






Discrete unified gas kinetic scheme for electrostatic plasma and its comparison with the particle-in-cell method

Hongtao Liu ^{1,*} Lulu Quan ^{1,†} Qing Chen ^{2,‡} Shengjin Zhou ^{1,§} and Yong Cao ^{1,||}

¹*School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, Guangdong 518055, China*

²*College of Mechanical and Electronic Engineering, Nanjing Forestry University, Nanjing 210037, China*

 (Received 23 December 2019; revised manuscript received 21 February 2020; accepted 27 March 2020; published 27 April 2020)

In this paper, we present a finite-volume direct kinetic method, the so-called discrete unified gas kinetic scheme (DUGKS), for electrostatic plasma. One key feature of this method is the semi-implicit unsplitting treatment of particle transport and collision, and thus the time step in current DUGKS is not limited by the particle collision time. In addition, a fourth-order compact MUSCL scheme with a positivity preserving limiter is implemented in the interface reconstruction, which enables present DUGKS to preserve the favorable conservative property and positivity of distribution function. Combined with this MUSCL method, the semi-Lagrangian scheme is used for the particle transport in velocity space to remove Courant-Friedricks-Lewy restriction induced by the large electric force. As a result, the proposed DUGKS becomes an efficient and stable multiscale scheme. Several numerical experiments, including plasma sheath, linear Landau damping, collisional nonlinear Landau damping, and plasma ion acceleration, are performed to validate current DUGKS. A comparative study of current DUGKS with a general particle in cell (PIC) method which could handle particle collision in a conservative way is also presented. Theory and numerical experiments demonstrate that DUGKS is preferred for plasma flows involving small electrostatic perturbation and high collision regimes, while the PIC method is desired for the field-dominated plasma flows involving a wide range of velocities.

DOI: [10.1103/PhysRevE.101.043307](https://doi.org/10.1103/PhysRevE.101.043307)

I. INTRODUCTION

The importance of plasma is constantly increasing in various fields, such as semiconductors, space propulsion, material processing, medicine, and so on [1]. The modeling and numerical simulation of plasma is a quite active research field in the plasma community for many years. Roughly speaking, two large classes of mathematical models are available: fluid and kinetic models. It is well accepted that the fluid models will become less accurate once the continuum assumptions begin to break down [2]. Given the multiscale characteristic of the plasma flow, e.g., the flow may covering a wide range of Knudsen number (Kn), where Kn is the ratio of particle free path to characteristic length, the kinetic model is desired if a unified method is required.

One basic kinetic model for electrostatic plasma simulations is the collisional Vlasov (Boltzmann) equation, coupled with the Poisson equation. If the Bhatnagar-Gross-Krook (BGK) collision term [3] is considered, the concerned model is called the BGK-Vlasov-Poisson (BGK-VP) equations. The numerical resolution of this kinetic model can mainly be classified into two categories: particle-based method and

grid-based method. The particle in cell (PIC) is the well-known particle-based method, which is based on the tracing the motion of finite number of macroparticles. The biggest advantage of the PIC method is its computation economy, which is easily and efficiently applied to the high-dimensional problems [4,5]. We refer two celebrated books [6,7] and a comprehensive review [8] for an overview of PIC method.

One key drawback in traditional PIC method is the unphysical grid-heating, which would lead to the increase of total energy. Several semi-implicit [9,10] and fully implicit PIC [11] solvers have been proposed to alleviate this issue. However, the PIC method suffers from inherent numerical noise, especially for the plasma flow with a small perturbation [12]. Although some improved methods were proposed [13–15], there still is not a well-accepted denoising technique in the plasma community. In the collisional regime, the PIC method coupled with Monte Carlo collision (PIC-MCC) is usually used [16,17]. Nevertheless, due to the splitting treatment of particle transport and collision, the time step and grid spacing in PIC-MCC are limited by the particle collision time and the mean free path, respectively. As a result, this method suffers from the expensive cost in the high collision (hydrodynamic) regime, where the mean free path is very small with respect to characteristic length.

An alternative to the PIC method is represented by directly solving kinetic equation in the phase space [18] by means of the grid-based method, which is referred to as the direct kinetic method (DKM) in this paper, such as high-order finite difference [19,20], finite-element [21–24], finite-volume

*liuhongtao@stu.hit.edu.cn

†15b953016@stu.hit.edu.cn

‡qchen@njfu.edu.cn

§18S053173@stu.hit.edu.cn

||Corresponding author: yongc@hit.edu.cn

[25–28], and the spectral methods [12,29–32], see Refs. [33–35] for recent review of these methods. Compared with the PIC method, as a deterministic method, the DKM is superior in reducing numerical noise. Another advantage of DKM is to achieve high-order accuracy for phase-space discretization, which allows the study of fine-scale detail that are typically inaccessible in PIC. However, the high-order schemes sometimes generate spurious oscillations and become unstable in long-time simulations [27]. Several efforts have been devoted to deal with numerical oscillations, such as MUSCL scheme [36,37], essentially nonoscillatory scheme [38] and weighted essentially nonoscillatory scheme [20,28,39]. Although most of them enforce the conservation of mass, all other conserved quantities, i.e., momentum, total energy, and entropy, can be accurate only to within truncation error. Besides, positivity preserving (PP) property is also highly desired in DKM, since nonpositive solution values may trigger unphysical oscillations. To guarantee positivity, some other mechanism [23,26,27,40] needs to be considered.

Another basic but very important problem in DKM is the handling of particle collision. In traditional DKM, the particle collision and transport process are explicitly uncoupled through splitting method, which involve high cost in strong collision regime due to the explicit treatment of collision term. To develop an efficient DKM for all Kn regimes, the asymptotic preserving (AP) schemes [41–45] are proposed. These AP schemes are stable with respect to arbitrary Kn, while their temporal step sizes are not restricted by Kn. Among these AP schemes, we especially appreciate favorable stability, efficiency, and accuracy of discrete unified gas kinetic scheme (DUGKS) [46–49] in all Kn regimes. The previous DUGKS is originally designed for neutral gas flows, where the conservation and PP property have not got much attention. However, these two properties are crucial in plasma simulation, since they affect the stability of the solver. In this study, we aim to extend the DUGKS to plasma simulation and improve its conservation and PP property in lengthy simulations.

To now, the PIC method still represents the preferred approach in plasma community. Although DKM solver and PIC seek the same ends, they differ fundamentally in their approach. Thus, it is necessary to implement the newly developed DKM solver to benchmark once and again the PIC code against it. Denavit and Kruer [50] presented a close quantitative comparison of two methods for two-stream instabilities and large-amplitude electron oscillations. More recently, Camporeale *et al.* [12] compared the computational efficiency and the cost-effectiveness of a novel DKM, i.e., the so-called Fourier-Hermite spectral method, and fully implicit PIC method. Besides, Saini *et al.* [51] presented a quantitative comparative study of two methods for the formation of coherent phase-space structures. The previous comparative studies [12,29,50,51] are only for collisionless plasma with periodic boundaries.

In this study, we present a novel DUGKS for electrostatic plasma with general boundary conditions, which is a nontrivial extension of our conserved DUGKS [48,49]. First, the semi-implicit coupling of particle transport and collision is implemented in the distribution function (DF) evolution and flux evaluation, which enables proposed method to

economically and accurately provide a satisfactory solution for all Kn regimes. Second, a fourth-order compact MUSCL scheme with PP limiter is used in the reconstruction to enable present DUGKS to preserve the favorable conservation and positivity. Besides, combined with this MUSCL method, the Eulerian scheme is imposed on DF evolution in physical space to easily handle general boundary conditions, while the semi-Lagrangian scheme is used for the evolution in velocity space to remove Courant-Friedricks-Lewy (CFL) restriction induced by the large electric field. Then we present numerical solutions of current DUGKS for particles-dominant plasma, including linear Landau damping, collisional nonlinear Landau damping and plasma sheath, as well as field-dominant plasma, e.g., plasma ion acceleration, and compare those solutions with results of a general PIC method which could handle particle collision in a conservative way.

The rest of the paper is organized as follows. The BGK-VP system for electrostatic plasma and its nondimensionalization are introduced in Sec. II. In Sec. III, a novel DUGKS and general PIC method are described in detail. We present the results of comparative numerical studies in Sec. IV. Finally, a summary is given in Sec. V.

II. THE BGK-VLASOV-POISSON SYSTEM

In this section, we recall the BGK-VP system for electrostatic plasma. First, the mathematic model of the BGK-VP system and its nondimensionalization are introduced. Then the time-splitting technologies for solving the BGK-Vlasov equation are reviewed.

In this paper, unless stated otherwise, we focus on the motion of electron and assume ions form a uniform static background. If BGK collision model is used [3], then the motion of electron can be described by BGK-VP equation

$$\begin{aligned} \frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \frac{q}{m} \nabla \phi \cdot \nabla_{\mathbf{v}} f &= -\frac{1}{\tau} (f - f^{\text{eq}}), \\ -\Delta \phi &= \frac{q}{\varepsilon} (n_i - n), \quad n = \int f d\mathbf{v}, \end{aligned} \quad (1)$$

where $f = f(\mathbf{x}, \mathbf{v}, t)$ is the velocity distribution function for particles moving in D -dimensional velocity space with $\mathbf{v} = (v_1, \dots, v_D)$ at position $\mathbf{x} = (x_1, \dots, x_D)$ and time t . Here the potential ϕ , electron mass m , electron number density n , vacuum permittivity ε , elementary charge q , and the background ion number density n_i are the plasma parameters. Besides, τ is the collision relaxation time, and f^{eq} is the Maxwellian equilibrium distribution function:

$$f^{\text{eq}} = \frac{n}{(2\pi RT)^{D/2}} \exp\left(-\frac{c^2}{2RT}\right), \quad (2)$$

where $R = k_B/m$ is the gas constant with Boltzmann constant k_B , T is the temperature, and $\mathbf{c} = (\mathbf{v} - \mathbf{u})$ is the peculiar velocity with macroscopic velocity \mathbf{u} . In addition, it is well known that the Debye length λ and the electron plasma frequency ω_p are two important plasma parameters, which are given by

$$\lambda = \left(\frac{\varepsilon k_B T}{q^2 n}\right)^{1/2}, \quad \omega_p = \left(\frac{n q^2}{\varepsilon m}\right)^{1/2}.$$

For simplicity, the following dimensionless variables are used:

$$\begin{aligned}\bar{x} &= \frac{x}{x_0}, \quad \bar{T} = \frac{T}{T_0}, \quad \bar{m} = \frac{m}{m_0}, \quad \bar{n} = \frac{n}{n_0}, \\ \bar{u} &= \frac{u}{u_0}, \quad \bar{t} = \frac{t}{t_0}, \quad \bar{f} = \frac{f}{f_0}, \quad \bar{\phi} = \frac{\phi}{\phi_0},\end{aligned}\quad (3)$$

where x_0 , T_0 , m_0 , and n_0 are reference length, temperature, mass, and number density, which are independent parameters. Then the reference velocity $u_0 = \sqrt{k_B T_0 / m_0}$, time $t_0 = x_0 / u_0$, potential $\phi_0 = k_B T_0 / q$, and distribution function $f_0 = n_0 / u_0^D$ are further obtained. For the sake of simplicity, we set dimensionless collision relaxation time $\bar{\tau} = \text{Kn}$, where Kn is Knudsen number which is defined as the ratio of the mean free path and the characteristic length.

In this paper, unless stated otherwise, we chose $m_0 = m_e$ and $n_0 = n_i$, where m_e is the electron mass. Define the dimensionless Debye length $\bar{\lambda} = \lambda / x_0$, and then the dimensionless electron plasma frequency is $\bar{\omega}_p = \omega_p t_0 = 1 / \bar{\lambda}$. In the rest of the paper, all variables are dimensionless unless stated otherwise, but we will drop the bar over the variables for simplicity. Then the dimensionless BGK-VP system Eq. (1) can be written as

$$\begin{aligned}\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \nabla \phi \cdot \nabla_{\mathbf{v}} f &= -\frac{1}{\tau}(f - f^{\text{eq}}), \\ \lambda^2 \Delta \phi &= n - 1, \quad n = \int f d\mathbf{v}.\end{aligned}\quad (4)$$

For the gas kinetic method, the key is to obtain the distribution function f . Once the distribution function f is known, the conservative variables \mathbf{W} can be obtained by

$$\mathbf{W} = (n, \mathbf{nu}, nE_t)^T = \int \boldsymbol{\psi} f d\mathbf{v}, \quad (5)$$

where $\boldsymbol{\psi} = (1, \mathbf{v}, \mathbf{v}^2/2)^T$ is the collision invariant and E_t is the kinetic energy. Furthermore, the temperature can be computed from \mathbf{W} :

$$T = \frac{2}{D} \left(E_t - \frac{1}{2} n u^2 \right). \quad (6)$$

For Eq. (4), the key is to solve the BGK-Vlasov equation

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \nabla \phi \cdot \nabla_{\mathbf{v}} f = -\frac{1}{\tau}(f - f^{\text{eq}}), \quad (7)$$

which is usually solved by time splitting method [18]. The traditional time splitting form of Eq. (7) can be written as

$$\begin{aligned}\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f &= 0, \\ \frac{\partial f}{\partial t} + \nabla \phi \cdot \nabla_{\mathbf{v}} f &= 0, \\ \frac{\partial f}{\partial t} &= -\frac{1}{\tau}(f - f^{\text{eq}}).\end{aligned}\quad (8)$$

Clearly, Eq. (8) shows the particle transport, particle acceleration, and particle collision process, respectively. Note that the first two equations are linear hyperbolic equations, which can be solved by the conservative scheme based on characteristic line and the last equation has an analytical solution. Thus, Eq. (8) is not difficult to solve and is widely used in both particle-based method [16,17] and DKM [52]. However, the splitting treatment of particle transport and collision process

requires spatial and temporal sizes are less than particle mean free path and collision time, which will involve expensive cost in high collision regime. An alternative time splitting form of Eq. (7) can be written as

$$\begin{aligned}\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f &= -\frac{1}{\tau}(f - f^{\text{eq}}), \\ \frac{\partial f}{\partial t} + \nabla \phi \cdot \nabla_{\mathbf{v}} f &= 0.\end{aligned}\quad (9)$$

The main feature of Eq. (9) is the unsplitting treatment of particle transport and particle collision, which has been used in several DKM [44,46,48]. Note that slightly different from Ref. [49], to have a better comparative study, the particle transport and particle acceleration are uncoupled in this study. In the following section, we will present a general particle-based method with collision term based on Eq. (8) and develop a novel positivity preserving DKM based on Eq. (9).

III. NUMERICAL METHODS

In this section, we will present two different kinetic methods for electrostatic plasma: a novel DKM, i.e., the discrete unified gas kinetic scheme (DUGKS), and a general PIC method. The evolution procedure of DUGKS and the general PIC method will be presented in Sec. III A and Sec. III B, respectively.

A. Discrete unified gas kinetic scheme

The proposed DUGKS for electrostatic plasma is a positivity preserving (PP) scheme, where the particle transport is coupled with particle collision, and a fourth-order MUSCL scheme with PP limiter is used in interface reconstruction. The major steps of the method include the update rule of distribution function, flux evaluation in phase space, and the interface reconstruction with a PP limiter, which will be presented in detail as follows.

1. Update rule

The starting point of proposed DUGKS is the Eq. (9). In the practical implementation, the velocity \mathbf{v} is discretized into a finite set of discrete velocities $\{\mathbf{v}_i\}$, where $\mathbf{i} = (i_1, i_2, \dots, i_D)$. Then Eq. (9) becomes

$$\begin{aligned}\frac{\partial f_i}{\partial t} + \mathbf{v}_i \cdot \nabla f_i &= \Omega_i, \\ \frac{\partial f_i}{\partial t} + \nabla \phi \cdot \nabla_{\mathbf{v}} f_i &= 0,\end{aligned}\quad (10)$$

where $\Omega_i = -(f_i - f_i^{\text{eq}}) / \tau$ is the collision term.

The DUGKS uses the idea of the finite volume method. The physical space and velocity space are divided into the control volume V_j and U_i , respectively. As a result, integrating Eq. (10) on physical space volume $|V_j|$ centered at physical cell j and the velocity space volume $|U_i|$ centered at velocity cell \mathbf{i} , from time t_k to $t_{k+1} = t_k + \Delta t$, one can obtain

$$f_{j,i}^b = f_{j,i}^k - \frac{\Delta t}{|V_j|} \mathbf{F}_{j,i}^{k+1/2} + \frac{\Delta t}{2} [\Omega_{j,i}^b + \Omega_{j,i}^k], \quad (11)$$

$$f_{j,i}^{k+1} = f_{j,i}^b - \frac{\Delta t}{|U_i|} \mathbf{G}_{j,i}^{b+1/2}, \quad (12)$$

Note that $f_{j,i}^b$ at time t_b is the intermedium variable between $f_{j,i}^k$ and $f_{j,i}^{k+1}$. Besides, $\mathbf{F}_{j,i}^{k+1/2}$ and $\mathbf{G}_{j,i}^{b+1/2}$ are the physical and velocity microflux, respectively. The physical microflux $\mathbf{F}_{j,i}^{k+1/2}$ is given by

$$\mathbf{F}_{j,i}^{k+1/2} = \sum_p \mathbf{v}_i \cdot \mathbf{A}_j^p f(\mathbf{x}_j^p, \mathbf{v}_i, t_{k+1/2}), \quad (13)$$

where \mathbf{x}_j^p is center of p th face of physical cell j and \mathbf{A}_j^p is the outward normal vector of physical face with area $|\mathbf{A}_j^p|$. The velocity microflux $\mathbf{G}_{j,i}^{b+1/2}$ is given by

$$\mathbf{G}_{j,i}^{b+1/2} = \sum_q \nabla \phi_j^b \cdot \mathbf{B}_i^q f(\mathbf{x}_j, \mathbf{v}_i^q, t_{b+1/2}), \quad (14)$$

where \mathbf{v}_i^q is center of q th face of velocity cell i and \mathbf{B}_i^q is the outward normal vector of velocity face with area $|\mathbf{B}_i^q|$.

It should be noted that the update rule given by Eq. (11) is implicit, since unknown conservative variables are required for the evaluation of $f_{j,i}^{\text{eq},b}$ and τ_j^b in collision term $\Omega_{j,i}^b$. In order to remove this implicit requirement, similarly to the conserved DUGKS [48], the conservative variables are updated as follows. Taking conservation moment ψ on Eq. (11), and given the conservative properties of the collision operators, the conservative variables \mathbf{W}_j^b can then be obtained by

$$\mathbf{W}_j^b = \mathbf{W}_j^k - \frac{\Delta t}{|V_j|} \int \psi \mathbf{F}_j^{k+1/2} d\mathbf{v}, \quad (15)$$

where $\mathbf{F}_j^{k+1/2}$ is the collection of $\mathbf{F}_{j,i}^{k+1/2}$ at discrete velocity space \mathbf{v}_i .

Then Eq. (11) can be written as the following explicit scheme:

$$f_{j,i}^b = \left(1 + \frac{\Delta t}{2\tau_j^b}\right)^{-1} \left[f_{j,i}^n + \frac{\Delta t}{2} \left(\frac{f_{j,i}^{\text{eq},b}}{\tau_j^b} + \frac{f_{j,i}^{\text{eq},k} - f_{j,i}^k}{\tau_j^k} \right) - \frac{\Delta t}{|V_j|} \mathbf{F}_{j,i}^{k+1/2} \right]. \quad (16)$$

Here Eqs. (15) and (16) are the update rule of macroscopic conservative variables and microscopic distribution function respectively for Eq. (11). Once $f_{j,i}^b$ is obtained, then one can get potential ϕ_j^b from the Poisson equation

$$\lambda^2 \Delta \phi_j^b = n_j^b - 1, \quad (17)$$

where $n_j^b = \int f_{j,i}^b d\mathbf{v}$. Then $f_{j,i}^{k+1}$ can be obtained from Eq. (12). To now, only physical microflux $\mathbf{F}_j^{k+1/2}$ and velocity microflux $\mathbf{G}_j^{b+1/2}$ remain to be evaluated.

2. Flux evaluation

In order to evaluate physical microflux $\mathbf{F}_j^{k+1/2}$, the first equation in Eq. (10) is integrated within a half time step $h = \Delta t/2$ along the characteristic line which ends at the physical interface center $\mathbf{x}_f(\mathbf{x}_f = \mathbf{x}_j^p)$,

$$\begin{aligned} & f(\mathbf{x}_f, \mathbf{v}_i, t_k + h) - f(\mathbf{x}_f^*, \mathbf{v}_i, t_k) \\ &= \frac{h}{2} [\Omega(\mathbf{x}_f, \mathbf{v}_i, t_k + h) + \Omega(\mathbf{x}_f^*, \mathbf{v}_i, t_k)], \end{aligned} \quad (18)$$

where $\mathbf{x}_f^* = \mathbf{x}_f - \mathbf{v}_i h$ is the particle trajectories in physical space.

Similarly to the original DUGKS [46], in order to remove the implicit collision term $\Omega(\mathbf{x}_f, \mathbf{v}_i, t_k + h)$, a new distribution function \bar{f} is introduced and defined as

$$\bar{f} = f - \frac{h}{2} \Omega = \frac{2\tau + h}{2\tau} f - \frac{h}{2\tau} f^{\text{eq}}. \quad (19)$$

From Eqs. (18) and (19), one can obtain

$$\bar{f}(\mathbf{x}_f, \mathbf{v}_i, t_k + h) = \bar{f}^+(\mathbf{x}_f - h\mathbf{v}_i, \mathbf{v}_i, t_k), \quad (20)$$

where the distribution function \bar{f}^+ is defined as

$$\bar{f}^+ = \bar{f} + \frac{h}{2} \Omega = \frac{2\tau - h}{2\tau} f + \frac{h}{2\tau} f^{\text{eq}}. \quad (21)$$

Now the focus is how to determine $\bar{f}^+(\mathbf{x}_f - h\mathbf{v}_i, \mathbf{v}_i, t_k)$. With Taylor expansion around the physical interface, one can obtain

$$\bar{f}^+(\mathbf{x}_f - h\mathbf{v}_i, \mathbf{v}_i, t_k) = \bar{f}^+(\mathbf{x}_f, \mathbf{v}_i, t_k) - h\mathbf{v}_i \cdot \boldsymbol{\sigma}_{f,i}, \quad (22)$$

where $\boldsymbol{\sigma}_{f,i} = \nabla \bar{f}^+(\mathbf{x}_f, \mathbf{v}_i, t_k)$ is the physical interface gradient. The calculation of $\bar{f}^+(\mathbf{x}_f, \mathbf{v}_i, t_k)$ and $\boldsymbol{\sigma}_{f,i}$ will be presented in the next section.

Given the conservative properties of the collision term, taking the conservation moment ψ on Eq. (19), the conservative variables $\mathbf{W}(\mathbf{x}_f, \mathbf{v}_i, t_k + h)$ can be computed from \bar{f} :

$$n = \sum w_i \bar{f}_i, \quad nu = \sum w_i \mathbf{v}_i \bar{f}_i, \quad nE_t = \frac{1}{2} \sum w_i \mathbf{v}_i^2 \bar{f}_i, \quad (23)$$

where w_i is the weights coefficient at discrete velocity point \mathbf{v}_i . Once the conservative variables $\mathbf{W}(\mathbf{x}_f, \mathbf{v}_i, t_k + h)$ are obtained, the equilibrium distribution function can be obtained from Eq. (2). Then according to Eq. (19), the original distribution function $f(\mathbf{x}_f, \mathbf{v}_i, t_k + h)$ can be obtained as well:

$$f = \frac{2\tau}{2\tau + h} \bar{f} + \frac{h}{2\tau + h} f^{\text{eq}}. \quad (24)$$

As a result, physical microflux $\mathbf{F}_j^{k+1/2}$ can be computed from Eq. (13).

Similarly, the velocity interface distribution function $f(\mathbf{x}_j, \mathbf{v}_f, t_{b+1/2})$ can be obtained from

$$f(\mathbf{x}_j, \mathbf{v}_f, t_{b+1/2}) = f(\mathbf{x}_j, \mathbf{v}_f, t_b) - h \nabla \phi_j^b \cdot \boldsymbol{\varphi}_{j,f}, \quad (25)$$

where $\mathbf{v}_f = \mathbf{v}_i^q$ is the velocity interface center and $\boldsymbol{\varphi}_{j,f} = \nabla_{\mathbf{v}} f(\mathbf{x}_j, \mathbf{v}_f, t_b)$ is the velocity interface gradient. Then the velocity microflux $\mathbf{G}_j^{b+1/2}$ can be computed from Eq. (14).

Thanks to semi-implicit coupling of particle transport and collision in distribution function evolution Eq. (16), the time step in DUGKS is not restricted by particle collision time, which will be verified by numerical test in Sec. IV C. So far, only physical and velocity interface distribution and their gradient remain to be evaluated.

3. Interface reconstruction and PP limiter

In this section, the interface reconstruction with a PP limiter is presented to obtain phase interface distribution. Here we focus on the discretization of one-dimensional transport

equation, since the extension to higher dimensions is straightforward. Take Eq. (25) for instance, the interface distribution $f(\mathbf{x}_j, \mathbf{v}_f, t_b)$ can be obtained by reconstruction from cell distribution $f(\mathbf{x}_j, \mathbf{v}_i, t_b)$ with a robust fourth-order MUSCL scheme [36]. The interface distribution function at the velocity interface \mathbf{v}_f , at time t_b reads

$$f(\mathbf{x}_j, \mathbf{v}_f, t_b) = \begin{cases} f_{i,r}, & \nabla\phi_j^b > 0, \\ f_{i+1,l}, & \nabla\phi_j^b < 0, \end{cases} \quad (26)$$

It should be noted that the reconstructed states: $f_{i,l} = f_i + (\Delta f)_i^-$ and $f_{i,r} = f_i + (\Delta f)_i^+$, where f_i is the abbreviation for $f(\mathbf{x}_j, \mathbf{v}_i, t_b)$ and $(\Delta f)_i^\pm$ are the reconstruction increments. Following [36], this fourth-order MUSCL reconstruction reads

$$\begin{aligned} (\Delta f)_i^- &= -\frac{1}{6}(2\Delta^* \tilde{f}_{i-1/2} + \Delta^* \tilde{f}_{i+1/2}), \\ (\Delta f)_i^+ &= \frac{1}{6}(\Delta^* \tilde{f}_{i-1/2} + 2\Delta^* \tilde{f}_{i+1/2}), \end{aligned} \quad (27)$$

where

$$\begin{aligned} \Delta^* \tilde{f}_{i-1/2} &= \text{minmod}(\Delta^* f_{i-1/2}, 4\Delta^* f_{i+1/2}), \\ \Delta^* \tilde{f}_{i+1/2} &= \text{minmod}(\Delta^* f_{i+1/2}, 4\Delta^* f_{i-1/2}), \end{aligned} \quad (28)$$

and

$$\begin{aligned} \Delta^* f_{i+1/2} &= \Delta f_{i+1/2} - \frac{1}{6}\Delta^3 \hat{f}_{i+1/2}, \\ \Delta^3 \hat{f}_{i+1/2} &= \Delta \hat{f}_{i-1/2} - 2\Delta \hat{f}_{i+1/2} + \Delta \hat{f}_{i+3/2}, \\ \Delta \hat{f}_{i-1/2} &= \text{minmod}(\Delta f_{i-1/2}, 2\Delta f_{i+1/2}, \Delta f_{i+3/2}), \\ \Delta \hat{f}_{i+1/2} &= \text{minmod}(\Delta f_{i+1/2}, 2\Delta f_{i+3/2}, \Delta f_{i-1/2}), \\ \Delta \hat{f}_{i+3/2} &= \text{minmod}(\Delta f_{i+3/2}, 2\Delta f_{i-1/2}, \Delta f_{i+1/2}), \end{aligned} \quad (29)$$

with $\Delta f_{i+1/2} = f_{i+1} - f_i$, and the minmod limiter is given by $\text{minmod}(\mathbf{y}) = \max[0, \min(\mathbf{y})] + \min[0, \max(\mathbf{y})]$.

In order to preserve the positivity of the distribution function, the PP limiter is imposed [53]

$$(\Delta f)_i^{\text{lim},\pm} = \theta \max((\Delta f)_i^\pm, -f_i), \quad (30)$$

where $\theta = \begin{cases} 1, & f_{\text{lim}} \leq 0, \\ \frac{f_i}{f_{\text{lim}}}, & f_{\text{lim}} > 0, \end{cases}$ with $f_{\text{lim}} = \max[(\Delta f)_i^-, -f_i] + \max[(\Delta f)_i^+, -f_i]$.

Then the final interface distribution function can be obtained by

$$f_{i,r} = f_i + (\Delta f)_i^{\text{lim},+}, \quad f_{i,l} = f_i + (\Delta f)_i^{\text{lim},-}. \quad (31)$$

And velocity interface gradient $\phi_{j,f}$ can be further obtained by

$$\phi_{j,f} = [f(\mathbf{x}_j, \mathbf{v}_f) - f(\mathbf{x}_j, \mathbf{v}_i)]/(\mathbf{v}_f - \mathbf{v}_i). \quad (32)$$

The physical interface distribution $\tilde{f}^+(\mathbf{x}_f, \mathbf{v}_i, t_k)$ and its gradient $\sigma_{f,i}$ can be calculated in a similar way. Thanks to the using of this compact MUSCL with PP limiter, current DUGKS preserves the favorable conservative property and positivity of distribution function in lengthy simulations, while the direct extension of the previous DUGKS [47,48] using the second-order interpolation scheme with van Leer slope limiter only well preserve the conservation of total energy within 100 plasma periods. This statement will be verified by the numerical test in Sec. IV A.

It is worth noting that, combined with above MUSCL method, similarly to original DUGKS, the Eulerian scheme in current DUGKS is used for distribution function evolution [the first equation in Eq. (10)] in physical space to easily handle general boundary conditions. However, the same treatment used for the evolution [the second equation in Eq. (10)] in velocity space would encounter expensive computation cost when the flow involving large electric forces, since the time step should meet the CFL condition, i.e., $\Delta t < \Delta \mathbf{v}/|\nabla\phi|_{\text{max}}$, where $\Delta \mathbf{v}$ and $|\nabla\phi|_{\text{max}}$ are velocity grid spacing and maximum electric force. Fortunately, different from the physical space, the homogeneous Dirichlet boundary condition can be generally used in the velocity space if appropriate discrete velocities are chosen. In such case, for the evolution in velocity space, the semi-Lagrangian scheme will be a better choice, which can be processed in two steps. First, we directly shift the distribution function in velocity space between cell centers following the characteristic line $f(\mathbf{x}_j, \mathbf{v}_i, t_{k+1}^*) = f(\mathbf{x}_j, \mathbf{v}_i - a^i \Delta t, t_b)$, where $a^i = \text{int}[\frac{\nabla\phi\Delta t}{\Delta \mathbf{v}}] \frac{\Delta \mathbf{v}}{\Delta t}$. Second, calculate the distribution at time t_{k+1} driven by the remain force a^r using above MUSCL reconstruction, i.e., $f(\mathbf{x}_j, \mathbf{v}_i, t_{k+1}) = f(\mathbf{x}_j, \mathbf{v}_i - a^r \Delta t, t_{k+1}^*)$, where $a^r = \nabla\phi - a^i$. The advantage of the semi-Lagrangian scheme presented here will be verified by the numerical test in Sec. IV D.

Thanks to semi-implicit coupling of particle transport and collision as well as the using of the semi-Lagrangian scheme in velocity space, the time step of current DUGKS is only restricted by CFL condition in physical space, i.e., $\Delta t = C\Delta \mathbf{x}_{\text{min}}/\mathbf{v}_{\text{max}}$, where C is the CFL number, $\Delta \mathbf{x}_{\text{min}}$ and \mathbf{v}_{max} are minimum grid spacing and maximum discrete velocity.

4. The iterative algorithm

To provide a more clear understanding of the proposed DUGKS and its implementation, in this section we present the iterative algorithm in details. For the sake of simplicity, some abbreviation for the distribution function f will be used in the following. For instance, $\tilde{f}_{x_f,i}^{k+1/2}$ stands for the distribution function \tilde{f} in physical interface \mathbf{x}_f and velocity cell i at $t = t_{k+1/2}$, which is equivalent to $\tilde{f}(\mathbf{x}_f, \mathbf{v}_i, t_{k+1/2})$. In summary, with the initialized $f_{j,i}^0$ in each cell j , the evolution procedure of present DUGKS from time t_k to t_{k+1} can be described as follows:

- (1) Reconstruct $\tilde{f}_{x_f,i}^{k+1/2}$ at the physical interface \mathbf{x}_f .
 - (a) calculate cell distribution function $\tilde{f}_{j,i}^{+,k}$ according to Eq. (21);
 - (b) reconstruct physical interface distribution function $\tilde{f}_{x_f,i}^{+,k}$ from $\tilde{f}_{j,i}^{+,k}$ according to Sec. III A 3;
 - (c) calculate physical interface distribution function $\tilde{f}_{x_f,i}^{k+1/2}$ according to Eqs. (20) and (22);
- (2) Calculate physical microflux $\mathbf{F}_{j,i}^{k+1/2}$ at time $t_{k+1/2}$.
 - (a) compute interface conversional variables $\mathbf{W}_{x_f}^{k+1/2}$ from Eq. (23);
 - (b) calculate original physical interface distribution function $f_{x_f,i}^{k+1/2}$ according to Eq. (24);
 - (c) obtain the physical microflux $\mathbf{F}_{j,i}^{k+1/2}$ from Eq. (13).

(3) Update conservative variables \mathbf{W}_j^b and cell distribution $f_{j,i}^b$ according to Eqs. (15) and (16).

(4) Calculate electric potential ϕ_j^b by solving Poisson equation Eq. (17).

(5) Compute velocity microflux $\mathbf{G}_{j,i}^{b+1/2}$ at time $t_{b+1/2}$.

(a) reconstruct velocity interface distribution function f_{j,v_f}^b from $f_{j,i}^b$ according to Sec. III A 3;

(b) calculate velocity interface distribution function $f_{j,v_f}^{b+1/2}$ from Eq. (25);

(c) obtain the velocity microflux $\mathbf{G}_{j,i}^{b+1/2}$ from Eq. (14).

(6) Update cell distribution function $f_{j,i}^{k+1}$ according to Eq. (12).

If particle collision is not considered, then the computation of physical microflux in steps (1) and (2) can be much simplified, which is similar to that of velocity microflux in step (5).

B. Particle-in-cell method

The PIC presented in this section is based on Refs. [6,54]. First, the update rule of classical collisionless PIC method is reviewed. Then the particle loading and particle collisions are addressed, since they are the main sources of numerical noise. Finally, the iterative algorithm of collisional PIC method is presented.

1. Update rule

In the PIC method, the distribution function $f(\mathbf{x}, \mathbf{v}, t)$ is discretized into a sum of delta measures in the phase space, which can be expressed as

$$f(\mathbf{x}, \mathbf{v}, t) = \sum_p w_p \delta(\mathbf{x} - \mathbf{x}_p) \delta(\mathbf{v} - \mathbf{v}_p), \quad (33)$$

where w_p is the particle weight. As a result, one should firstly load the particles located at $(\mathbf{x}_p, \mathbf{v}_p)$ to approximate the prescribed distribution $f(\mathbf{x}, \mathbf{v}, t)$, which will be introduced in next section.

For the classical PIC method without collision term, i.e. $\tau \rightarrow \infty$, the integration of Eq. (8) in the Lagrangian framework using “leap-frog” scheme is done as follows [6]:

$$\mathbf{x}_p^{k+1} = \mathbf{x}_p^k + \Delta t \mathbf{v}_p^{k+1/2}. \quad (34)$$

$$\mathbf{v}_p^{k+3/2} = \mathbf{v}_p^{k+1/2} + \Delta t \mathbf{E}_p^{k+1}. \quad (35)$$

Then the conservative variables \mathbf{W}_j^{k+1} in each physical cell j can be computed with

$$\mathbf{W}_j^{k+1} = \sum_p \psi w_p S(\mathbf{x}_j - \mathbf{x}_p^{k+1}), \quad (36)$$

where $S(\mathbf{x}_j - \mathbf{x}_p)$ is the shape function. In this paper, the first-order spline function is used, which reads

$$S(\mathbf{x}_j - \mathbf{x}_p) = \begin{cases} 1 - (\mathbf{x}_j - \mathbf{x}_p)/\Delta \mathbf{x}_j, & |\mathbf{x}_j - \mathbf{x}_p| < \Delta \mathbf{x}_j, \\ 0, & \text{otherwise.} \end{cases}$$

Then the electrical potential can be solved from the Poisson equation

$$\lambda^2 \Delta \phi_j^{k+1} = n_j^{k+1} - 1. \quad (37)$$

And the electrical field \mathbf{E}_j^{k+1} in each physical cell can be computed with $\mathbf{E}_j^{k+1} = \nabla \phi_j^{k+1}$. If \mathbf{E}_j is known, then the field acting on each particle can be computed as

$$\mathbf{E}_p^{k+1} = \sum_j \mathbf{E}_j^{k+1} S(\mathbf{x}_j - \mathbf{x}_p^{k+1}). \quad (38)$$

Once the initial particles $(\mathbf{x}_p^k, \mathbf{v}_p^k)$ at time $t = t_k$ is prescribed, the process [from Eq. (34) to Eq. (38)] is ceaselessly repeated until simulation time is up or the simulation particles keep nearly constant.

2. Particle loading and collisions

It is well known that the PIC method suffers from numerical noise, which mainly results from the randomization process in particle loading and particle collision. Generally, particle loading is done through random numbers, namely “noise start” (NS) in this paper, where the macroparticle velocities are randomized about their mean beamlet velocity [6]. The general alternative is to apply the acceptance-rejection method [55], which can be used to load any prescribed distribution function f_v . But this method involves certain numerical noise which may overwhelm the initial perturbation to be studied, which will be verified by the numerical test in Sec. IV B.

An alternative loading way is the “quiet start” (QS) method, where the prescribed distribution f_v is approximated by initially prescribing several monoenergetic beams of macroparticles without the using of random numbers [6]. In this paper, many equally weighted beams unevenly spaced in v are used to approximate f_v , which can be explained as follows. First, function f_v should be normalized to meet $\int_{-\infty}^{\infty} f_v dv = 1$. Then the integration of f_v in the velocity space is evenly divided into N_{pc} sections, where N_{pc} is the total particle number in each cell. Finally, solve above integration function and prescribe each particle velocity v . For instance, when prescribing the slowest particle v_1 , one should solve the equation $\int_{v_{min}}^{v_1} f_v dv = 1/N_{pc}$. Then the slowest particle is approximated by $v_1 = (v_x + v_{min})/2$. In a similar way, one can prescribe each particle. Encouragingly, QS significantly reduces the numerical noise with a modern particle numbers but involves multiple-beam-instability, which will be presented by the numerical test in Sec. IV B.

When the particle collision is considered, the integral solution of the third equation in Eq. (8) can be expressed as

$$f^{k+1} = f^k \exp(-\Delta t/\tau) + f^{eq,k} [1 - \exp(-\Delta t/\tau)]. \quad (39)$$

Equation (39) means that the probability that any given molecule in the cell undergoes collision is $P_j = 1 - \exp(-\Delta t/\tau)$. Note that the collision only change particle velocity \mathbf{v}_p , and this procedure is shown as follows.

Calculate collision probability P_j in each cell and assign a random number R_f to represent each particle. If this random R_f is less than the probability P_j , then this particle is randomly assigned a new velocity drawn from the Maxwellian distribution Eq. (2). After all particles in the cell have been processed in this way, one can obtain the provisional postcollision velocities \mathbf{v}_p^* , average velocity \mathbf{u}_j^* , and average temperature T_j^* of the particles. To enforce conservation of momentum and energy at the cell level, the provisional postcollision velocities

should be adjusted, and the final postcollision velocity \mathbf{v}_p of each particle is determined by [54]

$$\mathbf{v}_p = \mathbf{u}_j + (\mathbf{v}_p^* - \mathbf{u}_j^*) \sqrt{T_j/T_j^*}, \quad (40)$$

where \mathbf{u}_j and T_j are the average velocity and temperature of precollision particles. It should be noted that, due to the splitting treatment of particle collision and particle transport, the numerical time step in current PIC method should be less than the particle collision time, which will be presented by the numerical test in Sec. IV C.

3. The iterative algorithm

To have a better understanding of the collisional PIC, we present this iterative algorithm in this section. With the initial loading of the particle velocity \mathbf{v}_p^0 and position \mathbf{x}_p^0 in each cell j , the evolution procedure of current collisional PIC from time t_k to t_{k+1} can be briefly shown as follows:

- (1) Update particle position \mathbf{x}_p^{k+1} according to Eq. (34).
- (2) Compute conservative variables \mathbf{W}_j^{k+1} in each cell j from Eq. (36).
- (3) Solve Poisson equation Eq. (37), and then obtain the electrical field \mathbf{E}_j^{k+1} in each cell j .
- (4) Update particle velocity $\mathbf{v}_p^{k+3/2}$ according to Eq. (35), where \mathbf{E}_p^{k+1} is obtained from Eq. (38).
- (5) Handle the particle collision and update the final particle velocity $\mathbf{v}_p^{k+3/2}$ according to Eq. (40).

It should be noted that the collision process in current PIC is much efficient than that in the traditional direct simulation Monte Carlo (DSMC) method [55], but it is still time-consuming. Thus, it is better to skip step (5) if the particle collision is not taken into consideration.

IV. NUMERICAL EXPERIMENTS

In this section, four numerical experiments, e.g., plasma sheath, linear Landau damping, collisional nonlinear Landau damping, and plasma ion acceleration, are implemented to validate present DUGKS. We also compare the performance of current DUGKS with the PIC method. For all simulations, we consider homogeneous Dirichlet boundary conditions for the distribution function of DUGKS in velocity space and Poisson equation which is solved by the finite-element method [56]. For a fair comparison, the same physical grid N_x and numerical time step $\Delta t = C \Delta x_{\min}/v_{\max}$ are used. The velocity space is discretized by discrete velocity points N_v with equal weight $w_i = \Delta v$ in DUGKS, while discretized by particle numbers in each cell N_{pc} with equal weight $w_p = 1/N_{pc}$ in PIC unless otherwise stated.

A. Plasma sheath

Now the present DUGKS is applied to simulate plasma sheath for two main purposes. One is to investigate whether it can predict reasonable results for this nonequilibrium flow with nonperiodic boundary conditions, and the other is to compare its performance with direct extension of previous DUGKS (DE-DUGKS) [47,48] in lengthy simulations. To accelerate the convergence, we only consider the electron

dynamic in the collisionless regime. Here the PIC results are chosen as benchmark solutions.

In our simulation, dimensional domain length L , electron number density, and temperature are 0.1 m, 10^{13} m^{-3} , and 11 604 K, and their reference values are set to be $2.35 \times 10^{-3} \text{ m}$ (Debye length), 10^{13} m^{-3} , and 11 604 K, respectively. The initial condition is a standard Maxwell distribution. The velocity space is set to be $[-v_{\max}, v_{\max}]$ with $v_{\max} = 8$, which is discretized by $N_v = 256$ and $N_{pc} = 500$. The space grid $N_x = 256$ and CFL number $C = 0.6$. Besides, the zero-inflow boundary conditions are imposed on physical space [40,57]. Here we run this problem up to $t = 1000$, i.e., 1000 plasma periods.

Figure 1 shows phase-space plot of distribution predicted by present DUGKS and PIC. Note that the DE-DUGKS predicates similar results with the present DUGKS before $t = 100$, and thus only the present DUGKS results are presented here. As shown in the figure, two methods present identical results. At the beginning ($t = 5$), the electrons fast escape and leave a net positive charge near the wall. This gives rise to a potential barrier, i.e., plasma sheath, at the walls, which prevents remaining electrons to escape unless they have a large-enough velocity to overcome this sheath potential. Eventually electrons are confined to interior of domain, and the system reaches a dynamic steady state (from $t = 30$ to $t = 45$). In Fig. 2, we present plasma density n and electric potential ϕ against space x at $t = 140$. Clearly, the current DUGKS provides consistent results with PIC, while DE-DUGKS predicts a higher potential ϕ .

The disparities between current DUGKS and DE-DUGKS can be more clearly seen in lengthy simulations. Here the system L_1 norm $L_1 = \iint f dx dv$ and total energy $E_T = E_p + E_K$, where $E_p = \frac{1}{2} \int E^2 dx$ is potential energy and $E_K = \frac{1}{2} \iint f v^2 dx dv$ is kinetic energy, are used to diagnose the kinetic solver. As shown in Fig. 3, DE-DUGKS well preserves the E_K , but it could not preserve the L_1 norm [see Fig. 3(a)] and E_T [see Fig. 3(b)] after $t = 100$. At $t = 1000$, the system L_1 norm (mass) predicted by DE-DUGKS reduces about 2% due to the numerical dissipation and nonpositive distribution function. From the Poisson equation, one can obtain that E_p would increase with the decrease of mass, which further result in the increase of the E_T (about 9%). Encouragingly, thanks to the using of the MUSCL scheme with PP limiter in Sec. III A 3, present DUGKS, similarly to PIC, predicts fairly good conservation both on L_1 norm and energy. Besides, we also present time evolution of electric potential at the center of domain in Fig. 3(c) and its Fourier spectrum in Fig. 3(d). The plasma oscillation is well captured in all methods, and numerical oscillation frequency is consistent with theoretical value $\omega_p = 1$. However, the electric potential predicted by DE-DUGKS gradually increases, which more clearly reflects its bad conservative property in lengthy simulations.

The above arguments indicate that in comparison with the DE-DUGKS, present DUGKS has a better performance on the conservation of system mass and total energy in long-time simulations. For this reason, in the following paper, we only present the results predicted by present DUGKS. To further verify the capability of current DUGKS, several numerical experiments and a comparative study with PIC are presented in following sections.

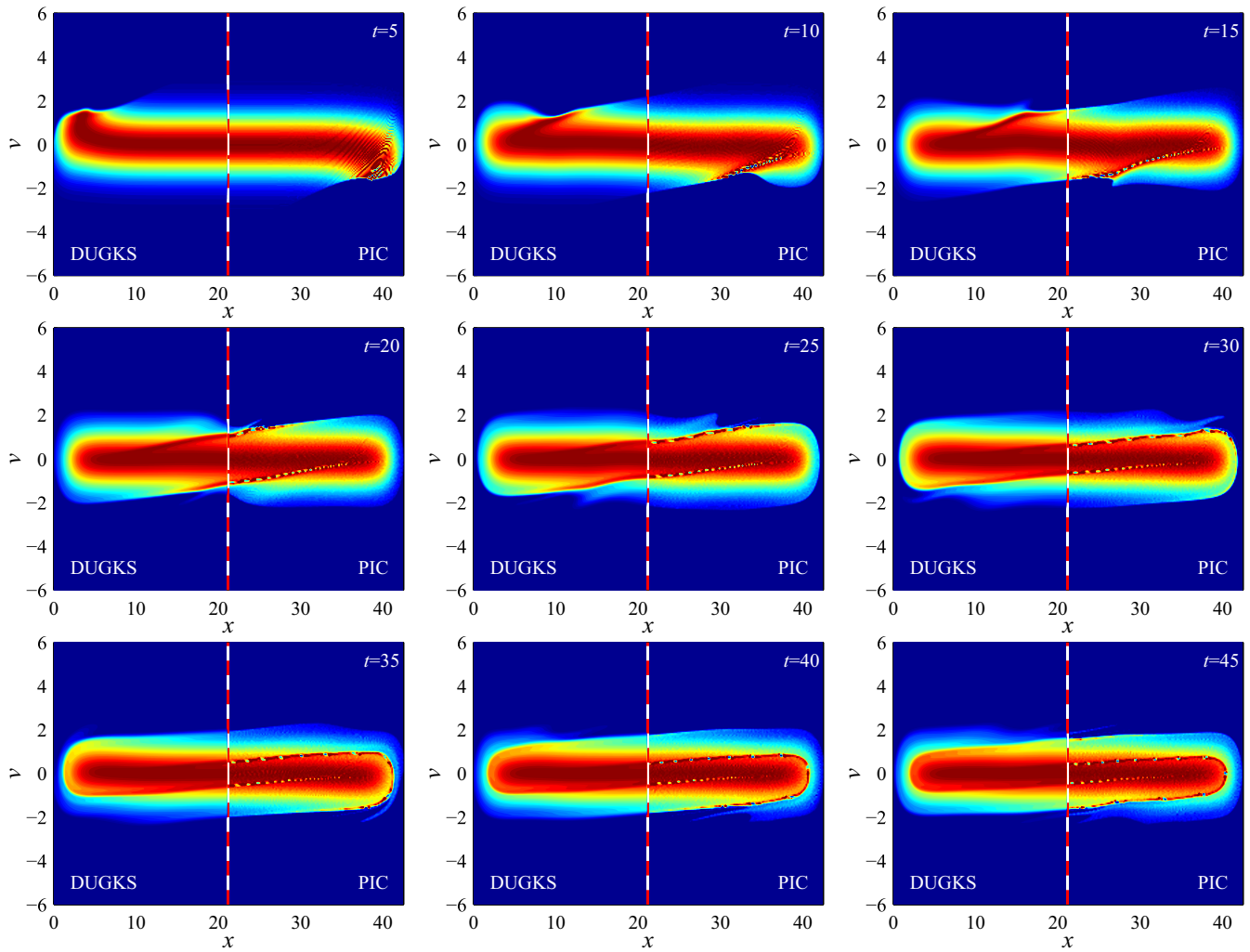


FIG. 1. Plasma sheath, phase-space plot of distribution function f at different time predicted by DUGKS and PIC: $N_x = 256$, $N_v = 256$, and $N_{pc} = 500$.

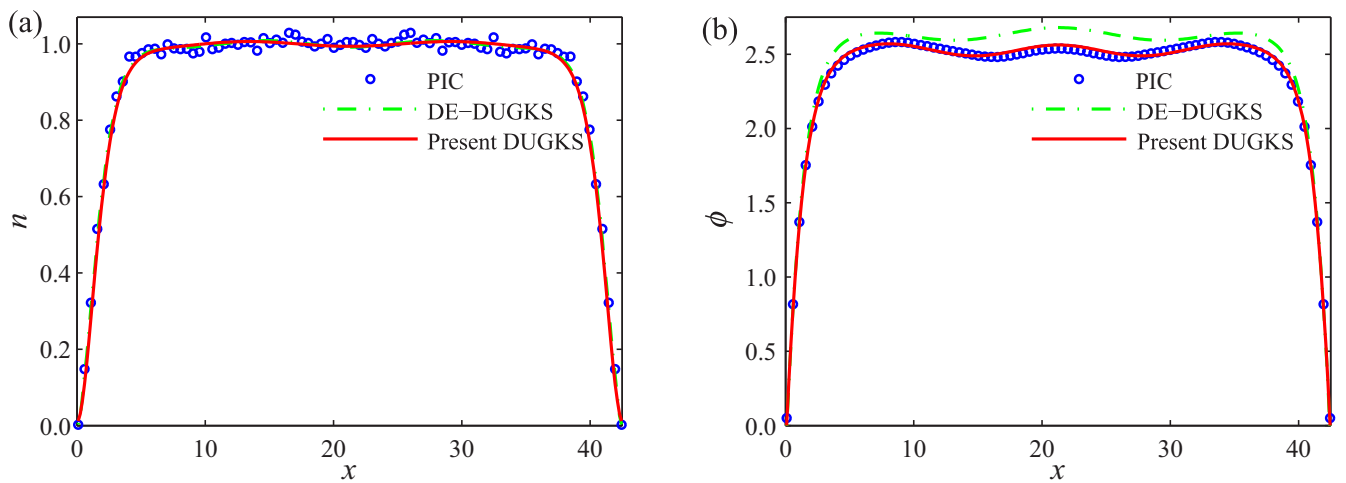


FIG. 2. Plasma sheath, macroscopic variables profile at $t = 140$: $N_x = 256$, $N_v = 256$, and $N_{pc} = 500$. The plasma density n (a) and electric potential ϕ (b) against the physical space x .

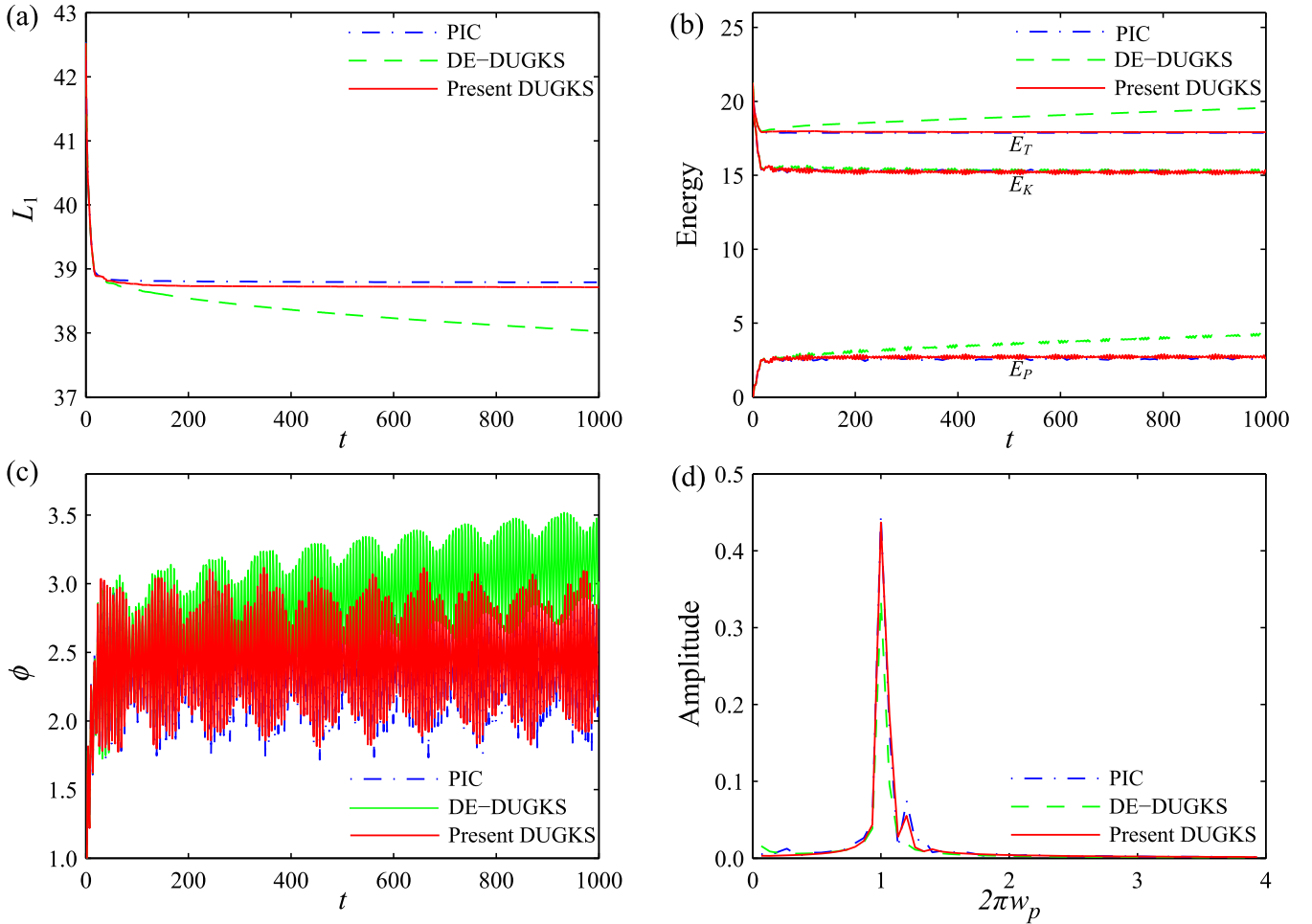


FIG. 3. Plasma sheath: $N_x = 256$, $N_v = 256$, and $N_{pc} = 500$. Time evolution of the L_1 norm (a) and energy (b), including total energy (E_T), potential energy (E_P), and kinetic energy (E_K). Time evolution of electric potential ϕ at $x = L/2$ (c) and its Fourier spectrum (d).

B. Linear Landau damping

Now the classical linear Landau damping is presented to investigate the performance of DUGKS and PIC for plasma flow with a weak electrostatic perturbation. The initial condition is

$$f^0(x, v) = \frac{1}{\sqrt{2\pi}} [1 + \alpha \cos(kx)] \exp(-v^2/2), \quad (41)$$

with $\alpha = 0.01$ and $k = 0.5$. In our simulation, the phase-space domain is set to be $[0, L] \times [-v_{\max}, v_{\max}]$, where $L = 2\pi/k$ and $v_{\max} = 5$. The physical grid $N_x = 128$ and the CFL number $C = 0.9$. The time evolution of electrostatic energy $|E|_{L_2}$ is used in diagnosing our proposed scheme, given by $|E|_{L_2} = \frac{\lambda^2}{2} \int E^2 dx$, where normalized Debye length $\lambda = 1$. Besides, the periodic boundary conditions are used for the physical space. Here we run this problem up to $t = 35$.

First, the importance of initial particle loading in PIC is addressed. Figure 4 shows the comparison between NS and QS. It can be seen that the distribution function predicted by both QS and NS agree well with the theoretical distribution. However, for NS, there are still some fluctuations near $v = 0$ due to the random noise [see Fig. 4(a)]. Although these deviations are slight, they strongly affect the predicted results

in a long-time simulation. With NS, Landau damping only can well be observed before $t = 18$ [see Fig. 4(b)]. Encouragingly, with QS, one can observe the electrostatic energy decay over a longer time, in which the damping rate is consistent with theoretical one $\gamma = -0.15$ [1,34]. This comparison denotes that QS is desired if initial electrostatic perturbation is weak. Therefore, in the following part, the QS is used to load particles unless otherwise stated.

Then we present the performance of DUGKS and PIC with against different discrete velocity points. Figure 5 shows the time evolution of electrostatic energy $|E|_{L_2}$ predicted by DUGKS and PIC. It can be seen that the energy variation in DUGKS can be well captured even with a low-velocity resolution, i.e., $N_v = 64$. In contrast, the energy variation in the PIC highly depends on the number of particles in each cell, ranging from 1000 to 128 000.

To further demonstrate their different performance, in Fig. 6 we present their computational error and efficacy. The error is calculated as L_1 norm of the difference between numerical electric field solution and reference one (PIC result with $N_{pc} = 256000$). The efficacy is defined as inverse of the product of error and CPU time, which is a useful indicator of cost-effectiveness of the algorithm [12]. Clearly, an algorithm performs well if the efficacy increases notably with increasing

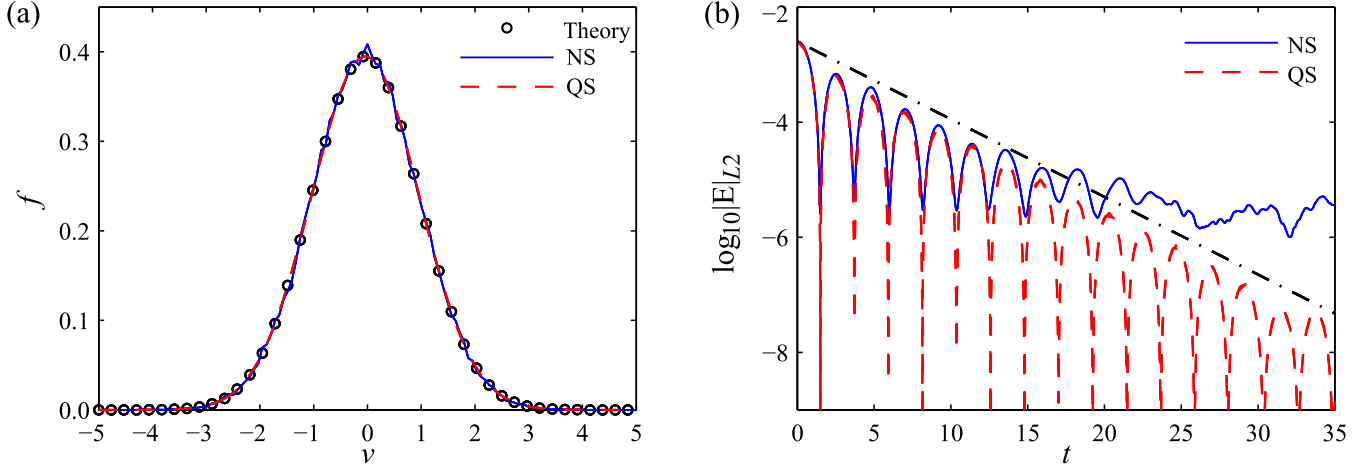


FIG. 4. Linear Landau damping, the comparisons of noise start (NS) and quiet start (QS): $N_x = 128$ $N_{pc} = 128000$. The initial distribution function f against velocity space v at $x = L/2$ (a). Time evolution of the electrostatic energy; the black dot dash line indicates theoretical damping rate $\gamma = -0.15$ (b).

time. As shown in Fig. 6(a), here DUGKS is more accuracy than PIC. Their different performances can be more clearly reflected in the efficacy. From Fig. 6(b), it can be seen that the PIC efficacy scales as the inverse of square root of CPU time (from 6 s to 576 s), while the DUGKS efficacy increase exponentially with the increase of CPU time (from 2.1 s to 7.4 s). Hence, in the view of pure cost-effectiveness, it is not advantageous to increase particles numbers in PIC to reduce the error. For example, to reach an error of 10^{-8} , the DUGKS takes about 2.4 s while PIC takes about 287 s, i.e., more than 100 times longer.

It is worth noting that PIC with QS can significantly reduce numerical noise, but it suffers from artificial multibeam instability [6,29]. Figure 7 shows evolution of energy predicted by PIC and DUGKS against different initial perturbation α . Clearly, numerical instability arises in PIC with the decrease of initial perturbation amplitude. To see long-time Landau damping, the required particle number should increase as α decreases. Encouragingly, DUGKS does not involve this

instability and can exhibit long-time Landau damping for a wide range of perturbation amplitude.

The above arguments indicate that the PIC suffers from expensive cost due to intrinsic statistical noise, while the noise-free DUGKS can achieve more accuracy solutions at a fraction of computational costs with respect to PIC. As a result, the DUGKS is superior to PIC for the study of warm plasma dynamics with a small perturbation amplitude.

C. Collisional nonlinear Landau damping

Now nonlinear Landau damping is presented to investigate the performance of DUGKS and PIC for plasma flow in different collisional regimes. The initial condition is same as Eq. (41) but with $\alpha = 0.5$ and $k = 0.5$. In our simulation, phase-space domain is set to be $[0, L] \times [-v_{\max}, v_{\max}]$, where $L = 2\pi/k$ and $v_{\max} = 8$. The space grid $N_x = 128$ and CFL number $C = 0.6$. Besides, the Debye length λ and boundary conditions are same as those of linear Landau damping in Sec. IV B. Here three flow regimes with $\text{Kn} = \infty, 1$, and 10^{-3} ,

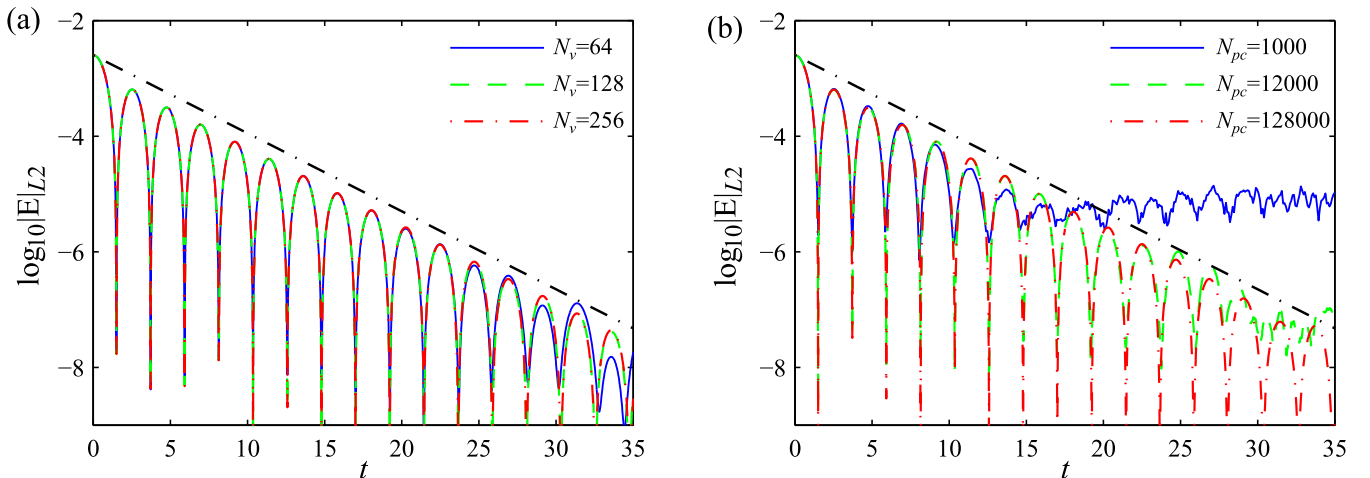


FIG. 5. Linear Landau damping; the comparisons of DUGKS and PIC with $N_x = 128$ against different discrete velocity points. Time evolution of electrostatic energy predicted by DUGKS (a) and PIC (b). The black dot dash line indicates theoretical Landau damping rate $\gamma = -0.15$.

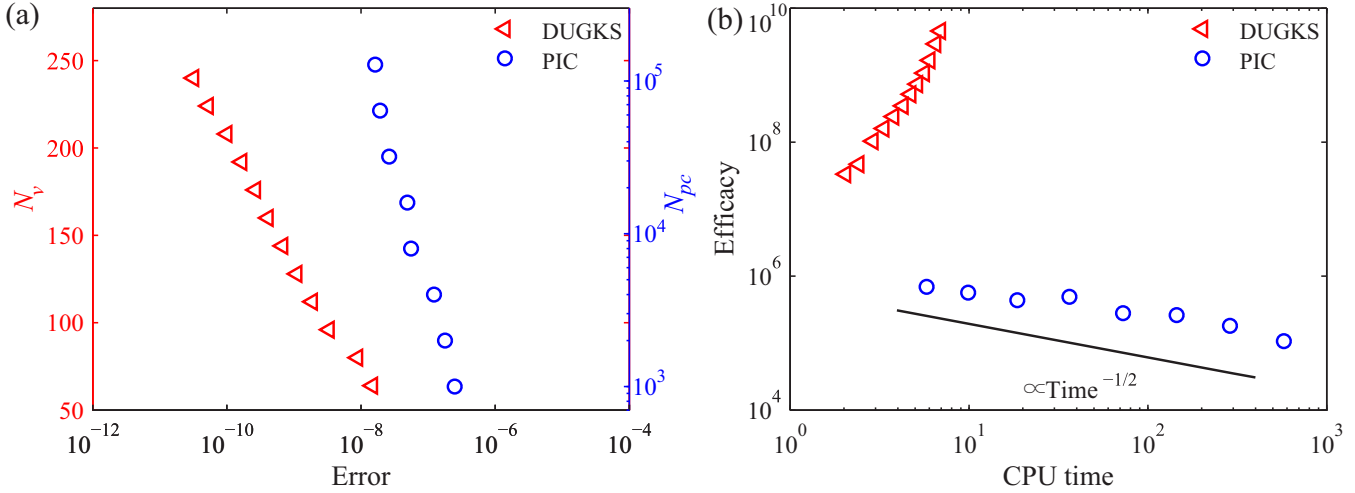


FIG. 6. Linear Landau damping; the comparisons of DUGKS and PIC with $N_x = 128$ against different discrete velocity points. The error against the discrete velocity points (a). The efficacy against the CPU time costs (in second) (b). The black line indicates the scaling $\text{time}^{-1/2}$.

which correspond to typical collisionless, mild collision, and strong collision regimes, are investigated. Note that in our simulation infinite value ∞ is replaced by a big one 10^{15} . For all simulations, we set N_v and N_{pc} to be 256 and 5000, except $N_{pc} = 1000$ for $\text{Kn} = 10^{-3}$.

Figure 8 shows the time evolution of electrostatic energy $|E|_{L2}$ with different Kn . As shown in Fig. 8(a), DUGKS and PIC provide identical results in the collisionless regime. The decay rate ($\gamma_1 = -0.287$) and growth rate ($\gamma_2 = 0.078$) of electric energy predicted by two methods are consistent with results in Refs. [18,21,46]. In the presence of mild collision [see Fig. 8(b)], DUGKS and PIC provide similar results. However, in the strong collision case Fig. 8(c), obvious deviations between two methods can be observed. Similarly to the previous study [58], the energy predicted by DUGKS oscillates around a constant value. But the energy predicted by PIC is damping, because the valid physical process, which is consistent with splitting treatment of particle transport and collision, is that physical grid size and time step used in PIC

should be less than particle mean free path and collision time. As expected, in Fig. 8(d) the energy converges to the expected one when physical grid N_x increases to 1024, where $\Delta t = 9.2 \times 10^{-4} < \text{Kn}$. Thanks to the coupling of particle transport and collision, the energy predicted by DUGKS with $N_x = 128$ can compare favorably with that of PIC with $N_x = 1024$.

To further verify the capability of present DUGKS, in Fig. 9 we present phase-space plot of distribution f . It can be clear seen that two methods present almost same results. At the beginning, the distribution f evolves in a similar way for the three cases. However, with the increase of simulation time, obvious differences can be observed. In the collisionless case [Fig. 9(a)], the filamentation process is clearly captured, which has been investigated by various methods [22,46]. In presence of mild collisions [see Fig. 9(b)], the distribution f becomes smoother. However, in the strong collision case Fig. 9(c), some fluctuations occur due to strong oscillating electric field [see Fig. 8(c)]. It is also worth noting that, although the numerical noise in PIC can be greatly reduced with

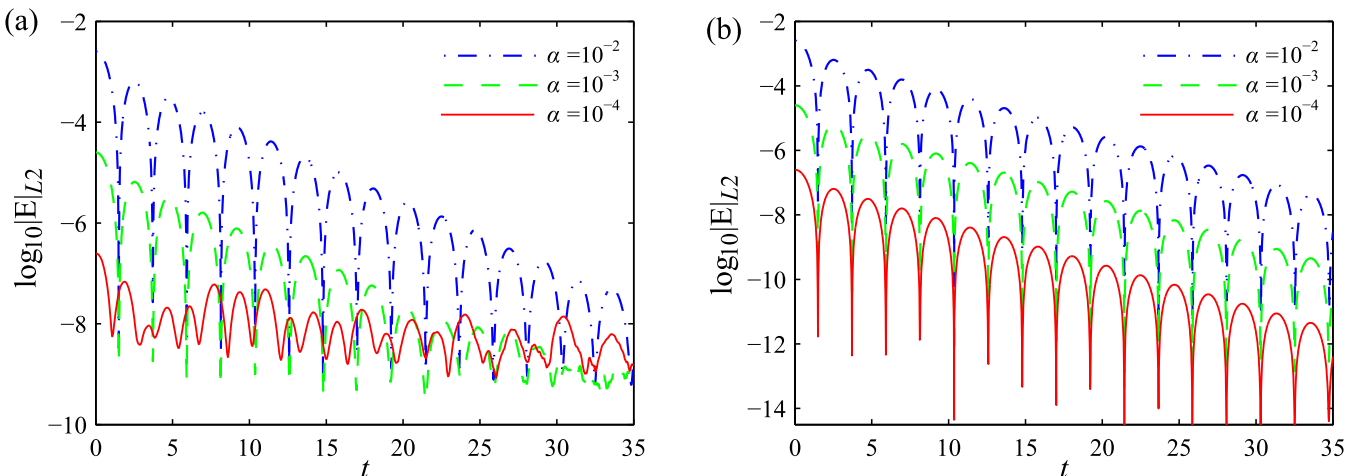


FIG. 7. Linear Landau damping: $N_x = 128$, $N_v = 256$, and $N_{pc} = 128\,000$. Time evolution of electrostatic energy with different perturbation α predicted by PIC (a) and DUGKS (b).

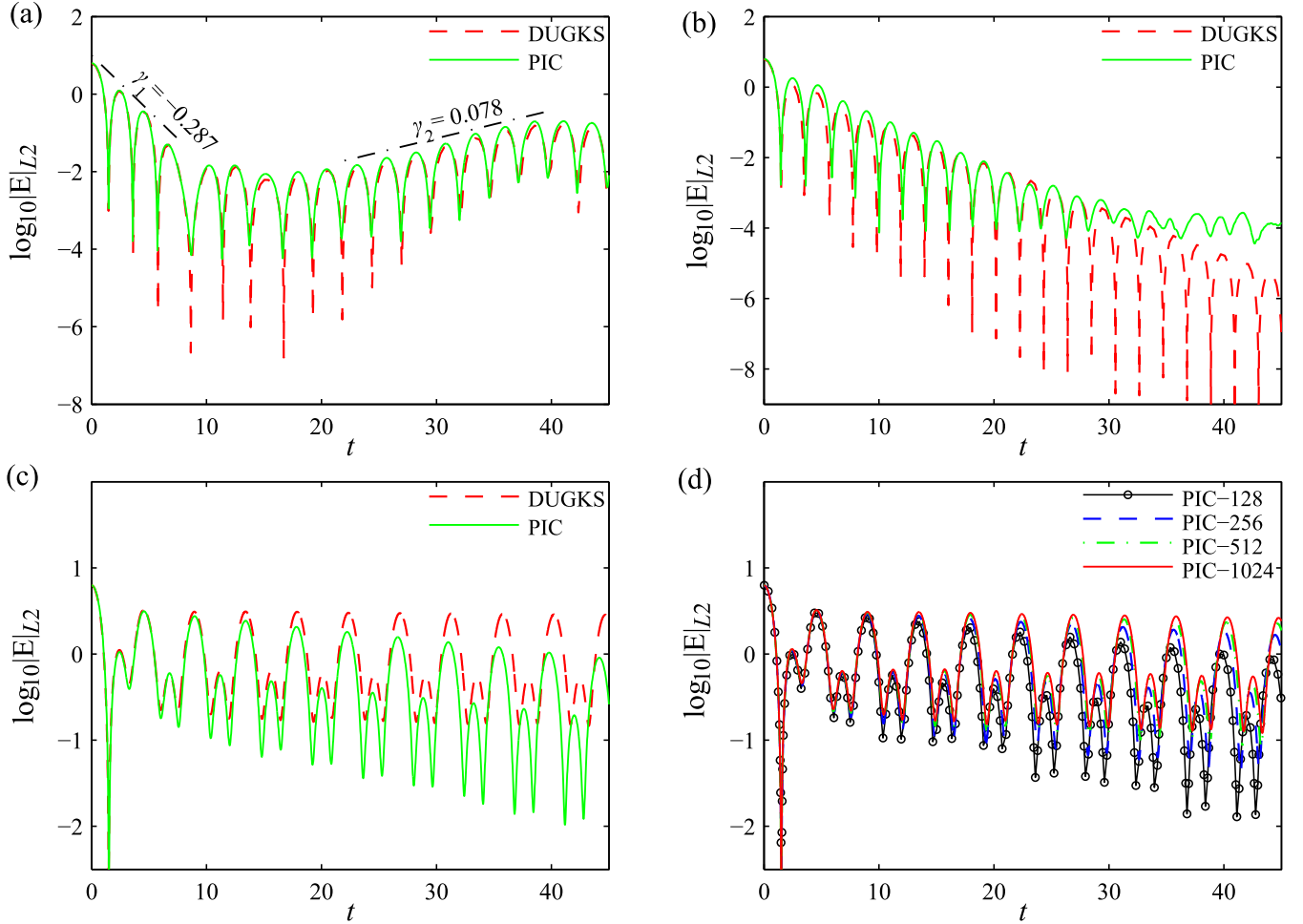


FIG. 8. Nonlinear Landau damping. The time evolution of electrostatic energy predicted by DUGKS and PIC for $\text{Kn} = \infty$ (a), $\text{Kn} = 1$ (b), and $\text{Kn} = 10^{-3}$ (c). The time evolution of electrostatic energy predicted by PIC using different physical grids for $\text{Kn} = 10^{-3}$ (d).

QS [see Fig. 9(a)], it is still a pending issue when collision is involved [see Figs. 9(b) and 9(c)]. Encouragingly, as a deterministic method, the DUGKS is free of numerical noise.

Finally, we make some discussions about the computational costs. Table I shows the CPU time of DUGKS and PIC. Note that the value within brackets is the result without consideration of particle collision. Clearly, the collision process is time-consuming in both methods, especially the PIC method. Besides, the computational costs in PIC increase with the decrease of Kn , while the cost of DUGKS for all simulations is nearly same. Since numerical time step in PIC is limited by collision time, the computation cost of PIC is quite expensive

in the strong collision regime. As shown in Table I, the CPU time for $\text{Kn} = 10^{-3}$ of PIC and DUGKS are 7736 s and 36 s, i.e., more than 200 times longer.

The above arguments indicate that the time step in PIC is limited by the particle collision time if the accuracy solution is expected, while the proposed DUGKS removes this limitation due to the semi-implicit unsplitting treatment of particle collision and transport. As a result, the proposed DUGKS is superior to PIC for the study of warm plasma dynamics in the strong collision regime.

D. Plasma ion acceleration

So far, all of the presented numerical experiments are particles-dominant plasma flow, in which the plasma dynamic is mainly determined by its self-consistent electric field. In this section, the field-dominant plasma is presented to investigate the performance of DUGKS and PIC for plasma flow involving a wide range of velocities, in which a large external electric field is imposed and the effect of self-consistent electric field can be neglected. Plasma ion acceleration is a good example, which is widely used in ion thrust. For convenience but without loss of generality, we focus on the ion dynamic

TABLE I. The total CPU time costs (in second) of DUGKS and PIC for nonlinear Landau damping with different Kn : $N_x = 128$, $N_v = 256$, and $N_{\text{pc}} = 5000$, except $N_x = 1024$ and $N_{\text{pc}} = 1000$ are used in PIC for $\text{Kn} = 10^{-3}$. Note that the value within brackets is the result without collision term.

Kn	∞	1	10^{-3}
DUGKS	37(21)	36	36
PIC	251(78)	420	7736

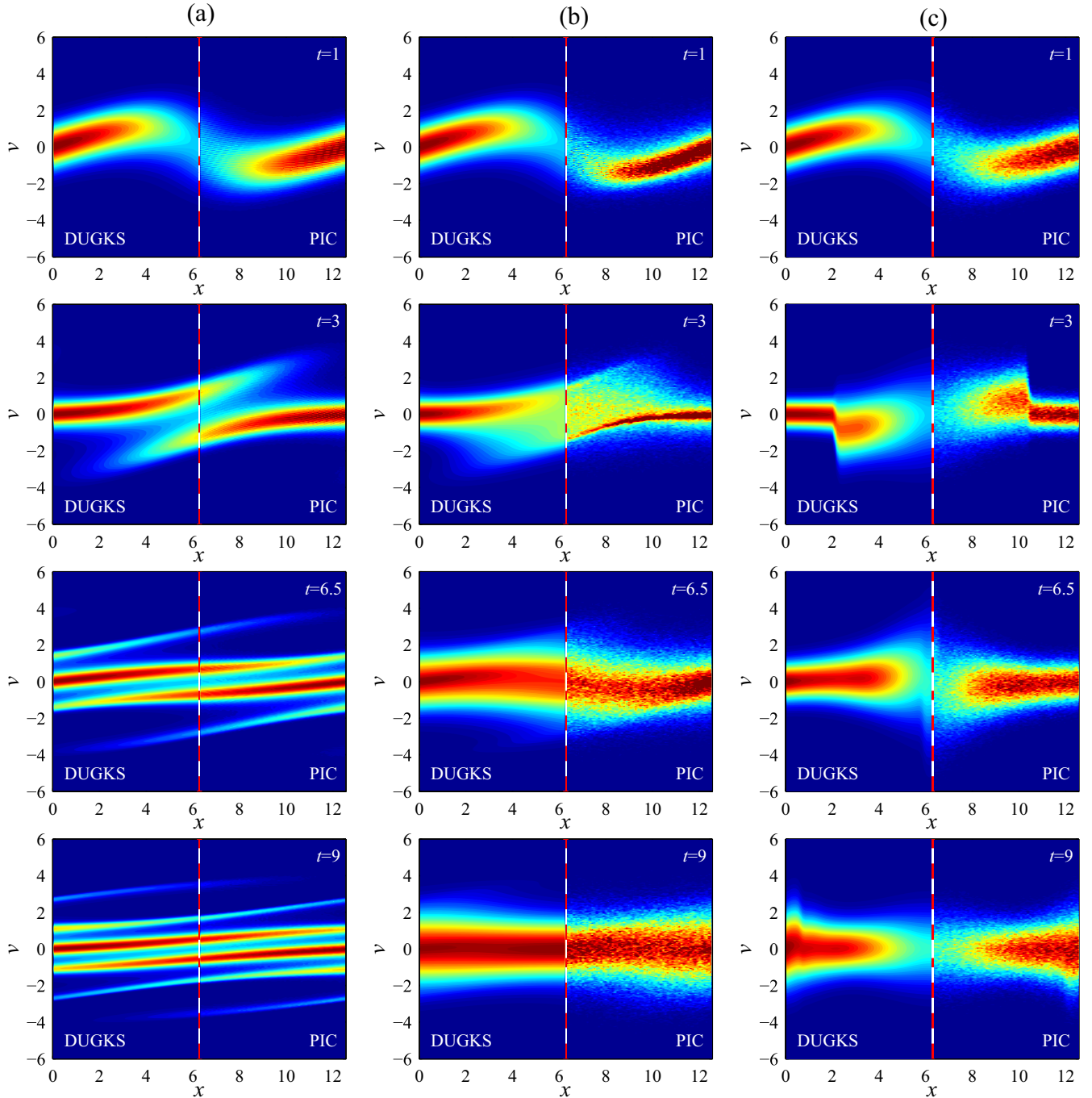


FIG. 9. Nonlinear Landau damping; phase-space evolution of distribution function at $\text{Kn} = \infty$ (a), $\text{Kn} = 1$ (b), and $\text{Kn} = 10^{-3}$ (c). Note that $N_x = 1024$ is used in PIC for $\text{Kn} = 10^{-3}$.

in the collisionless regime and simplify this problem as follows.

Initially, there are no ions in the domain $[0, L]$, where $L = 0.035$ m. The ions enter the domain at the anode $x = 0$ with a biased Maxwellian distribution, which is given by [59]

$$f(x=0, v_{\text{in}}) = c_0 n \beta^2 v_{\text{in}} \exp[-\beta^2 (v_{\text{in}} - u_d)^2], \quad v_{\text{in}} > 0, \quad (42)$$

where the inlet number density is $n = 10^{15} \text{ m}^{-3}$ and $\beta^2 = m_i / (2k_B T)$. The propellant is xenon ions, i.e., $m_i = 131$

amu, with temperature $T = 1500$ K. Note that the coefficient c_0 is used to normalize number density and given by $c_0 = 2 / (a_0 + b_0)$, where $a_0 = \exp(-\beta^2 u_d^2)$ and $b_0 = \sqrt{\pi} \beta u_0 [1 + \text{erf}(\beta u_d)]$. Thus the inject velocity at anode is $u_{\text{in}} = b_0 / [2\beta^2 u_d (a_0 + b_0)]$. Besides, the external electric potential is given by $\phi = \phi_d [1 - (x/L)^6]$ with anode potential $\phi_d = 100$ V. It is worth noting that the analytic solution for ion distribution function at an arbitrary position x in the domain can be obtained from the law of energy conservation, i.e., $v_{\text{in}}^2 = v^2 + 2\delta\phi$, where $\delta\phi = \phi - \phi_d$, which further can

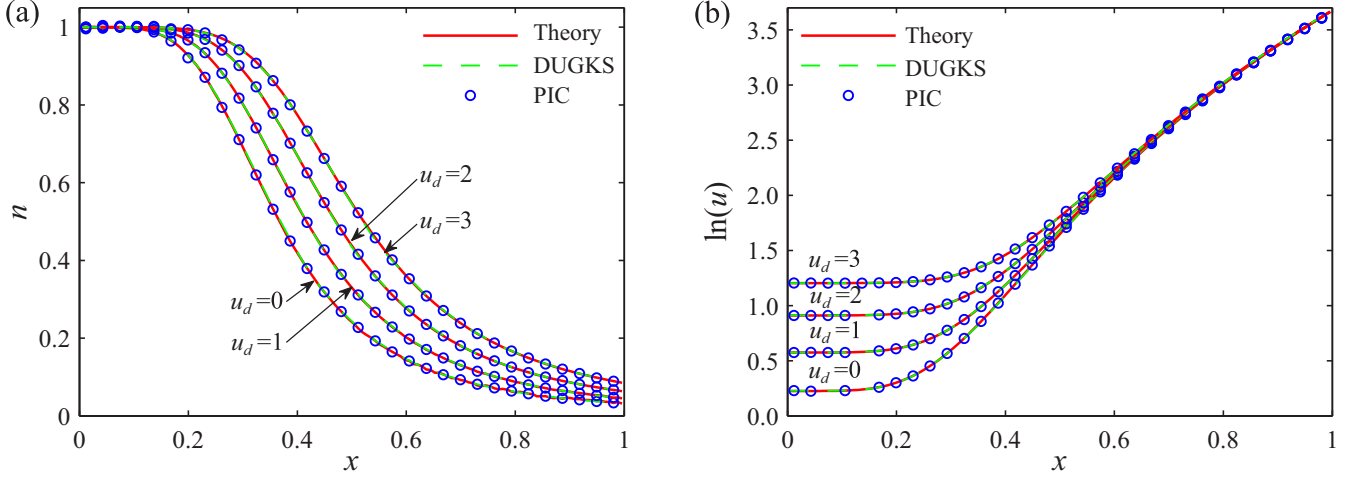


FIG. 10. Plasma ion acceleration, the comparison of DUGKS and PIC against different drift velocity: $N_x = 128$, $N_v = 256$, and $N_{pc} = 256$. Plasma density n (a) and velocity u (b) against the physical space x .

be written as

$$f(x, v) = c_0 n \beta^2 \sqrt{v^2 + 2\delta\phi} \exp[-\beta^2(\sqrt{v^2 + 2\delta\phi} - u_d)^2], \quad (43)$$

$$v > \sqrt{-2\delta\phi}.$$

In our simulation, the reference length, temperature, density, and mass are set to be $x_0 = L$, $T_0 = T$, $n_0 = n$, and $m_0 = m_i$, respectively. Following normalized process in Sec. II, one can further obtain reference potential $\phi_0 = 0.13$ V and $\beta = 1/2$. As a result, here the dimensionless simulation domain is $[0, 1]$ and the anode potential is 773. Besides, the velocity space is set to be $[0, v_{\max}]$ with $v_{\max} = 45$ and is discretized by N_v with velocity weight $w_i = v_{\max}/N_v$ and N_{pc} with particle weight $w_p = u_{in} \Delta t / (\Delta x N_{pc})$. The time step is set to be $\Delta t = 0.9 \Delta x / v_{\max}$. In this study, the inlet boundary, i.e., Eq. (42), and zero-inflow boundary condition [40] are imposed at $x = 0$ and $x = 1$, respectively.

First, we investigate their accuracy with the same phase-space resolution, i.e., $N_x = 128$, $N_v = 256$, and $N_{pc} = 256$. Here we run this problem up to $t = 1.5$. Thanks to the semi-Lagrangian scheme in velocity space presented in Sec. III A 3, the time step in DUGKS is not restricted by the large electric force, i.e., $\Delta t \approx 4 \Delta v / |\nabla \phi|_{\max}$. Figure 10 shows the ions number density n and velocity u with different drift velocity u_d . It can be seen that both DUGKS and PIC results are consistent with theoretical solutions, i.e., Eq. (43), which is integrated using rectangular rule with $N_v = 3000$. However, we should point out there are some deviations in temperature results. Given the similar results, in the following we only present the results for $u_d = 3$, and the output time is 0.5.

Figure 11 shows the numerical ions temperature T and the theoretical solution. One can observe that DUGKS temperature depends on the velocity points N_v , while PIC temperature is more accuracy even with moderate particles, i.e., $N_{pc} = 128$. It should be noted that there are still slight deviations due to the numerical diffusion in physical space. Given the sharp gradient of the velocity [see Fig. 10(b)], we further increase physical size N_x but fix $N_v = 512$ and $N_{pc} = 128$. As shown in Fig. 12, the DUGKS and PIC result with $N_x = 512$ both agree well with theoretical solution, but a slight

deviation still exists in DUGKS due to the numerical diffusion in velocity space. This diffusion can be more clearly reflected in distribution function profiles. As shown in Fig. 13(a), two methods present consistent distribution functions at $x = 0$ and $x = 0.5$. However, at $x = 1$ [see Fig. 13(b)], the current DUGKS predicts flatter distribution function f , in comparison with PIC and theoretical solution.

Finally, we compare their computational costs. Table II shows their CPU time costs with the same parameters as in Fig. 10. Given the using of fixed velocity points N_v , DUGK costs are independent of drift velocity. In contrast, PIC costs decrease with the increase of drift velocity, due to the reduction of total number of particles. This indicates that the particles in PIC are self-adaptive in the whole velocity space, which is a big advantage to simulate the flow with a wide range of velocities. To clearly show this view, in Table III we present the CPU time costs and L_2 norm error with respect to temperature, where the parameters are the same as in Fig. 12. It can be seen that, in order to obtain same accuracy temperature, i.e., the L_2 is about 5×10^{-2} , the DUGKS costs are about two orders slower than PIC.

The above arguments indicate that proposed DUGKS involves the numerical diffusion, and the discrete velocity points should be fine enough to resolve the corresponding distribution function. In contrast, PIC can predict more accuracy results even with moderate particles adaptive in velocity space. As a result, PIC is superior to DUGKS for the study of plasma dynamics involving a wide range of velocities.

TABLE II. The total CPU time costs (in second) of DUGKS and PIC for plasma ion acceleration with different drift velocity at $t = 1.5$: $N_x = 128$, $N_v = 256$, and $N_{pc} = 256$.

u_d	0	1	2	3
DUGKS	37	37	37	37
PIC	24	20	17	14

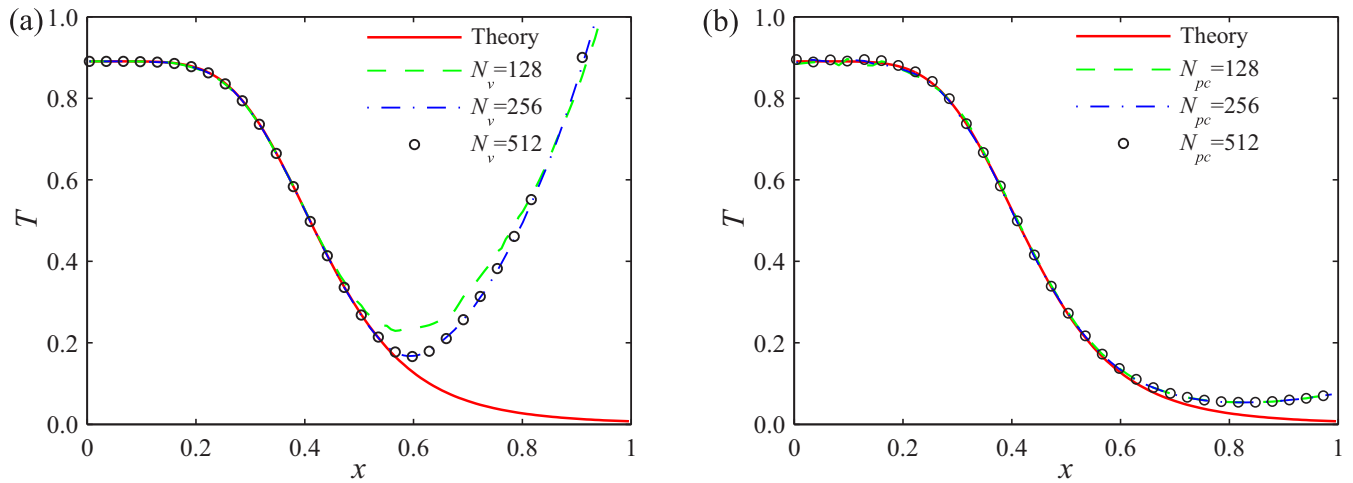


FIG. 11. Plasma ion acceleration, the comparison of DUGKS and PIC against different discrete velocity points: $N_x = 128$. The temperature predicted by DUGKS (a) and PIC (b).

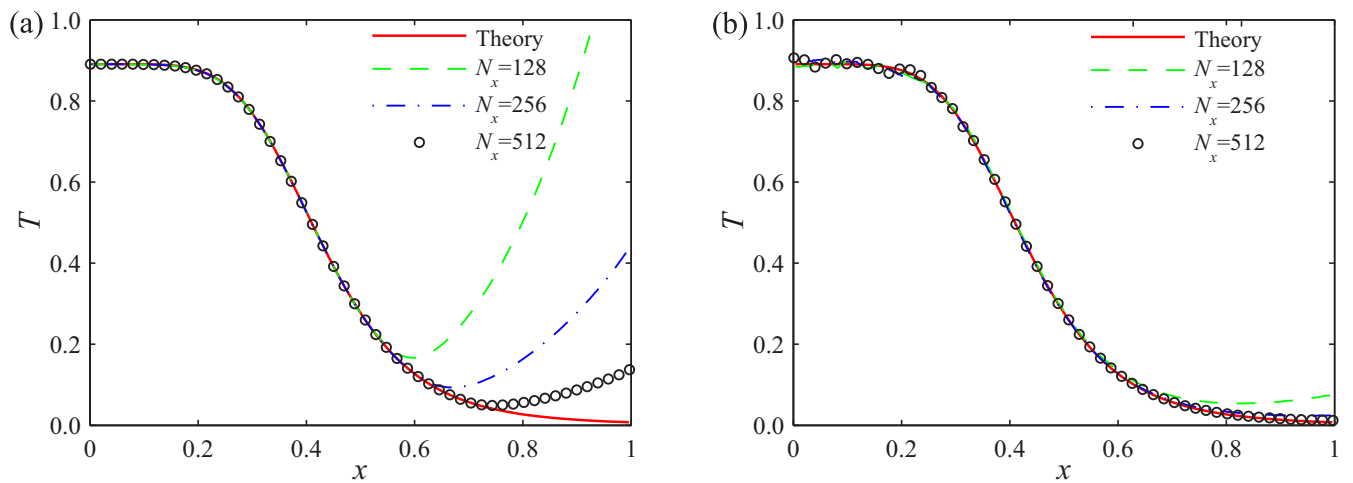


FIG. 12. Plasma ion acceleration, the comparison of DUGKS and PIC against different physical grids: $N_v = 512$ and $N_{pc} = 128$. The temperature predicted by DUGKS (a) and PIC (b).

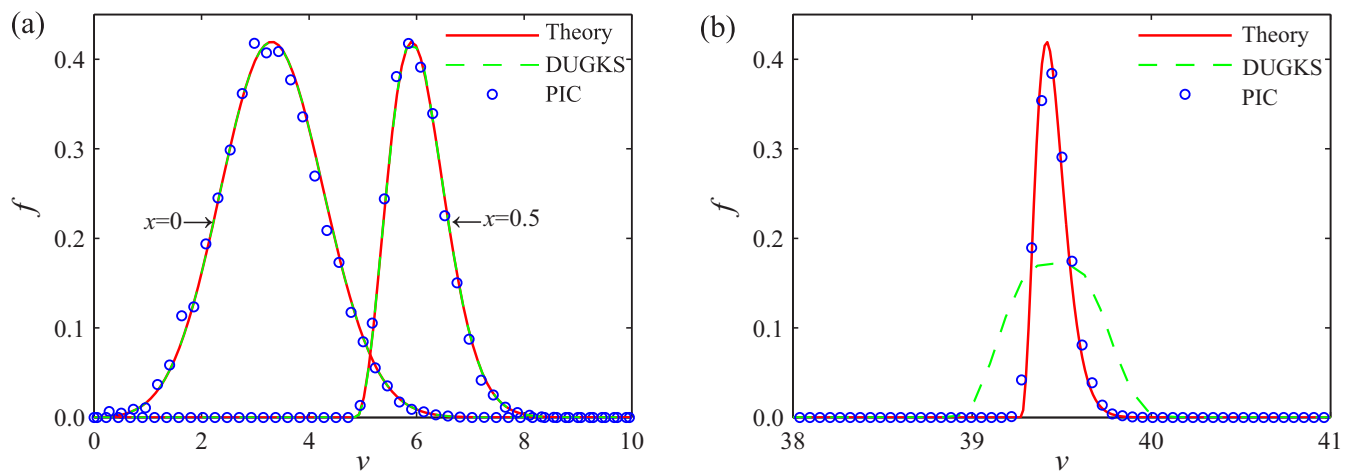


FIG. 13. Plasma ion acceleration: $N_x = 512$, $N_v = 512$, and $N_{pc} = 128$. The distribution function against the velocity space at $x = 0$ and $x = 0.5$ (a) and $x = 1$ (b).

TABLE III. The total CPU time costs (in second) and L_2 norm error with respect to temperature of DUGKS and PIC for plasma ion acceleration with different discrete physical grids at $t = 0.5$.

N_x	DUGKS, $N_v = 512$		PIC, $N_{pc} = 128$	
	Time	L_2	Time	L_2
128	26	6.12×10^{-1}	3	4.52×10^{-2}
256	107	1.91×10^{-1}	8	1.41×10^{-2}
512	445	5.23×10^{-2}	31	6.81×10^{-3}

V. CONCLUSIONS

In this paper, we present a stable, efficient and conservative DUGKS for electrostatic plasma with general boundary conditions. The current DUGKS not only economically and accurately provides a satisfactory solution for all Knudsen number regimes but also preserves the favorable conservative property and positivity in lengthy simulations. Several numerical experiments as well as the general PIC results are presented to validate the current DUGKS.

The current DUGKS is free of numerical noise, which can achieve more accuracy solutions at a fraction of computational costs with respect to PIC for the simulation of flows with small perturbation amplitude, e.g., the linear Landau damping. Besides, the semi-implicit coupling treatment of particle collision and transport enables the present DUGKS to be an efficient scheme in strong collision regimes. The numerical results suggest that the current DUGKS is superior to PIC for the study of warm plasma dynamics involving a small perturbation amplitude and strong particle collision. For these

reasons, we expect that the proposed DUGKS would be a promising tool for collisional electrostatic plasma waves.

The PIC is more desired for the study of plasma dynamics involving a wide range of velocities, e.g., plasma ion acceleration, since the particles are self-adaptive in the whole velocity space. In contrast, the discrete velocity points in DUGKS should be fine enough to resolve the corresponding distribution function which requires expensive computational cost. This issue will be more serious in high-dimensional simulations. Adaptive grid method and other advance technologies should be considered.

Clearly, either DUGKS or PIC has its pros and cons. One may want to use one or another method depending on the specific problem. We hope that the information given in this paper will help the researchers to make this choice. Finally, it should be noted that one often quoted that PIC is superior to the direct kinetic method in the numerical efficiency is mainly because the large noise level of PIC is accepted. For example, the noise in PIC, usually simulated with 100 macroparticles in per cell, exceeds the thermal noise level in typical space plasmas by several orders of magnitude. With the development of modern parallel computer hardware and the implementation of adaptive grid method, the efficiency of direct kinetic method will be much improved. These improvements will be considered in our future work.

ACKNOWLEDGMENTS

This work is partially supported by National Natural Science Foundation of China (Grant No. 11175052) and Shenzhen Technology Project (Grants No. JCYJ20170413103337899 and ZDSYS201707280904031).

- [1] F. F. Chen, *Introduction to Plasma Physics and Controlled Fusion*, Vol. 1 (Springer, Berlin, 1984).
- [2] R. W. Barber and D. R. Emerson, *Heat Transfer Eng.* **27**, 3 (2006).
- [3] P. L. Bhatnagar, E. P. Gross, and M. Krook, *Phys. Rev.* **94**, 511 (1954).
- [4] Y. Hu and J. Wang, *Phys. Plasmas* **24**, 033510 (2017).
- [5] Y. Hu and J. Wang, *Phys. Rev. E* **98**, 023204 (2018).
- [6] C. Birdsall and A. Langdon, *Plasma Physics via Computer Simulation* (Taylor & Francis, London, 2004).
- [7] R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (Taylor & Francis, London, 1988).
- [8] J. P. Verboncoeur, *Plasma Phys. Control. Fusion* **47**, A231 (2005).
- [9] D. R. Welch, D. V. Rose, R. E. Clark, T. C. Genoni, and T. Hughes, *Comput. Phys. Commun.* **164**, 183 (2004).
- [10] S. Markidis, G. Lapenta *et al.*, *Math. Comput. Simulat.* **80**, 1509 (2010).
- [11] S. Markidis and G. Lapenta, *J. Comput. Phys.* **230**, 7037 (2011).
- [12] E. Camporeale, G. L. Delzanno, B. Bergen, and J. D. Moulton, *Comput. Phys. Commun.* **198**, 47 (2016).
- [13] B. Wang, G. H. Miller, and P. Colella, *SIAM J. Sci. Comput.* **33**, 3509 (2011).
- [14] W. Deng and G.-Y. Fu, *Comput. Phys. Commun.* **185**, 96 (2014).
- [15] M. C. Pinto, E. Sonnendrücker, A. Friedman, D. P. Grote, and S. M. Lund, *J. Comput. Phys.* **275**, 236 (2014).
- [16] C. K. Birdsall, *IEEE T. Plasma Sci.* **19**, 65 (1991).
- [17] M. M. Turner, *Phys. Plasmas* **13**, 033506 (2006).
- [18] C.-Z. Cheng and G. Knorr, *J. Comput. Phys.* **22**, 330 (1976).
- [19] J. Wheaton, R. McGaffey, and P. Meszaros, *J. Comput. Phys.* **63**, 20 (1986).
- [20] T. Xiong, J.-M. Qiu, Z. Xu, and A. Christlieb, *J. Comput. Phys.* **273**, 618 (2014).
- [21] R. E. Heath, I. M. Gamba, P. J. Morrison, and C. Michler, *J. Comput. Phys.* **231**, 1140 (2012).
- [22] J. A. Rossmannith and D. C. Seal, *J. Comput. Phys.* **230**, 6203 (2011).
- [23] J.-M. Qiu and C.-W. Shu, *J. Comput. Phys.* **230**, 8386 (2011).
- [24] X. Cai, W. Guo, and J.-M. Qiu, *J. Comput. Phys.* **354**, 529 (2018).
- [25] E. Fijalkow, *Comput. Phys. Commun.* **116**, 319 (1999).
- [26] F. Filbet, E. Sonnendrücker, and P. Bertrand, *J. Comput. Phys.* **172**, 166 (2001).
- [27] J. W. Banks and J. A. F. Hittinger, *IEEE. T. Plasma. Sci.* **38**, 2198 (2010).
- [28] J.-M. Qiu and A. Christlieb, *J. Comput. Phys.* **229**, 1130 (2010).
- [29] J. W. Schumer and J. P. Holloway, *J. Comput. Phys.* **144**, 626 (1998).
- [30] B. Eliasson, *Comput. Phys. Commun.* **170**, 205 (2005).

- [31] S. Le Bourdiec, F. De Vuyst, and L. Jacquet, *Comput. Phys. Commun.* **175**, 528 (2006).
- [32] L. Wu, C. White, T. J. Scanlon, J. M. Reese, and Y. Zhang, *J. Comput. Phys.* **250**, 27 (2013).
- [33] T. Arber and R. Vann, *J. Comput. Phys.* **180**, 339 (2002).
- [34] F. Filbet and E. Sonnendrücker, *Comput. Phys. Commun.* **150**, 247 (2003).
- [35] G. Dimarco and L. Pareschi, *Acta Numer.* **23**, 369 (2014).
- [36] S. Yamamoto and H. Daiguji, *Comput. Fluids* **22**, 259 (1993).
- [37] R. Duclous, B. Dubroca, F. Filbet, and V. Tikhonchuk, *J. Comput. Phys.* **228**, 5072 (2009).
- [38] C.-W. Shu and S. Osher, *J. Comput. Phys.* **77**, 439 (1988).
- [39] J.-M. Qiu and C.-W. Shu, *Commun. Comput. Phys.* **10**, 979 (2011).
- [40] A. Christlieb, W. Guo, and Y. Jiang, *J. Comput. Phys.* **327**, 337 (2016).
- [41] S. Jin, *SIAM J. Sci. Comput.* **21**, 441 (1999).
- [42] F. Filbet and S. Jin, *J. Comput. Phys.* **229**, 7625 (2010).
- [43] G. Dimarco, L. Mieussens, and V. Rispoli, *J. Comput. Phys.* **274**, 122 (2014).
- [44] K. Xu and J.-C. Huang, *J. Comput. Phys.* **229**, 7747 (2010).
- [45] C. Liu and K. Xu, *Commun. Comput. Phys.* **22**, 1175 (2017).
- [46] Z. Guo, K. Xu, and R. Wang, *Phys. Rev. E* **88**, 033305 (2013).
- [47] Z. Guo, R. Wang, and K. Xu, *Phys. Rev. E* **91**, 033313 (2015).
- [48] H. Liu, Y. Cao, Q. Chen, M. Kong, and L. Zheng, *Comput. Fluids* **167**, 313 (2018).
- [49] H. Liu, M. Kong, Q. Chen, L. Zheng, and Y. Cao, *Phys. Rev. E* **98**, 053310 (2018).
- [50] J. Denavit and W. Kruer, *Phys. Fluids* **14**, 1782 (1971).
- [51] V. Saini, S. Pandey, P. Trivedi, and R. Ganesh, *Phys. Plasmas* **25**, 092107 (2018).
- [52] G. Manfredi and F. Valsaque, *Comput. Phys. Commun.* **164**, 262 (2004).
- [53] C. Berthon and F. Marche, *SIAM J. Sci. Comput.* **30**, 2587 (2008).
- [54] M. Gallis and J. Torczynski, *Phys. Fluids* **23**, 030601 (2011).
- [55] G. A. Bird and J. Brady, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Vol. 5 (Clarendon Press, Oxford, 1994).
- [56] Y. Cao, X. He, and T. Lü, *J. Comput. Phys.* **228**, 109 (2009).
- [57] A. J. Christlieb, R. Krasny, J. P. Verboncoeur, J. W. Emhoff, and I. D. Boyd, *IEEE T. Plasma Sci.* **34**, 149 (2006).
- [58] G. Dimarco, Q. Li, L. Pareschi, and B. Yan, *J. Plasma Phys.* **81**, 305810106 (2015).
- [59] K. Hara, Development of grid-based direct kinetic method and hybrid kinetic-continuum modeling of Hall thruster discharge plasmas, Ph.D. thesis, The University of Michigan, 2015.