WILEY | Hindawi

Research Article

Positivity-Preserving Discontinuous Galerkin Methods on Triangular Meshes for Macroscopic Pedestrian Flow Models

L. Yang [ID],[1,2] H. Liang [ID],[3] J. Du,[1,2] and S. C. Wong[4,5]

[1]*Yanqi Lake Beijing Institute of Mathematical Sciences and Applications, Beijing, China*
[2]*Yau Mathematical Sciences Center, Tsinghua University, Beijing, China*
[3]*College of Transportation Engineering, Tongji University, Shanghai, China*
[4]*Department of Civil Engineering, The University of Hong Kong, Hong Kong, China*
[5]*Guangdong-Hong Kong-Macau Joint Laboratory for Smart Cities, Hong Kong, China*

Correspondence should be addressed to H. Liang; lianghy@connect.hku.hk

The macroscopic models for solving the pedestrian flow problem can be generally classified into two categories as follows: first-order continuum models and high-order continuum models. In first-order continuum models, the density satisfies the mass conservation law, the speed is defined by a flow-density relationship, and the desired directional motion of pedestrians is determined by an Eikonal-type equation. In contrast, in high-order models, the velocity is governed by the momentum conservation law. In this study, we summarize existing first-order and high-order models and rewrite them in the form of unified scalar or system hyperbolic conservation laws. Next, we apply high-order discontinuous Galerkin methods with a positivity-preserving limiter on unstructured triangular meshes to solve the conservation law and a second-order fast-sweeping scheme to solve the Eikonal equations. Our method can efficiently model real-life complex computational regions and avoid nonphysical solutions and simulation blow-ups. Finally, numerical examples are presented to demonstrate the accuracy and effectiveness of the proposed solution algorithm. The numerical results validate the reliability of the proposed numerical method and highlight the advantages of triangular meshes.

1. Introduction

Due to the importance of the security and safety management of pedestrian traffic, the pedestrian and crowd dynamic problem has received considerable attention in recent decades. Models for simulating pedestrian flows can be classified into microscopic and macroscopic models. Commonly used microscopic models such as social force models, discrete element models, and cellular automaton-based models simulate pedestrian movement by quantitatively describing the physical and psychological interactions between individuals [1–3]. These models enable the simulation of pedestrian dynamics and the reproduction of empirical observations, such as the decision-making process [4], uni- and bidirectional fundamental diagrams [5], and pedestrian movements at bottlenecks [6] at low densities.

However, at high crowd density levels, microscopic models are not as computationally efficient as macroscopic models [7]. In addition, unlike microscopic models, dynamic macroscopic models can simulate the evolution of collective pedestrian behaviors, thus facilitating the analyses of key variables and parameters of crowd dynamics.

Macroscopic models can enhance the understanding of pedestrian flow dynamics by efficiently simulating large-scale and high-density crowds. The variables (e.g., flow intensity, density, and velocity) in a macroscopic model can be represented by smooth mathematical functions, and their relationship can be described by a set of partial differential equations. Among these models, the Lighthill–Whitham–Richards (LWR) model [8, 9] is one of the simplest continuum pedestrian models. Although this model was originally independently proposed by Lighthill and

Whitham [8] and Richards [9] (in 1950) to solve traffic models, it is widely used to study pedestrian dynamics due to the similarity between vehicular and pedestrian flows. In this model, a mass conservation law is used to describe the pedestrian flow, assuming an equilibrium state of the flow-density relationship, similar to that reported by Greenshields [10]. The traditional LWR model has several limitations, and thus, many extended LWR models have been designed to solve problems such as the multiclass problem [11]. Specifically, the traditional LWR model is defined in a one-dimensional (1D) space, in which the walk direction is unidirectional and known. The route-choice strategy can be identified using the flow-density relationship. In a two-dimensional (2D) pedestrian flow problem, although the speed is known, the walk direction is no longer a single entity, and many (even infinite) feasible walking directions may exist. Consequently, a mathematical formulation must be obtained to describe the route-choice strategy. To describe the route-choice strategy in a 2D pedestrian flow problem, Hughes [12] proposed a systematic modeling framework based on the continuum theory and used an Eikonal equation to implement the route-choice strategy. Subsequently, Hoogendoorn and Bovy [13, 14] developed a dynamic user-optimal pedestrian flow model to study user-optimal dynamic traffic assignment problems in continuous time and space, in which pedestrians are assumed to have perfect information regarding the modeled domain that can be used to choose the route that minimizes the actual walk cost. In this model, a time-dependent Hamilton–Jacobi equation is used to describe the route-choice strategy. Huang et al. [15] reformulated Hughes' model and discovered that the pedestrian route-choice strategy in Hughes' model satisfies the reactive dynamic user equilibrium (RDUE) principle, in which a pedestrian chooses a route to minimize their instantaneous travel cost to the destination. Xiong et al. [16] extended Huang's reactive DUE model to study bidirectional pedestrian flow problems. Later, Du et al. [17] proposed a predictive dynamic user equilibrium (PDUE) model, in which a pedestrian chooses a route to minimize their actual travel cost to the destination.

In the abovementioned 1D or 2D LWR models or extended LWR models, the speed is defined by a flow-density relationship. Such models are typically known as first-order or equilibrium models, and numerical simulations have demonstrated their usefulness in planning and designing walking facilities. Although equilibrium models can sensibly predict pedestrian dynamics, they cannot simulate the stop-and-go waves and forward propagation of disturbances in heavy traffic [18]. Thus, nonequilibrium models (second- or high-order models) have been designed. The Payne–Whitham (PW) model [19, 20] is the earliest known non-equilibrium model. The speed in the PW model satisfies the momentum equation rather than a flow-density relationship, as in the case of LWR models. The PW model has been used to study the wrong-way travel problem [21, 22], and many PW-like models have been developed, using different traffic sound speeds to solve different problems. In general, PW and PW-like models have the same structure, incorporating a mass conservation law and a momentum

equation. Such models are strictly hyperbolic when the characteristic speed exceeds the vehicle speed, which may lead to gas-like behavior [23]. Aw and Rascle [24] and Zhang [25] developed a nonequilibrium model devoid of this behavior. To describe the property of crowd pressure and identify the force chains, Liang et al. [26] proposed an extended PW (PWP) model that explicitly considers the crowd pressure and effects of pushing forces and high densities to describe pedestrian movement.

The abovementioned first- and higher-order models can be rewritten into a unified form that consists of a scalar or system conservation law coupled with an Eikonal equation (or Hamilton–Jacobi equation). In studies of numerical solution of these pedestrian flow models, finite difference (FD) [11, 14, 26–29] and finite volume (FV) methods [30] have been commonly used. These methods can be directly used to solve many pedestrian flow problems on structured grid points, rectangular meshes, or some simple unstructured meshes. However, FD methods cannot address problems involving real and complex computation regions, for which unstructured grids must be used. Moreover, FV methods are difficult to implement because of the challenges associated with reconstructions on complex meshes. Direct numerical simulations may lead to significant oscillations, nonphysical solutions, or even blow-ups when solving shock problems.

Discontinuous Galerkin (DG) methods are a class of finite element methods that have been widely used to solve hyperbolic equations. These methods use completely discontinuous basis functions and are thus more flexible than traditional continuous finite element methods. In addition, DG methods are characterized by a high-order accuracy, high parallel efficiency, flexibility for hp-adaptivity, and use of arbitrary geometries and meshes. The first DG method, introduced in 1973 by Reed and Hill [31], was used to solve a time-independent linear hyperbolic equation. Subsequently, Cockburn et al. [32–35] established a framework to solve nonlinear time-dependent problems by combining the explicit, nonlinearly stable high-order Runge–Kutta time discretization scheme [36] with the spatial DG discretization. In this method, exact or approximate Riemann solvers are used as interface fluxes, and total variation-bounded nonlinear limiters [37] are used to achieve nonoscillatory properties for strong shocks. There are some extended DG methods, such as local DG [38], direct DG [39], hybrid DG [40], and oscillation-free DG [41], for partial differential equations (PDEs) that contain higher-order spatial derivatives or to reduce the oscillation. This method is widely applied in the domains of aeroacoustics, gas dynamics, granular flows, magnetohydrodynamics, modeling of shallow water, oceanography, oil recovery simulations, turbulent flows, viscoelastic flows, and weather forecasting. In this study, we implement the high-order DG method over unstructured triangular meshes to solve the conservation law and establish a systematic numerical algorithm for modeling pedestrian dynamics.

Physically, the pedestrian density must be nonnegative in a continuum model. However, in numerical experiments, this property does not hold for many numerical schemes. The failure to preserve a positive density leads to ill-posedness of the models and can easily cause blow-ups,

especially when high-order numerical schemes are applied to low-density regions. In many applications, the negative values are simply truncated to zero or small positive values. Although this approach may be effective in certain problems, brute truncation will eventually result in blow-ups because the total mass increases every time a negative density is set as zero. To ensure stable computing, schemes that are demonstrably positivity-preserving while also preserving the conservation property must be designed. In reference [42], the authors first constructed an arbitrary high-order accurate maximum-principle-satisfying DG scheme for one-dimensional and multidimensional scalar conservation laws on rectangular meshes. Subsequently, the authors extended this method to a high-order accurate positivity-preserving scheme for Euler systems on rectangular and triangular meshes [43, 44].

To fill the research gap of robust numerical algorithms on triangular meshes for the reviewed pedestrian models, this study applies a genuinely high-order DG scheme with a positivity-preserving limiter on unstructured triangular meshes to solve the 2D conservation law, enabling the realization of robust simulations in complex computational domains. In addition, we use the second-order fast-sweeping method to solve the Eikonal equation that describes the route choice in first-order models or equilibrium speed in higher-order models. In numerical examples, we first provide a numerical example that has an exact solution to demonstrate the accuracy of the high-order scheme. Then, we test different cases of the PWP model. First, to validate the correctness of the schemes, we test an evacuation case involving a rectangular obstacle. The results are consistent with those obtained from the FD weighted essentially nonoscillatory method [26]. The simulations based on the two methods yield consistent numerical results, which confirm that the DG method can be applied to nonsmooth and unstructured meshes without the loss of accuracy or conservation properties. Second, an evacuation case with circular obstacles is designed to validate the higher-order model and highlight the advantages of the proposed approach. The averaged local flow-density relationship derived from the simulation produces a second-peak phenomenon, consistent with the findings of empirical studies [18, 45]. Moreover, the occurrence of stop-and-go waves is observed, which verifies the applicability of the proposed model to unstable crowd conditions. In addition to the density distribution, the aggregated pushing pressure calculated using the proposed model can serve as another risk-level indicator to strategically prevent crowd disasters. Thus, the high-order DG scheme with a positivity-preserving limiter is demonstrated to be accurate and effective.

The remainder of the paper is organized as follows. Section 2 reviews different continuum pedestrian dynamic models and describes the method of rewiring them in a unified form. Section 3 describes the numerical algorithm, including the DG scheme with a positivity-preserving limiter for hyperbolic equations, fast-sweeping method for Eikonal equations, and total variation diminishing (TVD) Runge–Kutta temporal discretization. Section 4 presents the numerical examples designed to test the properties and effectiveness of the proposed methods. Section 5 presents the concluding remarks.

2. Pedestrian Dynamic Model

When studying the pedestrian and crowd dynamics through the continuum modeling approach, the pedestrian dynamic model can be formulated as a set of partial differential equations (PDEs) or PDEs with relaxation. In this section, we review several commonly used continuum pedestrian models, including first-order and high-order models, and rewrite them into a unified form. The considered first-order models are the traditional LWR model and 2D extended LWR models (such as RDUE/PDUE). Among high-order models, different second-order models are introduced.

2.1. First-Order Models. We first introduce the first-order models, in which pedestrians always choose routes that minimize their cost. The LWR model is one of the simplest continuum pedestrian dynamic models, in which the pedestrian flow is considered to be in a 1D space. The traffic state variables such as the flow, speed, and density satisfy the following mass conservation law:

$$\rho_t + (\rho V_e(\rho))_x = q, \tag{1}$$

where ρ is the pedestrian density and q is the travel demand. $V_e(\rho)$ is the preferred speed, which is a given nonnegative and nonincreasing function with respect to ρ and defined in $[0, \rho_{jam}]$; ρ_{jam} is the maximum density, corresponding to a traffic jam. Thus, the flow flux f is defined as follows:

$$f(\rho) = \rho V_e(\rho). \tag{2}$$

In the LWR model, because $V_e(\rho)$ is a given function of ρ, the flow flux satisfies several fundamental diagrams, such as the Greenshields model.

The traditional LWR model is defined in a 1D space, in which the pedestrian walks forward to the destination and the speed (norm of the velocity vector) function $V_e(\rho)$ is given. Thus, the velocity is known. Although this model is useful for analytical problems, it cannot be applied to real-world problems because pedestrians typically walk in 2D scenarios. Therefore, we review the 2D-extended LWR models coupled with different route strategy assumptions. In 2D space, the following conservation law is satisfied:

$$\rho_t + (\rho u_e)_x + (\rho v_e)_y = q, \tag{3}$$

where $\mathbf{v}_e = (u_e, v_e)$ is the 2D-preferred velocity vector. Although the speed intensity $V_e = |\mathbf{v}_e|$ can be computed through a given speed-density relationship $V_e(\rho)$, as in the 1D case, pedestrians can move freely in any direction. Consequently, additional route-choice strategies are required to describe the travel direction of $\mathbf{v}_e$. Commonly used route strategy models include RDUE and PDUE models.

In RDUE models, pedestrians always choose routes that minimize their instantaneous walk cost and change their movement directions in a reactive manner. The route-choice strategy satisfies

$$\mathbf{v}_e(x, y, t) \,//\, -\nabla\phi_e(x, y, t), \tag{4}$$

where // indicates that two vectors are parallel. Here, ϕ_e is defined as the instantaneous cost potential, which can be computed using the following Eikonal equation:

$$\left|\nabla\phi_e(x, y, t)\right| = \pi(x, y, t), \tag{5}$$

where $\pi = \pi(\rho)$ is the local cost function.

In the PDUE model, pedestrians are assumed to have perfect information regarding the modeling domain that can be used to predict future traffic information. Thus, the users can choose routes that minimize their actual walk cost. The route-choice strategy satisfies the form presented in equation (4). However, the total travel cost ϕ_e must be redefined as the actual travel cost and computed using a time-dependent Hamilton–Jacobi equation as follows:

$$\frac{1}{V_e(x, y, t)}(\phi_e)_t(x, y, t) - \left|\nabla\phi_e(x, y, t)\right| = -\pi(x, y, t). \tag{6}$$

According to the route-choice strategy presented in equation (4), the density in both the RDUE and PDUE models is governed by the following 2D conservation law:

$$\rho_t(x, y, t) + \nabla \cdot \mathbf{f}(x, y, t) = q(x, y, t), \tag{7}$$

where $\mathbf{f}(x, y, t)$ is the flow flux, which is defined as follows:

$$\mathbf{f}(x, y, t) = \rho(x, y, t)\mathbf{v}_e(x, y, t)$$
$$= -\rho(x, y, t)V_e(x, y, t)\frac{\nabla\phi_e(x, y, t)}{\left|\nabla\phi_e(x, y, t)\right|}. \tag{8}$$

2.2. High-Order Models. In first-order models, the magnitude of the movement speed V_e is a given function of the density ρ (corresponding to the fundamental diagram), and the movement direction is computed using a suitable route-choice strategy. The existing fundamental diagrams are typically based on empirical observations and indicate the average behavior of moving pedestrians. However, evident individual differences and unstable movement patterns have been observed from empirical data. To describe these behavioral characteristics of pedestrian movement, the second-order PW model with relaxation has been designed. Similar to LWR models, the first equation in the PW model still expresses the mass conservation, and the density, flux, and travel demand satisfy the following conservation law:

$$\rho_t + (\rho u)_x + (\rho v)_y = q, \tag{9}$$

where u and v are the velocities in the x and y directions, respectively. In contrast to the LWR models, the velocities are no longer computed by the given speed functions and route-choice algorithms and satisfy the following momentum equations:

$$u_t + uu_x + vu_y + \frac{h_0^2}{\rho}\rho_x = \frac{u_e - v}{\tau},$$
$$v_t + uv_x + vv_y + \frac{h_0^2}{\rho}\rho_y = \frac{v_e - v}{\tau}, \tag{10}$$

where τ is the relaxation time; u_e and v_e are the equilibrium speed in the x and y directions, respectively; and $h_0 < 0$ is the traffic sound speed. Because h_0 is a negative parameter, the traditional PW model may produce negative speeds (i.e., wrong-way travel). To address this problem, many nonequilibrium PW models have been designed, in which the constant traffic sound speed h_0 is replaced by a function $h(\rho)$ that depends on the density ρ. The new momentum equations are defined as follows:

$$u_t + uu_x + vu_y + \frac{h^2(\rho)}{\rho}\rho_x = \frac{u_e - v}{\tau},$$
$$v_t + uv_x + vv_y + \frac{h^2(\rho)}{\rho}\rho_y = \frac{v_e - v}{\tau}. \tag{11}$$

The mathematical structure of these improved models is similar to that of the PW model, and thus, such models are named PW-like models.

To eliminate the gas-like behavior in PW-like models, Aw and Rascle [24] and Zhang [25] developed a non-equilibrium model shown to be devoid of the gas-like behavior. In this model, the momentum equation in PW-like models is replaced with the following equations:

$$(u + P_u)_t + u(u + P_u)_x + v(u + P_u)_y = \frac{u_e - v}{\tau},$$
$$(v + P_v)_t + u(v + P_v)_x + v(v + P_v)_y = \frac{v_e - v}{\tau}, \tag{12}$$

where P_u and P_v are the pressures in the x and y directions, respectively, which are increasing functions of the density.

The abovementioned higher-order models focus on the simulation of normal pedestrian movement in low-density scenarios. To address large-scale and dense problems such as crowd disasters, Liang et al. [26] proposed a higher-order continuum model that considers panic effects and explicitly takes into account the effect of the aggregated pushing force. The model formulation is

$$\begin{cases} \rho_t + (\rho u)_x + (\rho v)_y = 0, \\ \\ (\rho u)_t + (\rho u^2 + P_1)_x + (\rho uv)_y = \rho\frac{u_e - u}{\tau} - \frac{1}{m}\frac{\partial P_2}{\partial x}, \\ \\ (\rho v)_t + (\rho uv)_x + (\rho v^2 + P_1)_y = \rho\frac{v_e - v}{\tau} - \frac{1}{m}\frac{\partial P_2}{\partial y}, \end{cases} \tag{13}$$

with suitable initial boundary conditions. Here, $\overline{m}$ is the average mass of a pedestrian, and P_1 is the compression force, which is determined by the pedestrian density as follows:

$$P_1 = s(\rho), \tag{14}$$

where $s(\rho)$ is a given function that indicates the relationship between the average compression force and density. P_2 is the pushing force, which is generated by panic in a dense flow and satisfies the following equation:

$$\left\|\nabla\left(\frac{P_2}{\alpha}\right)\right\| = \frac{\delta(x, y, t)k(\rho)}{\alpha}; P_2 = 0, \quad \text{if } \alpha = 0, \tag{15}$$

where $k(\rho)$ is a given function that describes the relationship between the pushing capacity and density; $\delta(x, y, t) \in [0, 1]$ is a given parameter that describes the level of panic in the crowd; and α represents the relaxation factor defined as follows:

$$\alpha(x, y, t) = \begin{cases} 1, & \text{if } \nabla\rho \cdot \mathbf{v_e} \geq 0, \\ \max\left(\frac{\rho - \rho_0}{\rho_m - \rho_0}, 0\right), & \text{if } \nabla\rho \cdot \mathbf{v_e} < 0, \end{cases} \tag{16}$$

where ρ_0 represents the critical density when the pushing capacity is higher than 0, and ρ_m represents the maximum density.

2.3. Unified Form of Different Models. In the previous sections, we reviewed several classical pedestrian dynamic models including the first-order models (equilibrium models) and high-order models (nonequilibrium models). All these models can be rewritten in the following unified form. Note that we present the 2D cases of the models, except for the traditional LWR model as follows:

$$\mathbf{U}_t + \mathbf{f}(\mathbf{U})_x + \mathbf{g}(\mathbf{U})_y = \mathbf{Q}. \tag{17}$$

Table 1 presents the variables $\mathbf{U}, \mathbf{f}(\mathbf{U}), \mathbf{g}(\mathbf{U})$, and $\mathbf{Q}$ for the different models.

The equilibrium velocity vector (u_e, v_e) is computed using equations (4) and (5) (or equation (6)).

Remark. Many other continuum pedestrian models are available, but they have not been introduced in this section due to limited space. Most of these models can also be written as the conservation law, with different definitions of the conservation variables, pressure $h(\rho), P_u, P_v$, source terms, and equilibrium speeds.

3. Solution Algorithms

Equation (17) represents a 2D system of hyperbolic conservation laws, and we define $\mathbf{F}(\mathbf{U}) = (\mathbf{f}(\mathbf{U}), \mathbf{g}(\mathbf{U}))$. In this model, the source term $\mathbf{Q}$ is computed implicitly. At each fixed time, equation (5) must be solved to determine the equilibrium speed (u_e, v_e). Moreover, in the PWP model, equation (15) must be solved to obtain the gradient of P_2 and compute the source term $\mathbf{Q}$. Equations (5) and (15) can be written as the following Eikonal-type equation with suitable boundary conditions:

$$\|\nabla\phi\| = C(x, y). \tag{18}$$

Several numerical algorithms have been designed to solve the conservation law system. The DG method is one of the most popular algorithms and can be used to study problems with complex geometries by using unstructured meshes. In this study, we use the high-order DG method with a positivity-preserving limiter on triangular meshes to solve equation (17). At each time level of the DG method, we use the second-order fast-sweeping method on triangular meshes to solve the Eikonal equation, equation (18).

The formulation of the fast-sweeping method is presented in Section 3.1. The high-order DG method and positivity-preserving limiter are discussed in Sections 3.2 and 3.3, respectively.

3.1. Fast-Sweeping Method for Solving Eikonal Equations. We apply the fast-sweeping method to solve equation (18) on triangular meshes. Details of first- and second-order fast-sweeping methods on triangular meshes can be found in References [46, 47], respectively. In this study, the second-order fast-sweeping method is used.

We discretize the computational domain Ω into unstructured triangular cells Δ_j. As shown in Figure 1, for a given triangle Δ_j, the three angles are α, β, and γ and the lengths of the three edges are a, b, and c. The fast-sweeping method is an iterative method. Assuming that ϕ_{j1} and ϕ_{j2} at nodes $\mathbf{x}_{j1}$ and $\mathbf{x}_{j2}$ and their related derivatives $\nabla\phi_{jl} := (\partial_x\phi_{jl}, \partial_y\phi_{jl}), l = 1, 2$ are known, we aim to update ϕ_j at node $\mathbf{x}_j$.

Before introducing the local solver for updating ϕ_j, we examine the directional derivatives along edges a and b at node $\mathbf{x}_j$. The second-order approximations are defined as follows:

$$D_l\phi_j = 2\frac{\phi_j - \phi_{jl}}{\|\mathbf{x}_j - \mathbf{x}_{jl}\|} - \frac{\mathbf{x}_j - \mathbf{x}_{jl}}{\|\mathbf{x}_j - \mathbf{x}_{jl}\|} \cdot \nabla\phi_{jl}, \quad l = 1, 2. \tag{19}$$

The following definitions are considered:

TABLE 1: Pedestrian flow models.

Models	U	f(U)	g(U)	Q
LWR	ρ	ρV_e	—	q
RDUE/PDUE	ρ	ρu_e	ρv_e	q
Payne–Whitham	$\begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}$	$\begin{pmatrix} \rho u \\ \rho u^2 + h_0^2 \rho \\ \rho uv \end{pmatrix}$	$\begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + h_0^2 \rho \end{pmatrix}$	$\begin{pmatrix} q \\ \rho u_e - u/\tau \\ \rho v_e - v/\tau \end{pmatrix}$
PW-like	$\begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}$	$\begin{pmatrix} \rho u \\ \rho u^2 + h^2(\rho)\rho \\ \rho uv \end{pmatrix}$	$\begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + h^2(\rho)\rho \end{pmatrix}$	$\begin{pmatrix} q \\ \rho u_e - u/\tau \\ \rho v_e - v/\tau \end{pmatrix}$
Aw–Rascle–Zhang	$\begin{pmatrix} \rho \\ \rho(u + P_u) \\ \rho(v + P_v) \end{pmatrix}$	$\begin{pmatrix} \rho \\ \rho u(u + P_u) \\ \rho v(u + P_u) \end{pmatrix}$	$\begin{pmatrix} \rho \\ \rho u(v + P_v) \\ \rho v(v + P_v) \end{pmatrix}$	$\begin{pmatrix} q \\ \rho u_e - u/\tau \\ \rho v_e - v/\tau \end{pmatrix}$
PWP	$\begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}$	$\begin{pmatrix} \rho u \\ \rho u^2 + P_1^2 \\ \rho uv \end{pmatrix}$	$\begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + P_1^2 \end{pmatrix}$	$\begin{pmatrix} q \\ \rho u_e - u/\tau - 1/m\, \partial P_2/\partial x \\ \rho v_e - v/\tau - 1/m - \partial P_2/\partial x \end{pmatrix}$
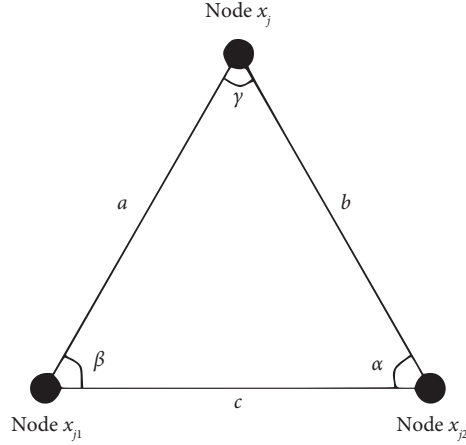
FIGURE 1: Triangle Δ_j.

$$\delta_1 = 2\phi_{j1} + \left(\mathbf{x}_j - \mathbf{x}_{j1}\right) \cdot \nabla\phi_{j1}, \delta_2 = 2\phi_{j2} + \left(\mathbf{x}_j - \mathbf{x}_{j2}\right) \cdot \nabla\phi_{j2} \text{ and}$$

$$p_1 = \frac{2\phi_j - \delta_1}{\left\|\mathbf{x}_j - \mathbf{x}_{j1}\right\|},$$

$$p_2 = \frac{2\phi_j - \delta_2}{\left\|\mathbf{x}_j - \mathbf{x}_{j2}\right\|}, \delta = \delta_2 - \delta_1. \tag{20}$$

According to the connection between the gradient $\nabla\phi_j$ and directional derivatives,

$$D\phi_j \approx \mathscr{H}\nabla\phi_j, \quad \mathscr{H} = \begin{bmatrix} \dfrac{\mathbf{x}_j - \mathbf{x}_{j1}}{\|\mathbf{x}_j - \mathbf{x}_{j1}\|} \\[2mm] \dfrac{\mathbf{x}_j - \mathbf{x}_{j2}}{\|\mathbf{x}_j - \mathbf{x}_{j2}\|} \end{bmatrix} \triangleq \begin{bmatrix} \mathscr{H}_1 \\[1mm] \mathscr{H}_2 \end{bmatrix}, \tag{21}$$

where $D\phi_j = [D_1\phi_j, D_2\phi_j]^T$, and superscript T represents transposition. According to equation (21), the gradient $\nabla\phi_j$ at node $\mathbf{x}_j$ can be approximated as follows:

$$\nabla\phi_j \approx \mathscr{H}^{-1}D\phi_j. \tag{22}$$

If this equation is combined with equation (18), the value of ϕ_j at node $\mathbf{x}_j$ satisfies

$$\left(D\phi_j\right)^T \left(\mathscr{H}\mathscr{H}^T\right)^{-1} D\phi_j = C\left(x_j, y_j, t\right), \tag{23}$$

where $C(x_j, y_j, t)$ is the source term at location (x_j, y_j) at time t.

As presented in Reference [46], the computed solution from equation (23) can be accepted only if the following upwind criteria are satisfied:

$$D_1\phi_j \geq \left(\mathscr{H}_1 \cdot \mathscr{H}_2^T\right)D_2\phi_j \text{ and } D_2\phi_j \geq \left(\mathscr{H}_1 \cdot \mathscr{H}_2^T\right)D_1\phi_j. \tag{24}$$

Otherwise, the gradient $\nabla\phi_j$ satisfies

$$\left|\nabla\phi_j\right| \approx \min\left(\frac{\phi_j - \phi_{j1}}{a}, \frac{\phi_j - \phi_{j2}}{b}\right). \tag{25}$$

Thus, by solving equation (23) and combining it with equation (25), the following expression can be obtained:

$$\left|\nabla\phi_j\right| \approx \begin{cases} \dfrac{1}{\sin\gamma}\sqrt{p_1^2 - 2p_1 p_2 \cos\gamma + p_2^2}, & \text{if equation (27) holds,} \\[4mm] \min\left(\dfrac{\phi_j - \phi_{j1}}{a}, \dfrac{\phi_j - \phi_{j2}}{b}\right), & \text{otherwise.} \end{cases} \tag{26}$$

This form pertains to the Godunov numerical Hamiltonian for Eikonal equations. According to equation (26),

$$\phi_j = \begin{cases} \dfrac{\delta_1 + \delta_2}{2} + \dfrac{\sin(\alpha - \beta)}{2 \sin \gamma}(\delta_2 - \delta_1) + \dfrac{\sin \alpha \sin \beta}{\sin \gamma}\sqrt{c^2 C_j^2 - \delta^2}, & \text{if equation (27) holds,} \\ \\ \min\left(\phi_{j1} + a C_j, \phi_{j2} + b C_j\right), & \text{otherwise,} \end{cases} \tag{27}$$

where $C_j = C(x_j, y_j, t)$.

Next, we introduce the sweeping process, which is a Gauss–Seidel iteration process. The sweeping order is a key parameter in the fast-sweeping method. The systematic order must efficiently cover all directions. The order of a rectangular mesh can be derived in a straightforward manner, and the following four sweeping directions are considered:

$$i = 1: I, j = 1: J \,(\text{up} - \text{right}); i = 1: I, j = J: 1\,(\text{up} - \text{left});$$
$$i = I: 1, j = 1: J \,(\text{down} - \text{left}); \text{and } i = I: I, j = J: 1\,(\text{down} - \text{right}), \tag{28}$$

where i and j are indices in the x and y directions, respectively. Notably, these natural orderings cannot be directly applied to unstructured meshes. A new systematic ordering must be identified to cover all directions. Following Reference [46], we choose at least three noncollinear reference points and then sort all nodes according to the l^p ($p = 1, 2$) distance to each reference point. In this manner, we can sweep all nodes based on their ascending and descending orders.

The process of the fast-sweeping method on a triangular mesh can be summarized as follows.

(1) Initialization

    (a) Choose the reference points $\mathbf{x}_{\text{ref}}^i, i = 1, \ldots, R$, and sort all nodes according to the l^p ($p = 1, 2$) distance to each reference point. Define S_i^+ and S_i^- as the arrays of all nodes according to the distance to the i-th reference point in decreasing and increasing orders, respectively.

    (b) Based on the boundary condition, assign the exact values ϕ_j and $D\phi_j$ at the inflow boundary. Large values (e.g., 10^{12}) are assigned as the initial estimates at all other grid points.

(2) Gauss–Seidel iteration

    (c) Denote the vector of the initial solutions as ϕ^k and set $k = 1$.

    (d) Sweeping

        (i) Choose the first sweeping direction and set $l = 1$.

        (ii) Loop through all nodes in set S_l^+ and apply the local solver to update the value.

        (iii) Loop through all nodes in set S_l^- and apply the local solver to update the value.

        (iv) If $l \geq R$, proceed to Step 2(e); otherwise, set $l = l + 1$ and return to Step ii.

    (e) If $\|\phi^{k+1} - \phi^k\| \leq \epsilon$, stop; otherwise, return to Step 2(d). Here, ϵ is a convergence threshold and $\|\cdot\|$ is the L_1 norm.

3.2. DG Methods. Equation (17) is solved using the DG method. The triangulation of the domain Ω is $\mathcal{F}$, and $\Delta_j \in \mathcal{F}$ is any triangular element. Equation (17) is multiplied by a test function v and then integrated over each cell Δ_j as follows:

$$\int_{\Delta_j} (\mathbf{U}(x, y, t))_t v(x, y) dx\, dy + \int_{\Delta_j} (\nabla \cdot \mathbf{F}(\mathbf{U}(x, y, t))) v(x, y) dx\, dy = \int_{\Delta_j} \mathbf{Q}(x, y, t) v(x, y) dx\, dy. \tag{29}$$

Integrating by parts yields

$$\frac{d}{dt}\int_{\Delta_j} \mathbf{U}(x, y, t) v(x, y) dx\, dy - \int_{\Delta_j} \mathbf{F}(\mathbf{U}(x, y, t)) \cdot \nabla v(x, y) dx\, dy + \int_{\partial \Delta_j} \mathbf{F}(\mathbf{U}(\mathbf{x}, \mathbf{y}, \mathbf{t})) \cdot \mathbf{n} v(x, y) ds$$

$$= \int_{\Delta_j} \mathbf{Q}(x, y, t) v(x, y) dx\, dy, \tag{30}$$

where $\mathbf{n}$ is the outward unit normal of the cell boundary $\partial\Delta_j$. The line integral in this equation is typically discretized by a Gaussian quadrature with a sufficiently high order of accuracy as follows:

$$\int_{\partial\Delta_j} \mathbf{F} \cdot \mathbf{n} v \, ds \approx |\partial\Delta_j| \sum_{k=1}^{M} w_k \mathbf{F}(\mathbf{U}(G_k, t)) \cdot \mathbf{n} v(G_k), \quad (31)$$

where G_k is the quadrature point and w_k is the quadrature weight.

Next, we approximate solution $\mathbf{U}$ by using a numerical solution within the finite element solution space V_h^k, defined as follows:

$$V_h^k = \left\{ v \in L^2(\Omega) : |v|_{\Delta_j} \in P^k(\Delta_j), \forall \Delta_j \in \mathscr{F} \right\}, \quad (32)$$

where $P^k(\Delta_j)$ represents the space consisting of polynomials of degree up to k in element Δ_j. In the finite element space V_h^k, the functions can be totally discontinuous across the element interfaces. We replace function U in equation (30) with $\mathbf{U}_h \in V_h^k$. Moreover, we select test functions from the

space V_h^k. Function v in equation (30) is replaced with $v_h \in V_h^k$. As $\mathbf{U}_h$ and v_h are allowed to have discontinuities across the cell interface, the flux term $\mathbf{F}(\mathbf{U}_h)$ and test function v_h on the cell boundary are not well defined. In the DG methods, $\mathbf{F}(\mathbf{U}_h) \cdot \mathbf{n}$ on the cell boundary must be replaced with a numerical flux. Although many types of numerical fluxes are available, the Lax–Friedrichs flux is used in this study as follows:

$$\widehat{\mathbf{F}}(\mathbf{U}_h) = \frac{1}{2} \left[(\mathbf{F}(\mathbf{U}_h^-) + \mathbf{F}(\mathbf{U}_h^+)) \cdot \mathbf{n} - a(\mathbf{U}_h^+ - \mathbf{U}_h^-) \right], \quad (33)$$

where $\mathbf{U}_h^-$ and $\mathbf{U}_h^+$ are the values of $\mathbf{U}_h$ on the cell boundary obtained from the inside and outside of triangle Δ_j, respectively, and $a = \max_{\mathbf{U}, \mathbf{n}} |\mathbf{F}'(\mathbf{U}) \cdot \mathbf{n}|$. The test function v_h on the cell boundary is set as v_h^-, derived from the inside of Δ_j. The DG method for solving equation (20) can be summarized as follows: find the unique solution $\mathbf{U}_h \in V_h^k$ such that $\forall v_h \in V_h^k$. For any triangular element Δ_j in the triangulation $\mathscr{F}$,

$$\frac{d}{dt}\int_{\Delta_j} \mathbf{U}_h(x, y, t) v_h(x, y) dx \, dy - \int_{\Delta_j} \mathbf{F}(\mathbf{U}_h(x, y, t)) \cdot \nabla v_h(x, y) dx \, dy + \int_{\partial\Delta_j} \widehat{\mathbf{F}}(\mathbf{U}_h(x, y, t)) v_h(x, y) ds$$

$$= \int_{\Delta_j} \mathbf{Q}(x, y, t) v_h(x, y) dx \, dy. \quad (34)$$

Similar to that in equation (31), this cell boundary integral can be approximated using the Gaussian quadrature rule. Similarly, the term $\int_{\Delta_j} \mathbf{F}(\mathbf{U}_h(x, y, t)) \cdot \nabla v_h(x, y) dx \, dy$ is computed by the following numerical quadrature:

$$\int_{\Delta_j} \mathbf{F}(\mathbf{U}_h(x, y, t)) \cdot \nabla v_h(x, y) dx \, dy \approx |\Delta_j| \sum_{k=1}^{N} \lambda_k \mathbf{F}(\mathbf{U}_h(H_k, t)) \cdot \nabla v_h(H_k), \quad (35)$$

where H_k is the quadrature point within Δ_j and λ_k is the quadrature weight.

The basis functions for the space $P^k(\Delta_j)$ are $\varphi_j^m(x, y)$ for $m = 1, \ldots, K$, where $K = (k+1)(k+2)/2$. Thus, the numerical solution can be represented as follows:

$$\mathbf{U}_h(x, y, t) = \sum_{m=1}^{K} \mathbf{U}_j^m(t) \varphi_j^m(x, y), \quad \forall (x, y) \in \Delta_j. \quad (36)$$

Substituting this form into equation (34) and taking $v_h = \varphi_j^l, l = 1, \ldots, K$ yield

$$A_j \frac{d}{dt} \mathbf{U}_j = \text{rhs}, \quad (37)$$

where $\mathbf{U}_j = (\mathbf{U}_j^1, \ldots, \mathbf{U}_j^K)^T$, and A_j is a $K \times K$ matrix with the entry $a_j^{l,m}$, defined as follows:

$$a_j^{l,m} = \int_{\Delta_j} \varphi_j^l(x, y) \varphi_j^m(x, y) dx \, dy, \quad (38)$$

and rhs is a $K \times 1$ vector with the l-th entry, defined as follows:

$$\text{rhs}^l = \int_{\Delta_j} \mathbf{F}(\mathbf{U}_h(x, y, t)) \cdot \nabla \varphi_j^l(x, y) dx \, dy - \int_{\partial\Delta_j} \widehat{\mathbf{F}}(\mathbf{U}_h(x, y, t)) \varphi_j^l(x, y) ds + \int_{\Delta_j} \mathbf{Q}(x, y, t) \varphi_j^l(x, y) dx \, dy. \quad (39)$$

Equation (37) can be rewritten as the following ordinary differential equation (ODE):

$$\frac{d}{dt}\mathbf{U}_j = \left(A_j\right)^{-1}\text{rhs} \triangleq L_j\left(\mathbf{U}_j\right). \quad (40)$$

To discretize the time to solve this ODE system, we use the third-order TVD Runge–Kutta method [36, 48], which can maintain the stability of the spatial discretization as follows:

$$\mathbf{U}_j^{(1)} = \mathbf{U}_j^n + \Delta t L_j\left(\mathbf{U}_j^n\right),$$

$$\mathbf{U}_j^{(2)} = \frac{3}{4}\mathbf{U}_j^n + \frac{1}{4}\left(\mathbf{U}_j^{(1)} + \Delta t L_j\left(\mathbf{U}_j^{(1)}\right)\right), \quad (41)$$

$$\mathbf{U}_j^{n+1} = \frac{1}{3}\mathbf{U}_j^n + \frac{2}{3}\left(\mathbf{U}_j^{(2)} + \Delta t L_j\left(\mathbf{U}_j^{(2)}\right)\right).$$

3.3. Positivity-Preserving Limiter.

This section describes the positivity-preserving schemes for triangular meshes developed in Reference [42]. As TVD time discretizations are convex combinations of the Euler forward operators, we need to consider only the first-order Euler forward time discretization. In this study, we use the global Lax–Friedrichs scheme, but the results can be easily extended to other monotonic schemes.

For the test function $v_h = 1$ in equation (34), we obtain a scheme that is satisfied by the cell averages of the DG method as follows:

$$\overline{\mathbf{U}}_j^{n+1} = \overline{\mathbf{U}}_j^n - \frac{\Delta t}{\left|\Delta_j\right|}\sum_{i=1}^{3}\int_{e_j^i}\widehat{\mathbf{F}}\left(\mathbf{U}_i^-, \mathbf{U}_i^+\right)ds + \frac{\Delta t}{\left|\Delta_j\right|}\int_{\Delta_j}\mathbf{Q}(x, y, t^n)dx\,dy, \quad (42)$$

where $\overline{\mathbf{U}}_j^n$ is the cell average of numerical solutions over triangle Δ_j at the time level n and $\mathbf{U}_i^-$ and $\mathbf{U}_i^+$ are the values of $\mathbf{U}_h$ on the i-th boundary e_j^i obtained from the inside and outside of the triangle Δ_j, respectively. For the first component of $\mathbf{U}_h$, i.e., the density ρ,

$$\overline{\rho}_j^{n+1} = \overline{\rho}_j^n - \frac{\Delta t}{\left|\Delta_j\right|}\sum_{i=1}^{3}\int_{e_j^i}\widehat{\mathbf{F}}_1\left(U_i^-, U_i^+\right)ds + \frac{\Delta t}{\left|\Delta_j\right|}\Delta t\int_{\Delta_j}q(x, y, t^n)dx\,dy, \quad (43)$$

where $\widehat{\mathbf{F}}_1$ and q are the first components of $\widehat{\mathbf{F}}$ and $\mathbf{Q}$, respectively. Let $\rho_j^n(x, y)$ be the k-th degree polynomial on triangle Δ_j. Then, the line integral must be approximated by

the least $k + 1$ Gauss quadrature point. This scheme can be rewritten as follows:

$$\overline{\rho}_j^{n+1} = \overline{\rho}_j^n - \frac{\Delta t}{\left|\Delta_j\right|}\sum_{i=1}^{3}\sum_{\beta=1}^{k+1}\widehat{\mathbf{F}}_1\left(\mathbf{U}_{i,\beta}^-, \mathbf{U}_{i,\beta}^+\right)w_\beta l_j^i + \frac{\Delta t}{\left|\Delta_j\right|}\int_{\Delta_j}q(x, y, t^n)dx\,dy, \quad (44)$$

where ω_β is the Gauss quadrature weight on the interval $[-1/2, 1/2]$ and $\sum_{\beta=1}^{k+1}\omega_\beta = 1$. $\mathbf{U}_{i,\beta}^-$ and $\mathbf{U}_{i,\beta}^+$ represent the values of $\mathbf{U}$ at the β-th Gauss quadrature point on the i-th edge, taken from inside and outside the triangle Δ_j, respectively; and l_j^i is the length of the i-th edge.

Next, we decompose the cell average $\overline{\rho}_j^n$ as a convex combination of the point values of the DG polynomial $\rho_j^n(x, y)$. Following Reference [44], we denote the set of quadrature points as follows:

$$S_k^j = \left\{ \left(\frac{1}{2} + v^\beta \right) \mathbf{V}_1 + \left(\frac{1}{2} + \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_2 + \left(\frac{1}{2} - \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_3, \right.$$

$$\left(\frac{1}{2} - \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_1 + \left(\frac{1}{2} + v^\beta \right) \mathbf{V}_2 + \left(\frac{1}{2} + \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_3, \text{ and}$$

$$\left(\frac{1}{2} + \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_1 + \left(\frac{1}{2} - \widehat{u}^\alpha \right) \left(\frac{1}{2} - v^\alpha \right) \mathbf{V}_2 + \left(\frac{1}{2} + v^\beta \right) \mathbf{V}_3,$$

$$\left. \alpha = 1, \ldots, N, \beta = 1, \ldots, k+1 \right\}, \tag{45}$$

where $\mathbf{V}_1, \mathbf{V}_2,$ and $\mathbf{V}_3$ are the position vectors of the three vertices of triangle Δ_j; and $\widehat{u}_\alpha, \alpha = 1, \ldots, N$ and $v^\beta, \beta = 1, \ldots, k+1$ are the Gauss–Lobatto and Gauss quadrature points on the interval $[-1/2, 1/2]$, respectively, where N is the smallest integer such that $2N - 3 \geq k$. If $k = 1$, the set of quadrature points S_1^j is as shown in Figure 2. The cell average of $\overline{\rho}_j^n$ can be decomposed as follows:

$$\overline{\rho}_j^n = \sum_{i=1}^{3} \sum_{\beta=1}^{k+1} \frac{2}{3} w_\beta \widehat{w}_1 \rho_{i,\beta}^{\text{int}(\Delta_j)} + \sum_{\gamma=1}^{L} \widetilde{w}_\gamma \rho_\gamma^{\text{int}}, \tag{46}$$

where $\rho_{i,\beta}^{\text{int}(\Delta_j)}$ is the value of the polynomial $\rho_j^n(x, y)$ at the quadrature points on edge e_j^i, ρ_γ^{int} is the value of the polynomial $\rho_j^n(x, y)$ at the quadrature points in the interior of Δ_j, and the number of interior points is $L = 3(N-2)(k+1)$. The coefficient is $\widetilde{w}_\gamma = 2/3 w_\beta \widehat{w}_\alpha, \beta = 1, \ldots, k \ldots, \alpha = 2, \ldots, N-1$. Similarly, the integral of the source term q can be decomposed as follows:

$$\int_{\Delta_j} q(x, y, t^n) dx \, dy = |\Delta_j| \left(\sum_{i=1}^{3} \sum_{\beta=1}^{k+1} \frac{2}{3} w_\beta \widehat{w}_1 q_{i,\beta}^{\text{int}(\Delta_j)} + \sum_{\gamma=1}^{L} \widetilde{w}_\gamma q_\gamma^{\text{int}} \right), \tag{47}$$

where $q_{i,\beta}^{\text{int}(\Delta_j)}$ is the value of $q(x, y, t)$ at the quadrature points on edge e_j^i and q_γ^{int} is the value of $q(x, y, t)$ at the quadrature points in the interior of Δ_j.

Next, we consider equation (42), which is satisfied by the cell averages of the DG method. If $\rho_j^n(x, y)$ is nonnegative for any $(x, y) \in S_k^j$ in the Courant–Friedrichs–Lewy condition with $a\Delta t / |\Delta_j| \sum_{i=1}^{3} l_j^i \leq 1/3 \widehat{w}_1$ and $\Delta t \leq \min_{j,m} A_q^{j,m}$, then $\overline{\rho}_j^{n+1}$ is also nonnegative. Here, l_j^i is the length of the i-th edge e_j^i of triangle Δ_j; $\widehat{w}_1$ is the quadrature weight of the N-point Gauss–Lobatto rule on $[-1/2, 1/2]$ for the first quadrature point; $A_q^{j,m}$ is the value of A_q at point $G_m^j \in S_k^j$; and A_q is a function of ρ and x and y, which is positive for positive density values. Its explicit form depends on the specific source term q [49].

At the time level n, the original DG polynomial does not satisfy the sufficient condition in this theorem in certain cases. Thus, we need to modify the polynomial $\rho_j^n(x, y)$ such that it is nonnegative for all $(x, y) \in S_k^j$. For any triangle, we assume that $\overline{\rho}_j^n \geq 0$ and define $\mathbf{U}_j^n(x, y)$ as the DG polynomial for $\mathbf{U}$ in triangle Δ_j. Next, we introduce the linear scaling limiter to derive the modified polynomial $\widetilde{\mathbf{U}}_j^n(x, y)$ as follows:

$$\widetilde{\mathbf{U}}_j^n(x, y) = \theta \left(\mathbf{U}_j^n(x, y) - \overline{\mathbf{U}}_j^n \right) + \overline{\mathbf{U}}_j^n, \theta = \min \left\{ 1, \frac{\overline{\rho}_j^n}{\overline{\rho}_j^n - m_j} \right\}, \tag{48}$$

with

$$m_j = \min_{(x,y) \in S_k^j} \left\{ \rho_j^n(x, y) \right\}. \tag{49}$$

According to equations (48) and (49), if the density ρ is always nonnegative, the parameter θ always takes a value of 1. In this case, $\widetilde{\mathbf{U}}_j^n(x, y) = \mathbf{U}_j^n(x, y)$ and the limiter does not work. However, it can be easily derived that $\widetilde{\mathbf{U}}_j^n(x, y) \geq 0$ for all $(x, y) \in S_k^j$. Therefore, we use the modified polynomial $\widetilde{\mathbf{U}}_j(x, y)$ to replace $\mathbf{U}_j^n(x, y)$ in equation (42). According to Theorem 1, the cell average $\overline{\rho}_j^{n+1}$ at the time level $n + 1$ is also nonnegative.

4. Numerical Examples

In this section, we first provide a numerical test with a simplified setup to the model which has an exact solution, to demonstrate the accuracy of the high-order positivity-preserving DG scheme. Then, the PWP model is considered as an example to evaluate the proposed numerical algorithms.

4.1. Example 1. We solve the following model equation:

$$\begin{cases} \rho_t + (\rho u)_x + (\rho v)_y = 0, \\ (\rho u)_t + (\rho u^2 + P)_x + (\rho u v)_y = s_1, \\ (\rho v)_t + (\rho u v)_x + (\rho v^2 + P)_y = s_2, \\ \rho(x, 0) = \rho_0, u(x, 0) = u_0, v(x, 0) = v_0, \end{cases} \tag{50}$$
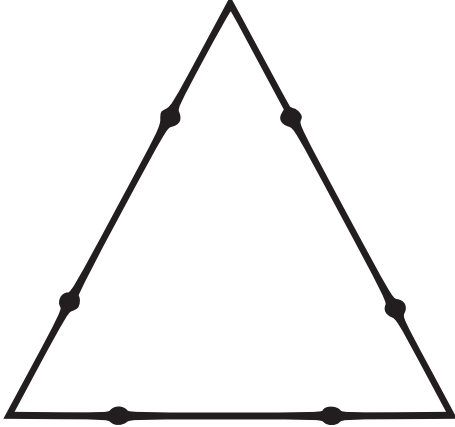
FIGURE 2: Set of quadrature points S_1^j.

where $P = \rho$ and s_1 and s_2 are the source terms, chosen so that an explicit exact solution is available. In this example, we solve the above system on a domain of $[0, 1] \times [0, 1]$, and the initial data take $\rho_0 = 1 + 0.2 \sin(2\pi(x + y))$, $u_0 = 1$, $v_0 = 1$. The exact solutions for this problem are $\rho = 1 + 0.2 \sin(2\pi(x + y - 2t))$, $u = 1$, $v = 1$ when the source terms are $s_1 = s_2 = 0.4\pi \cos(2\pi(x + y - 2t))$.

The errors and orders of accuracy are shown in Table 2. We can clearly see that the desired second-order of accuracy is achieved with the L^2 error of ρ on the triangular meshes.

4.2. Numerical Example of the PWP Model.

We consider two panic evacuation scenarios on a $100 \, \text{m} \times 50 \, \text{m}$ rectangular railway platform with obstacles of different shapes. Specifically, the platform is considered to have rectangular and circular obstacles in Examples 2.1 and 2.2, respectively.

In both examples, the railway platform is initially empty, with $\rho(x, y, 0) = 0$ at $t = 0$ s. Over time, pedestrians enter the platform from the left boundary and move from the left to the right within the platform. The boundary conditions for both examples are set as follows. The origin Γ_o is at $x = 0$ m, $y \in [0, 50 \, \text{m}]$. The inflow at the origin is

$$\mathbf{F}_{\text{in}} = \begin{bmatrix} \rho_{\text{in}} f_d (\rho_{\text{in}}) \\ \rho_{\text{in}} f_d (\rho_{\text{in}})^2 + s (\rho_{\text{in}}) \\ 0 \end{bmatrix}, \tag{51}$$

where

$$\rho_{\text{in}}(t) = \begin{cases} 2.5 \times \dfrac{t}{60}, & t \in [0, 60 \, \text{s}], \\[2ex] 2.5, & t \in [60, 120 \, \text{s}], \\[2ex] 2.5 \times \left(3 - \dfrac{t}{60}\right), & t \in [120, 180 \, \text{s}], \\[2ex] 0, & t \in [180, 400 \, \text{s}], \end{cases} \tag{52}$$

and the unit of ρ_{in} is ped/m^2. The destination is $x = 100$ m, $y \in [0, 50 \, \text{m}]$, where the cost potential is zero. The upper ($y = 50$) and lower ($y = 0$) boundaries are solid wall boundaries, and the projection of the speed vector on the normal direction of a boundary is set as zero. Table 3 specifies the other parameters and functions.

4.2.1. Example 2.1. A rectangular obstacle is located at $[50, 70] \times [10, 30]$ (unit: m). The proposed methods are used to solve the PWP model on triangular meshes. Figure 3 illustrates a sample triangular mesh with $N = 9893$ cells. The characteristics of the numerical results obtained by the DG method are consistent with those of the findings of our previous study [26].

Figure 4 compares the density cut along the lines $y = 40$ m for the PWP model simulated by the P^1 DG scheme without and with the positivity-preserving limiter at $t = 0.15$ s (see subfigures 4(a) and 4(b)). The P^1 DG scheme without the positivity-preserving limiter produces negative solutions, whereas the positivity of the density is maintained when using the P^1 DG scheme with the positivity-preserving limiter. The solution will blow up if the PWP model is continually computed using the P^1 DG scheme without the positivity-preserving limiter.

Figure 5 shows the density curves in the 1D section corresponding to $x = 48$ m (sub-Figure 5(a)) and $y = 36$ m (subfigure 5(b)) at $t = 200$ s for different mesh sizes, as obtained from the DG method. The numbers of elements in the triangular meshes are $N = 1401, 5560$, and 9893. By gradually refining the mesh, the deviation between the density curves can be reduced. The convergence of both numerical solutions is thus validated.

Figure 6 plots the density distributions during the evacuation at different time points, determined using the second-order DG method on triangular meshes with $N = 9893$. Before $t = 150$ s, the pedestrian flow is characterized by calm behaviors; the pedestrians try to avoid the obstacle and move at the free-flow speed (see subfigure 6(a)). At $t = 150$ s, congestion is induced near the corner of the rectangle because the obstacle decreases the available space, leading to a maximum density of approximately 6.0 ped/m^2. The passage is further blocked by the congested pedestrian flow (see subfigure 6(b)). After $t = 150$ s, panic begins to propagate in the crowd (see subfigures 6(c)–6(f)). In this condition, the density increases more slowly than in the normal condition as it is already extremely high. However, pedestrians continue to enter the flow, generating a high pushing pressure in the dense crowd, as shown in Figure 7. The pushing pressure propagates from the low-density region to the high-density region (close to the obstacle) and reaches a value of approximately 250|N/m (see subfigures 7(a) and 7(b)). The pressure increases rapidly in the dense crowd, and the density level remains high. In other words, the crowd pressure characteristics, which have been observed in many crowd disasters [49–51], are correctly rendered.

The averaged local flow-density relationship obtained by the DG method on triangular meshes also reveals the

TABLE 2: Errors and orders of accuracy for the density ρ.

N	L^2 error of ρ	Order of accuracy
139	$5.877590E-003$	—
506	$1.021613E-003$	2.52
1932	$2.312943E-004$	2.14
7579	$5.612360E-005$	2.04
29956	$1.394325E-005$	2.01

TABLE 3: Parameters and functions.

Critical density	$\rho_0 = 5$ ped/m^2
Maximum density	$\rho_m = 7$ ped/m^2
Panic sentiment	$\delta = \begin{cases} 0, & t \in [0, 150\text{ s}] \\ \max\{0, (\rho - \rho_0)/(\rho_m - \rho_0)\}, & t \in [150, 400\text{ s}] \end{cases}$
Sonic speed	$c_0 = 0.5$ m/s
Compression force	$P_1 = s(\rho) = c_0 \rho^2$ N/m
Relaxation time	$\tau = 0.61$ s
Average mass	$\overline{m} = 60$ kg
Local cost	$\pi(\rho) = 0.01\rho^2 + 1/f_d(\rho)$
Fundamental diagram	$f_d(\rho) = 1.034 \exp(-0.06\rho^2)$
Pushing capacity	$k(\rho) = 600\sqrt{\max(0, \rho - \rho_0)}$ N/m
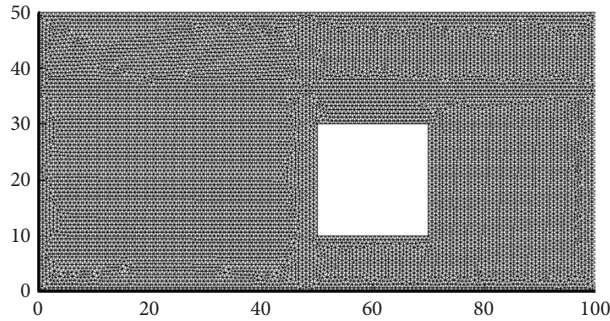
FIGURE 3: Sample triangular mesh ($N = 9893$) for example 2.1.

(a)

(b)
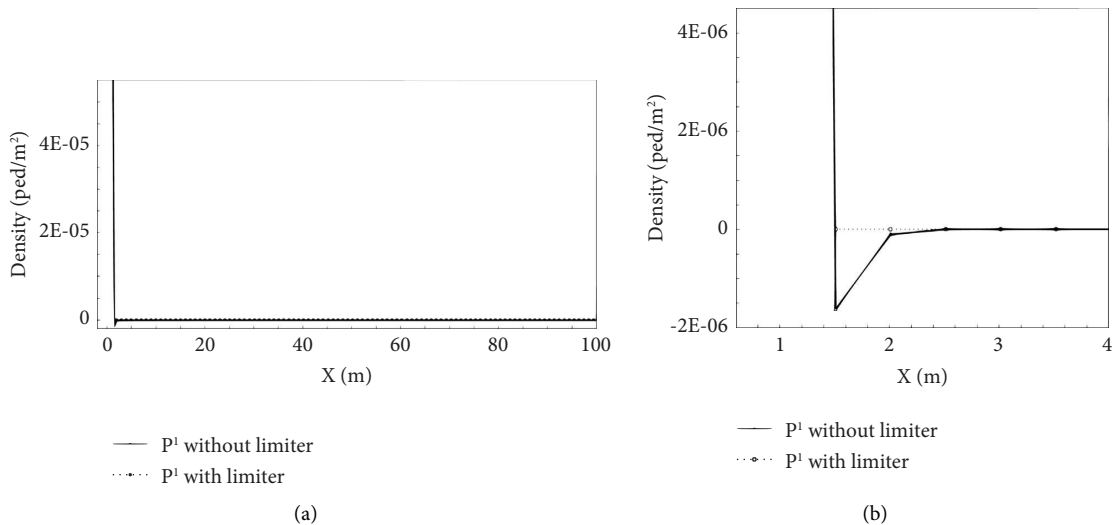
FIGURE 4: Comparison of the density cut along the lines $y = 40$ m for the PWP model simulated by the P^1 DG scheme without and with the positivity-preserving limiter at $t = 0.15$ s. The number of elements in the triangular mesh is $N = 9893$. (a) Complete region and (b) magnified view of the local density plot in (a).

(a)

(b)
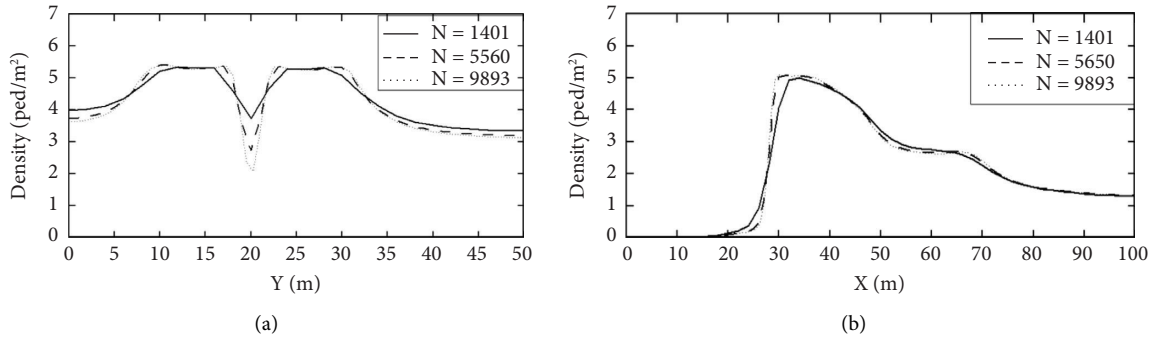
FIGURE 5: Density curves corresponding to (a) $x = 48$ m and (b) $y = 46$ m at $t = 200$ s in example 2.1, simulated using the DG method.
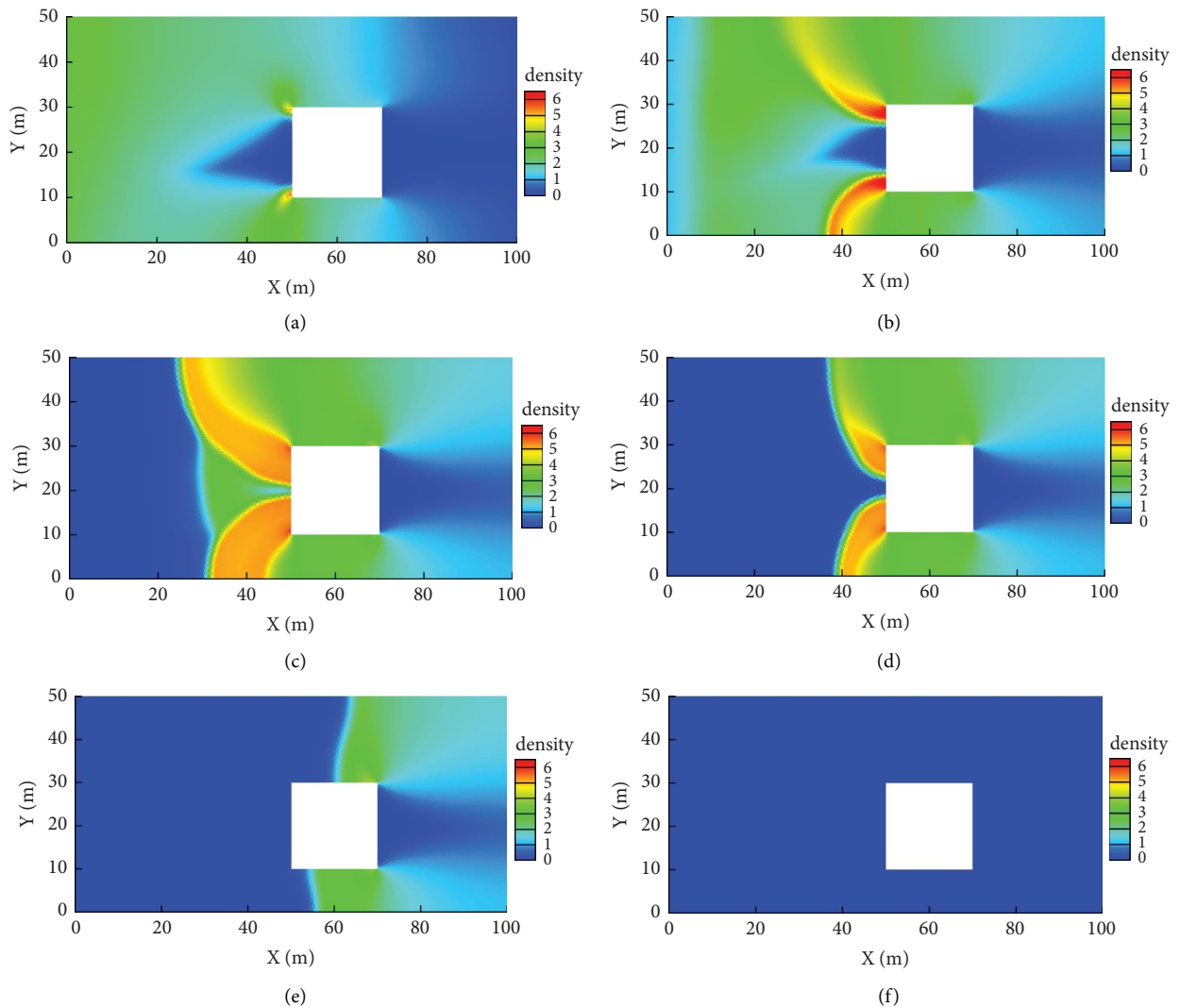

(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 6: Density distributions (unit: ped/m^2) simulated in example 2.1 using the second-order DG method ($c_0 = 0.5$ m/s). The number of elements in the triangular mesh is $N = 9893$. (a) $t = 100$ s. (b) $t = 150$ s. (c) $t = 200$ s. (d) $t = 250$ s. (e) $t = 300$ s. (f) $t = 400$ s.

second-peak phenomenon, consistent with the findings reported by Reference [26]. The averaged local flow and corresponding density are calculated in the simulation of a panic evacuation between $t \in [150, 400]$ by using the method presented in Reference [26]. In Figure 8, these entities correspond to the "actual flow" and are compared with the values derived from the fundamental diagram. A notable second peak occurs at approximately $\rho = 5.6$ ped/m^2 due to the panic sentiment and aggregated pushing pressure. Helbing et al. [18] first observed this phenomenon when they

(a)

(b)
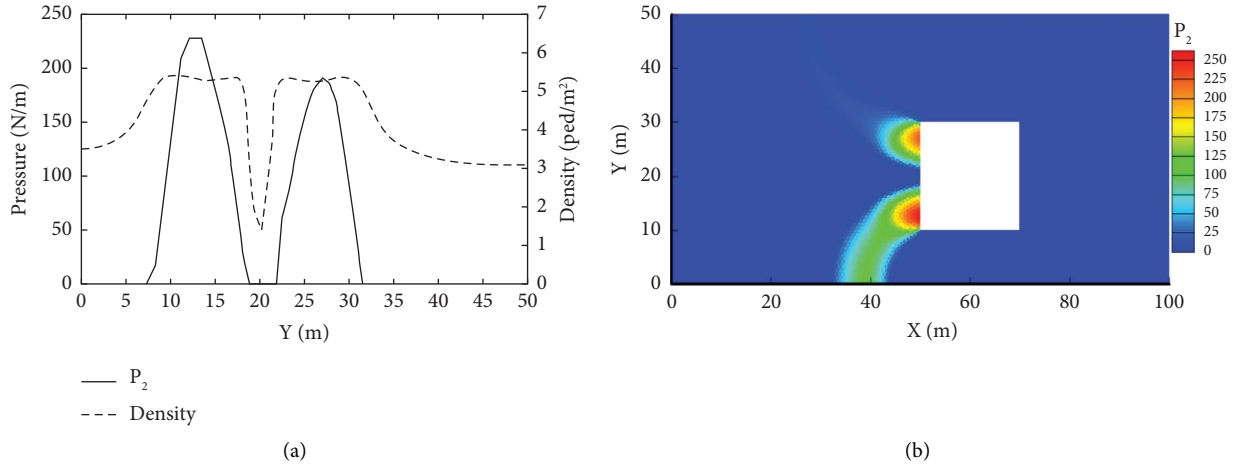
FIGURE 7: Pressure distribution simulated at $t = 200$ s in example 2.1 using the DG method ($c_0 = 0.5$ m/s). (a) Plot of density curve and pushing pressure curve corresponding to $x = 48$ m. (b) Contour plot of the pushing pressure distribution (unit: N/m).
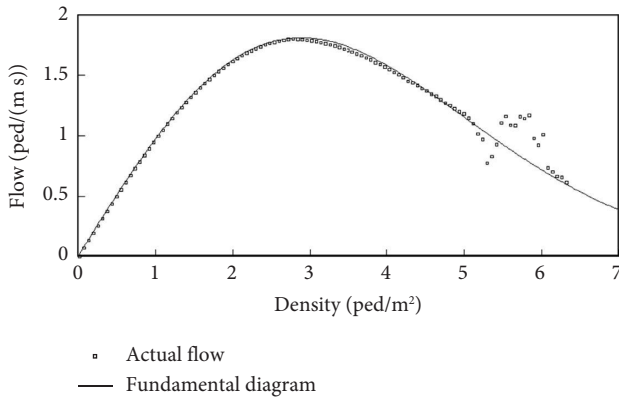

FIGURE 8: Local density-flow relationship.

FIGURE 9: Sample triangular mesh for example 2.2 ($N = 8793$).

conducted a postdisaster analysis of a Hajj crowd disaster and related it to the forward and backward moving shock waves in the dense crowd. In our model, we generate the waves by considering the aggregated pushing pressure created by the panic, thereby reproducing this complex phenomenon. Therefore, the realistic nature of the results of the continuum model is validated.

4.2.2. Example 2.2. Example 2.2 corresponds to a more realistic computational domain than that in Example 2.1. Three circular obstacles are placed at $(x, y) = \{(60, 25), (80, 10), \text{ and } (80, 40)\}$ with radii of $r = \{10, 6, \text{ and } 6\}$ (unit: m), respectively. This complex region cannot be easily divided by structured meshes, and thus, the traditional FD or FV methods cannot be applied. Therefore, we apply the DG methods over unstructured triangular meshes, as illustrated in Figure 9.

Figure 10 plots the density distribution during the evacuation at different time points. Before $t = 150$ s, the pedestrian flow is characterized by calm behaviors; the pedestrians try to avoid the obstacles and follow the shortest path to their destinations (see subfigure 10(a)). However, the three circular obstacles cause congestion upstream and
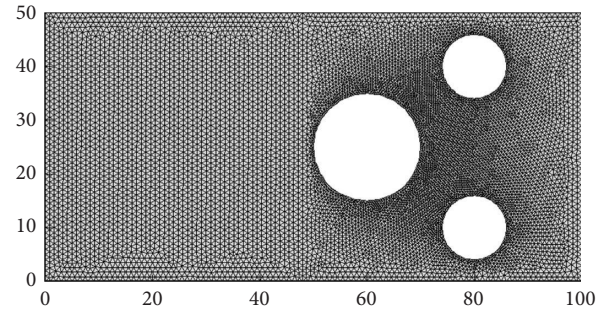
downstream of the obstacles, with the maximum density being approximately 5.3|ped/m² (see subfigure 10(b)). This maximum density is smaller than that in example 2.1 (with rectangular obstacles), which suggests that the influence of circular boundaries on pedestrian movement is weaker. This finding is supported by existing simulation results [52, 53]. After $t = 150$ s, panic begins to propagate in the crowd (see subfigures 10(c)–10(f)). The density gradually increases to approximately 5.8|ped/m², following a similar pattern to that in example 2.1. The pedestrians continue to enter the flow, generating a high pushing pressure in the dense crowd, as shown in Figure 11. The propagated pushing pressure increases to approximately 110|N/m, considerably lower than that in example 2.1 (see subfigures 11(a) and 11(b)).

4.2.3. Illustration of Instability Using the DG Method. At low sonic speeds ($c_0 = 0.1$|m/s in this study), the simulation results exhibit stop-and-go waves. At $t = 150$ s, three stop-and-go waves are observed near the largest circular obstacle, as illustrated in Figure 12(a). These waves are generated near the bottleneck and propagate backward to the low-density area until they vanish. Moreover, when the pedestrians panic, small and dense waves are generated and propagated continuously in the high-density regions, as illustrated in Figure 12(b). This phenomenon is similar to the turbulence
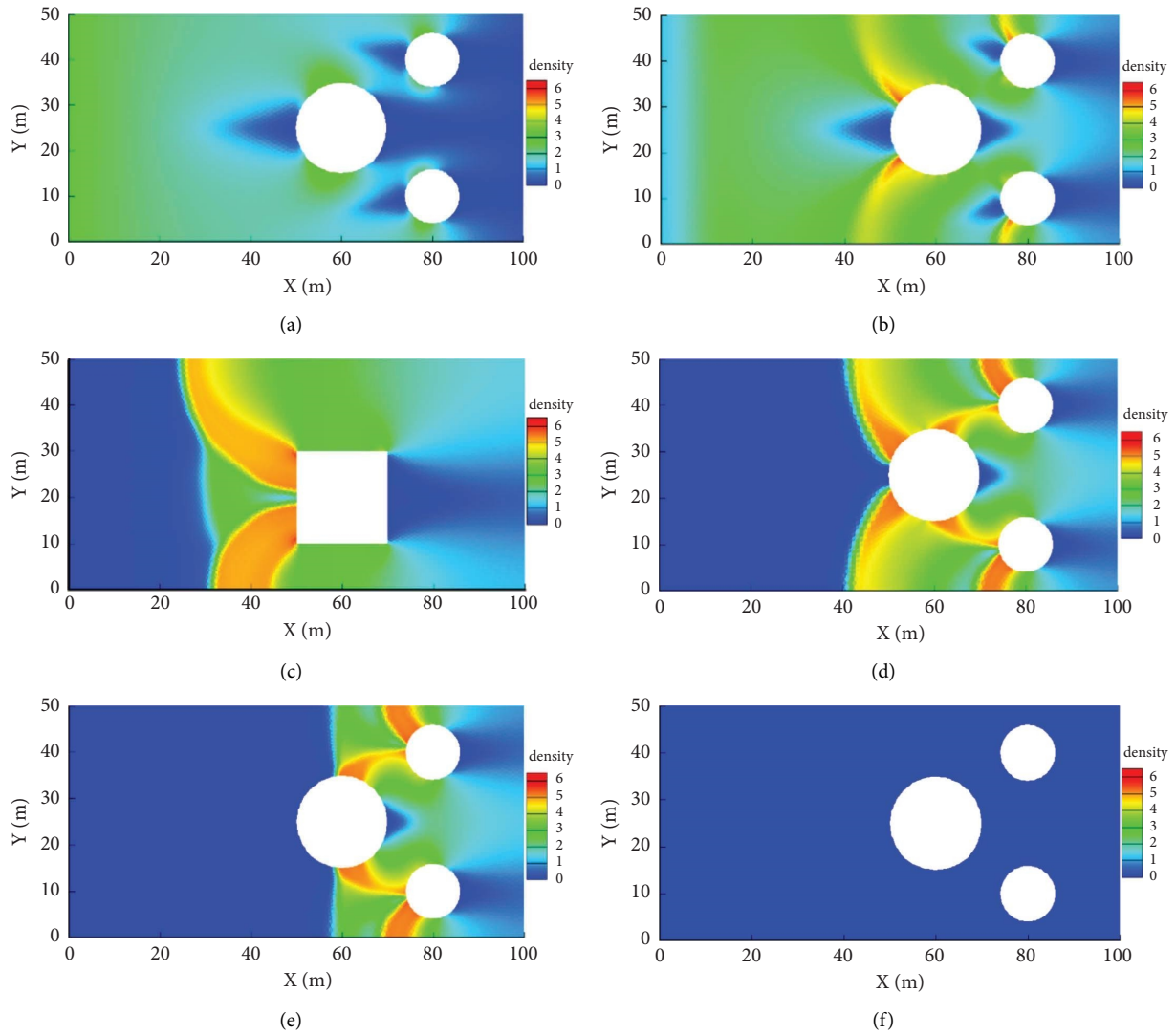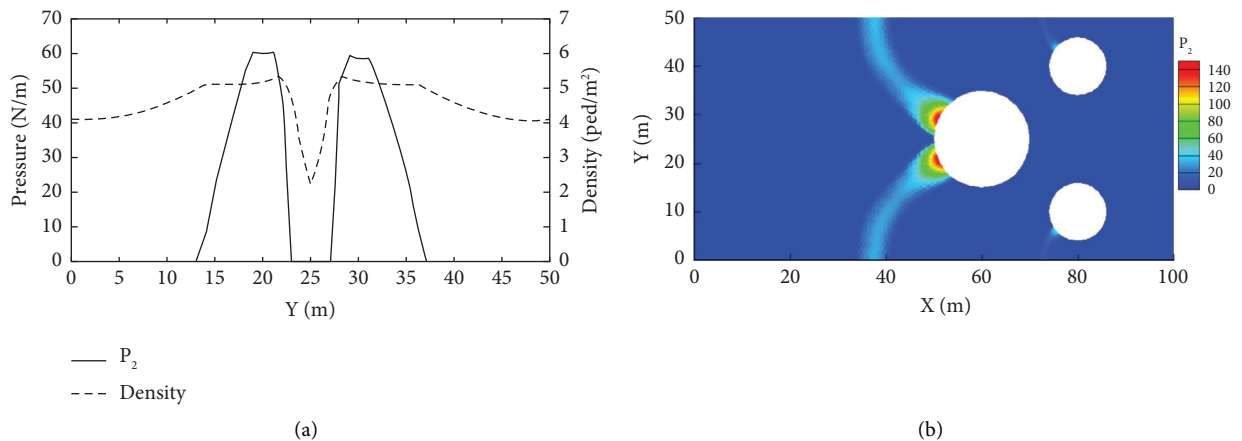
(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 10: Density distribution (unit: ped/m^2) simulated in example 2.2 using the second-order DG method ($c_0 = 0.5$ m/s). The number of elements in the triangular mesh is $N = 8793$. (a) $t = 100$ s. (b) $t = 150$ s. (c) $t = 200$ s. (d) $t = 250$ s. (e) $t = 300$ s. (f) $t = 400$ s.

— P$_2$
--- Density

(a)

(b)

FIGURE 11: Pressure distribution simulated at $t = 200$ s in example 2.2 using the DG method ($c_0 = 0.5$ m/s). (a) Density curve and pushing pressure curve corresponding to $x = 48$ m. (b) Contour plot of the distribution of the pushing pressure (unit: N/m).
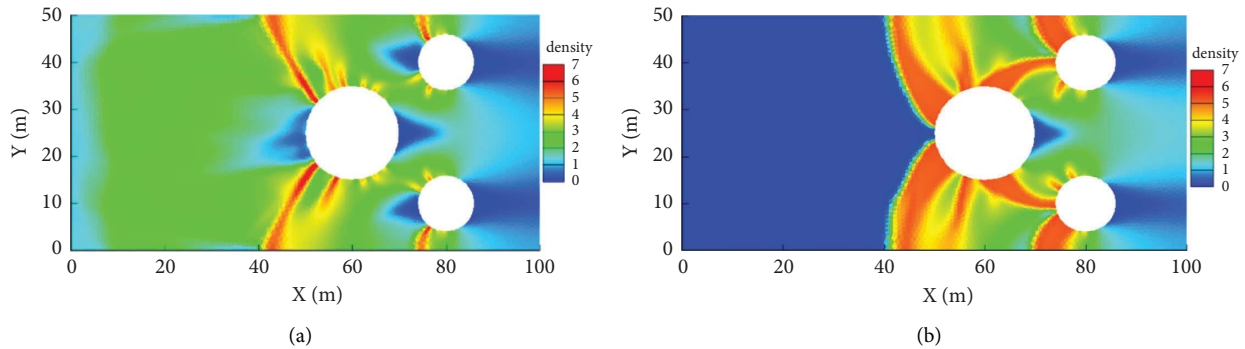
FIGURE 12: Simulated density distribution (unit: ped/m^2) in example 2.2 using the second-order DG method ($c_0 = 0.1$ m/s). The number of elements in the triangular mesh is $N = 8793$. (a) $t = 150$ s. (b) $t = 250$ s.

observed by Fruin [52], who indicated that shockwaves can propagate through dense crowds. The DG scheme can successfully reproduce the instability of crowd dynamics induced at low sonic speeds, which validates the continuum model.

5. Conclusion

Several commonly used continuum pedestrian flow models are reviewed. According to the equilibrium state of speed, these models can be classified into first- and high-order models. The first-order models include the traditional LWR model and other extended LWR models, coupled with RDUE and PDUE route strategies. High-order models include the PW model, PW-like model, Aw–Zhang model, and PWP model. Both types of models can be rewritten in a unified form as a scalar or system conservation law with different conservation variables, flow fluxes, and source terms. To address problems in real and complex computational regions, a high-order DG scheme with a positivity-preserving limiter is applied to solve the conservation law, and a second-order fast-sweeping scheme is used to solve the Eikonal equations. The proposed solution algorithm can efficiently solve all types of pedestrian dynamic models considered in this study. We provide a numerical example that has the exact solution to demonstrate the accuracy of the high-order scheme and validate the proposed numerical method through numerical examples involving evacuation cases with a rectangular obstacle and three circular obstacles based on the PWP model. The numerical results (which indicate the occurrence of the second-peak phenomenon and stop-and-go waves) are consistent with those reported in our previous work. Thus, the proposed numerical algorithm is accurate and effective.

Notably, because of the limitations of the fast-sweeping method for solving the Eikonal equation on triangular meshes, the complete algorithm is no more than second-order accurate. Future research can be aimed at developing numerical methods with higher-order accuracy to solve the Eikonal equation on triangular meshes to ensure that the algorithm can reach third-order accuracy or higher with a higher-order DG scheme. In this manner, the solution efficiency and applicability to more complicated real-world cases can be enhanced.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Liangze Yang and Haoyang Liang were responsible for conceptualization, methodology, software, and writing of the original draft. S. C. Wong and Jie Du were responsible for conceptualization, methodology, review writing, and Editing.

Acknowledgments

References

[1] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[2] P. A. Langston, R. Masling, and B. N. Asmar, "Crowd dynamics discrete element multi-circle model," *Safety Science*, vol. 44, no. 5, pp. 395–417, 2006.

[3] C. Y. Chu, "A computer model for selecting facility evacuation design using cellular automata," *Computer-Aided Civil and Infrastructure Engineering*, vol. 24, no. 8, pp. 608–622, 2009.

[4] Q. Sun and S. Wu, "A configurable agent-based crowd model with generic behavior effect representation mechanism,"

Computer-Aided Civil and Infrastructure Engineering, vol. 29, no. 7, pp. 531–545, 2014.

[5] L. Crociani and G. Lämmel, "Multidestination pedestrian flows in equilibrium: a cellular automaton-based approach," *Computer-Aided Civil and Infrastructure Engineering*, vol. 31, no. 6, pp. 432–448, 2016.

[6] S. Zhang, J. Zhang, M. Chraibi, and W. Song, "A speed-based model for crowd simulation considering walking preferences," *Communications in Nonlinear Science and Numerical Simulation*, vol. 95, Article ID 105624, 2021.

[7] K. Cao, Y. Chen, D. Stuart, and D. Yue, "Cyber-physical modeling and control of crowd of pedestrians: a review and new framework," *IEEE/CAA Journal of Automatica Sinica*, vol. 2, no. 3, pp. 334–344, 2015.

[8] M. J. Lighthill and G. B. Whitham, "On kinematic waves II. A theory of traffic flow on long crowded roads," *Proceedings of the Royal Society of London- Series A: Mathematical and Physical Sciences*, vol. 229, no. 1178, pp. 317–345, 1955.

[9] P. I. Richards, "Shock waves on the highway," *Operations Research*, vol. 4, no. 1, pp. 42–51, 1956.

[10] B. D. Greenshields, "A study of traffic capacity," *Proceedings of the Highway Research Board*, vol. 14, pp. 448–477, 1934.

[11] G. C. K. Wong and S. C. Wong, "A multi-class traffic flow model–an extension of LWR model with heterogeneous drivers," *Transportation Research Part A: Policy and Practice*, vol. 36, no. 9, pp. 827–841, 2002.

[12] R. L. Hughes, "A continuum theory for the flow of pedestrians," *Transportation Research Part B: Methodological*, vol. 36, no. 6, pp. 507–535, 2002.

[13] S. P. Hoogendoorn and P. H. Bovy, "Pedestrian route-choice and activity scheduling theory and models," *Transportation Research Part B: Methodological*, vol. 38, no. 2, pp. 169–190, 2004.

[14] S. P. Hoogendoorn and P. H. Bovy, "Dynamic user-optimal assignment in continuous time and space," *Transportation Research Part B: Methodological*, vol. 38, no. 7, pp. 571–592, 2004.

[15] L. Huang, S. C. Wong, M. Zhang, C. W. Shu, and W. H. Lam, "Revisiting Hughes' dynamic continuum model for pedestrian flow and the development of an efficient solution algorithm," *Transportation Research Part B: Methodological*, vol. 43, no. 1, pp. 127–141, 2009.

[16] T. Xiong, M. Zhang, C. W. Shu, S. C. Wong, and P. Zhang, "High-order computational scheme for a dynamic continuum model for bi-directional pedestrian flows," *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 4, pp. 298–310, 2011.

[17] J. Du, S. C. Wong, C. W. Shu, T. Xiong, M. Zhang, and K. Choi, "Revisiting Jiang's dynamic continuum model for urban cities," *Transportation Research Part B: Methodological*, vol. 56, pp. 96–119, 2013.

[18] D. Helbing, A. Johansson, and H. Z. Al-Abideen, "Dynamics of crowd disasters: an empirical study," *Physical Review A*, vol. 75, no. 4, Article ID 046109, 2007.

[19] H. J. Payne, "Models of freeway traffic and control," *Mathematical Models of Public Systems*, vol. 28, no. 1, pp. 51–61, 1971.

[20] G. B. Whitham, *Linear and Nonlinear Waves*, John Wiley and Sons, New York, NY, USA, 1974.

[21] H. M. Zhang, "Analyses of the stability and wave properties of a new continuum traffic theory," *Transportation Research Part B: Methodological*, vol. 33, no. 6, pp. 399–415, 1999.

[22] H. M. Zhang, "Structural properties of solutions arising from a nonequilibrium traffic flow theory," *Transportation Research Part B: Methodological*, vol. 34, no. 7, pp. 583–603, 2000.

[23] C. F. Daganzo, "Requiem for second-order fluid approximations of traffic flow," *Transportation Research Part B: Methodological*, vol. 29, no. 4, pp. 277–286, 1995.

[24] A. A. T. M. Aw and M. Rascle, "Resurrection of "second order" models of traffic flow," *SIAM Journal on Applied Mathematics*, vol. 60, no. 3, pp. 916–938, 2000.

[25] H. M. Zhang, "A non-equilibrium traffic model devoid of gas-like behavior," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 275–290, 2002.

[26] H. Liang, J. Du, and S. C. Wong, "A continuum model for pedestrian flow with explicit consideration of crowd force and panic effects," *Transportation Research Part B: Methodological*, vol. 149, pp. 100–117, 2021.

[27] R. Jiang, Q. S. Wu, and Z. J. Zhu, "A new continuum model for traffic flow and numerical tests," *Transportation Research Part B: Methodological*, vol. 36, no. 5, pp. 405–419, 2002.

[28] R. Zhao, Q. Hu, Q. Liu, C. Li, D. Dong, and Y. Ma, "Panic propagation dynamics of high-density crowd based on information entropy and Aw-Rascle model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4425–4434, 2020.

[29] K. Huang, X. Di, Q. Du, and X. Chen, "A game-theoretic framework for autonomous vehicles velocity control: bridging microscopic differential games and macroscopic mean field games," 2019, https://arxiv.org/abs/1903.06053.

[30] Y. Q. Jiang, P. Zhang, S. C. Wong, and R. X. Liu, "A higher-order macroscopic model for pedestrian flows," *Physica A: Statistical Mechanics and Its Applications*, vol. 389, no. 21, pp. 4623–4635, 2010.

[31] W. H. Reed and T. R. Hill, *Triangular Mesh Methods for the Neutron Transport Equation (No. LA-UR-73-479; CONF-730414-2)*, Los Alamos Scientific Lab, Los Alamos, NM, USA, 1973.

[32] B. Cockburn and C. W. Shu, "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework," *Mathematics of Computation*, vol. 52, no. 186, pp. 411–435, 1989.

[33] B. Cockburn, S. Y. Lin, and C. W. Shu, "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems," *Journal of Computational Physics*, vol. 84, no. 1, pp. 90–113, 1989.

[34] B. Cockburn, S. Hou, and C. W. Shu, "The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case," *Mathematics of Computation*, vol. 54, no. 190, pp. 545–581, 1990.

[35] B. Cockburn and C. W. Shu, "The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems," *Journal of Computational Physics*, vol. 141, no. 2, pp. 199–224, 1998.

[36] C. W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock-capturing schemes," *Journal of Computational Physics*, vol. 77, no. 2, pp. 439–471, 1988.

[37] C. W. Shu, "TVB uniformly high-order schemes for conservation laws," *Mathematics of Computation*, vol. 49, no. 179, pp. 105–121, 1987.

[38] Y. Xu and C.-W. Shu, "Local discontinuous Galerkin methods for the Kuramoto-Sivashinsky equations and the Ito-type coupled KdV equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 25-28, pp. 3430–3447, 2006.

[39] H. Liu and J. Yan, "The direct discontinuous Galerkin (DDG) methods for diffusion problems," *SIAM Journal on Numerical Analysis*, vol. 47, no. 1, pp. 675–698, 2009.

[40] B. Cockburn, D. Pietro, and A. Ern, "Bridging the hybrid high-order and hybridizable discontinuous Galerkin methods," *Esaim: Mathematical Modelling And Numerical Analysis*, vol. 50, 2016.

[41] J. Lu, Y. Liu, and C. W. Shu, "An oscillation-free discontinuous Galerkin method for scalar hyperbolic conservation laws," *SIAM Journal on Numerical Analysis*, vol. 59, no. 3, pp. 1299–1324, 2021.

[42] X. Zhang and C. W. Shu, "On maximum-principle-satisfying high order schemes for scalar conservation laws," *Journal of Computational Physics*, vol. 229, no. 9, pp. 3091–3120, 2010.

[43] X. Zhang and C. W. Shu, "On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes," *Journal of Computational Physics*, vol. 229, no. 23, pp. 8918–8934, 2010.

[44] X. Zhang, Y. Xia, and C. W. Shu, "Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes," *Journal of Scientific Computing*, vol. 50, no. 1, pp. 29–62, 2012.

[45] A. Johansson, D. Helbing, H. Z. Al-Abideen, and S. Al-Bosta, "From crowd dynamics to crowd safety: a video-based analysis," *Advances in Complex Systems*, vol. 11, no. 04, pp. 497–527, 2008.

[46] J. Qian, Y. T. Zhang, and H. K. Zhao, "Fast sweeping methods for Eikonal equations on triangular meshes," *SIAM Journal on Numerical Analysis*, vol. 45, no. 1, pp. 83–107, 2007.

[47] Y. Xia, S. C. Wong, M. Zhang, C. W. Shu, and W. H. Lam, "An efficient discontinuous Galerkin method on triangular meshes for a pedestrian flow model," *International Journal for Numerical Methods in Engineering*, vol. 76, no. 3, pp. 337–350, 2008.

[48] C. E. Nicholson and B. Roebuck, "The investigation of the hillsborough disaster by the health and safety executive," *Safety Science*, vol. 18, no. 4, pp. 249–259, 1995.

[49] X. Zhang and C. W. Shu, "Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms," *Journal of Computational Physics*, vol. 230, no. 4, pp. 1238–1248, 2011.

[50] D. Helbing and P. Mukerji, "Crowd disasters as systemic failures: analysis of the Love Parade disaster," *EPJ Data Science*, vol. 1, no. 1, pp. 7–40, 2012.

[51] J. J. Fruin, "The causes and prevention of crowd disasters," *Engineering for Crowd Safety*, vol. 1, no. 10, pp. 99–108, 1993.

[52] N. Shiwakoti, X. Shi, and Z. Ye, "A review on the performance of an obstacle near an exit on pedestrian crowd evacuation," *Safety Science*, vol. 113, pp. 54–67, 2019.

[53] G. S. Jiang and S. H. Yu, "Discrete shocks for finite difference approximations to scalar conservation laws," *SIAM Journal on Numerical Analysis*, vol. 35, no. 2, pp. 749–772, 1998.