.ssci.2018.03.015.

Kok, V.J., Lim, M.K., Chan, C.S., 2016. Crowd behavior analysis: A review where physics meets biology. Neurocomputing 177, 342–362. http://dx.doi.org/10.1016/j.neucom.2015.11.021.

Ljung, L., 1999. System Identification: Theory for the User, second ed. Prentice-Hall PTR, NJ.

Ljung, L., Chen, T., Mu, B., 2020. A shift in paradigm for system identification. Internat. J. Control 93, 173–180. http://dx.doi.org/10.1080/00207179.2019.1578407.

Lopez-Carmona, M.A., Paricio Garcia, A., 2021. CellEVAC: An adaptive guidance system for crowd evacuation through behavioral optimization. Saf. Sci. 139, 105215. http://dx.doi.org/10.1016/j.ssci.2021.105215.

Lopez-Carmona, M.A., Paricio Garcia, A., 2022. Adaptive cell-based evacuation systems for leader-follower crowd evacuation. Transp. Res. C 140, 103699. http://dx.doi.org/10.1016/j.trc.2022.103699.

Lovreglio, R., Ronchi, E., Kinsey, M.J., 2020. An online survey of pedestrian evacuation model usage and users. Fire Technol. 56, 1133–1153. http://dx.doi.org/10.1007/s10694-019-00923-8.

Ma, Y., Lee, E.W.M., Shi, M., 2017. Dual effects of guide-based guidance on pedestrian evacuation. Phys. Lett. A 381, 1837–1844. http://dx.doi.org/10.1016/j.physleta.2017.03.050.

Ma, Y., Yuen, R., Lee, E., 2016. Effective leadership for crowd evacuation. Physica A 450, 333–341. http://dx.doi.org/10.1016/j.physa.2015.12.103.

Murakami, H., Feliciani, C., Shimura, K., Nishinari, K., 2020. A system for efficient egress scheduling during mass events and small-scale experimental demonstration. R. Soc. Open Sci. 7, 201465. http://dx.doi.org/10.1098/rsos.201465.

Nguyen-Huu, K., Lee, K., Lee, S.-W., 2017. An indoor positioning system using pedestrian dead reckoning with WiFi and map-matching aided. In: 2017 International Conference on Indoor Positioning and Indoor Navigation. IPIN, pp. 1–8. http://dx.doi.org/10.1109/IPIN.2017.8115898.

Pan, X., Han, C.S., Dauber, K., Law, K.H., 2007. A multi-agent based framework for the simulation of human and social behaviors during emergency evacuations. AI Soc. 22, 113–132. http://dx.doi.org/10.1007/s00146-007-0126-1.

Pelechano, N., Malkawi, A., 2008. Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches. Autom. Constr. 17, 377–385. http://dx.doi.org/10.1016/j.autcon.2007.06.005.

Saberi, M., Mahmassani, H.S., 2014. Exploring areawide dynamics of pedestrian crowds: Three-dimensional approach. Transp. Res. Rec. 2421, 31–40. http://dx.doi.org/10.3141/2421-04.

Schmid, C., Biegler, L.T., 1994. Quadratic programming methods for reduced hessian SQP. Comput. Chem. Eng. 18, 817–832. http://dx.doi.org/10.1016/0098-1354(94)E0001-4.

Shi, M., Lee, E.W.M., Ma, Y., 2019a. A dynamic impatience-determined cellular automata model for evacuation dynamics. Simul. Model. Pract. Theory 94, 367–378. http://dx.doi.org/10.1016/j.simpat.2019.04.003.

Shi, X., Ye, Z., Shiwakoti, N., Tang, D., Lin, J., 2019b. Examining effect of architectural adjustment on pedestrian crowd flow at bottleneck. Physica A 522, 350–364. http://dx.doi.org/10.1016/j.physa.2019.01.086.

Song, X., Ma, L., Ma, Y., Yang, C., Ji, H., 2016. Selfishness- and selflessness-based models of pedestrian room evacuation. Physica A 447, 455–466. http://dx.doi.org/10.1016/j.physa.2015.12.041.

Song, X., Zhang, Z., Peng, G., Shi, G., 2017. Effect of authority figures for pedestrian evacuation at metro stations. Physica A 465, 599–612. http://dx.doi.org/10.1016/j.physa.2016.08.015.

Steffen, B., Seyfried, A., 2010. Methods for measuring pedestrian density, flow, speed and direction with minimal scatter. Physica A 389, 1902–1910.

Sticco, I.M., Cornes, F.E., Frank, G.A., Dorso, C.O., 2017. Beyond the faster-is-slower effect. Phys. Rev. E 96, 052303. http://dx.doi.org/10.1103/PhysRevE.96.052303.

Takimoto, K., Nagatani, T., 2003. Spatio-temporal distribution of escape time in evacuation process. Physica A 320, 611–621. http://dx.doi.org/10.1016/S0378-4371(02)01540-6.

Wang, H.-R., Chen, Q.-G., Yan, J.-B., Yuan, Z., Liang, D., 2015. Emergency guidance evacuation in fire scene based on pathfinder. In: Proceedings - 7th International Conference on Intelligent Computation Technology and Automation, ICICTA 2014. pp. 226–230. http://dx.doi.org/10.1109/ICICTA.2014.62.

Wang, J., Jin, B., Li, J., Chen, F., Wang, Z., Sun, J., 2019. Method for guiding crowd evacuation at exit: The buffer zone. Saf. Sci. 118, 88–95. http://dx.doi.org/10.1016/j.ssci.2019.05.014.

Yanagisawa, D., Tomoeda, A., Nishinari, K., 2012. Improvement of pedestrian flow by slow rhythm. Phys. Rev. E 85, 016111. http://dx.doi.org/10.1103/PhysRevE.85.016111.

Yang, X., Yang, X., Wang, Q., Kang, Y., Pan, F., 2020. Guide optimization in pedestrian emergency evacuation. Appl. Math. Comput. 365, 124711. http://dx.doi.org/10.1016/j.amc.2019.124711.

Zeng, G., Schadschneider, A., Zhang, J., Wei, S., Song, W., Ba, R., 2019. Experimental study on the effect of background music on pedestrian movement at high density. Phys. Lett. A 383, 1011–1018. http://dx.doi.org/10.1016/j.physleta.2018.12.019.

Zhang, J., Seyfried, A., 2013. Empirical characteristics of different types of pedestrian streams. Procedia Eng. 62, 655–662. http://dx.doi.org/10.1016/j.proeng.2013.08.111.

Zhang, X.L., Weng, W.G., Yuan, H.Y., 2012. Empirical study of crowd behavior during a real mass event. J. Stat. Mech. Theory Exp. 2012, P08012. http://dx.doi.org/10.1088/1742-5468/2012/08/P08012.

Zheng, X., Cheng, Y., 2011. Modeling cooperative and competitive behaviors in emergency evacuation: A game-theoretical approach. Comput. Math. Appl. 62, 4627–4634. http://dx.doi.org/10.1016/j.camwa.2011.10.048.

Zhou, J., Guo, Y., Dong, S., Zhang, M., Mao, T., 2019. Simulation of pedestrian evacuation route choice using social force model in large-scale public space: Comparison of five evacuation strategies. PLOS ONE 14, e0221872. http://dx.doi.org/10.1371/journal.pone.0221872.

Zou, B., Lu, C., Mao, S., Li, Y., 2020. Effect of pedestrian judgement on evacuation efficiency considering hesitation. Physica A 547, 122943. http://dx.doi.org/10.1016/j.physa.2019.122943.