Journal of Computational Physics 489 (2023) 112278

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Predicting continuum breakdown with deep neural networks

Tianbai Xiao [a,*], Steffen Schotthöfer [b], Martin Frank [b]

[a] State Key Laboratory of High Temperature Gas Dynamics, Institute of Mechanics, Chinese Academy of Sciences, Beijing, China
[b] Steinbuch Centre for Computing, Karlsruhe Institute of Technology, Karlsruhe, Germany

ARTICLE INFO

Article history:
Received 15 March 2022
Received in revised form 2 April 2023
Accepted 4 June 2023
Available online 8 June 2023

Keywords:
Computational fluid dynamics
Kinetic theory
Boltzmann equation
Multi-scale method
Deep learning

ABSTRACT

The multi-scale nature of gaseous flows poses tremendous difficulties for theoretical and numerical analysis. The Boltzmann equation, while possessing a wider applicability than hydrodynamic equations, requires significantly more computational resources due to the increased degrees of freedom in the model. The success of a hybrid fluid-kinetic flow solver for the study of multi-scale flows relies on accurate prediction of flow regimes. In this paper, we draw on binary classification in machine learning and propose the first neural network classifier to detect near-equilibrium and non-equilibrium flow regimes based on local flow conditions. Compared with classical semi-empirical criteria of continuum breakdown, the current method provides a data-driven alternative where the parameterized implicit function is trained by solutions of the Boltzmann equation. The ground-truth labels are derived rigorously from the deviation of particle distribution functions and the approximations based on the Chapman-Enskog ansatz. Therefore, no tunable parameter is needed in the criterion. Following the entropy closure of the Boltzmann moment system, a data generation strategy is developed to produce training and test sets. Numerical analysis shows its superiority over simulation-based samplings. A hybrid Boltzmann-Navier-Stokes flow solver is built correspondingly with an adaptive partition of local flow regimes. Numerical experiments including the one-dimensional Riemann problem, shear flow layer, and hypersonic flow around a circular cylinder are presented to validate the current scheme for simulating cross-scale and non-equilibrium flow physics. The quantitative comparison with a semi-empirical criterion and benchmark results demonstrates the capability of the current neural classifier to accurately predict continuum breakdown. The code for the data generator, hybrid solver, and neural network implementation is available in the open source repositories [1,2].

1. Introduction

Gases present a wonderfully diverse set of behaviors in different flow regimes. Such regimes are often categorized according to the Knudsen number, which is defined as the ratio of molecular mean free path to a characteristic length scale. With the variation of Knudsen number, the domain of flow physics can be qualitatively divided into continuum ($\mathrm{Kn} < 0.001$), slip ($0.001 < \mathrm{Kn} < 0.1$), transition ($0.1 < \mathrm{Kn} < 10$), and free molecular regimes ($\mathrm{Kn} > 10$) [3]. The Knudsen number indicates the relative importance between individual particle transports and their collective dynamics.

* Corresponding author.
E-mail address: txiao@imech.ac.cn (T. Xiao).

https://doi.org/10.1016/j.jcp.2023.112278
0021-9991/

Different governing equations are routinely established to describe the fluid motions at different scales. As an example, in rarefied gas where Kn is of $O(1)$, the particle transport and collision processes are distinguishable and can thus be modeled by two independent operators in the Boltzmann equation. In another limit with asymptotically small Kn, the Euler and Navier-Stokes equations are used to describe collective behaviors of fluid elements at a macroscopic level. It is worth mentioning that there is no quantitative description of the scale of a fluid element. Usually, it refers to a macroscopically infinitesimal concept, where the flow variables inside the element can be considered almost constant. With a high amount of intermolecular collisions, the fluid inside an element is considered to be in local thermodynamic equilibrium.

Computational fluid dynamics focuses on the numerical solution of the corresponding governing equations. The direct Boltzmann solvers employ a discretized phase space to compute transport and collision terms respectively. An alternative methodology is a direct simulation Monte Carlo (DSMC) method, which mimics the probability distribution function with a large number of test particles and the collision term is calculated statistically. On the other hand, the compressible Navier-Stokes solvers are mostly based on the Riemann solvers for inviscid flux and the central difference method for viscous terms. Only the macroscopic flow variables are tracked in the simulation. Compared with the kinetic methods, the computational cost of continuum fluid solvers is much lower.

Macroscopic and microscopic equations describe the physical evolution of the same substance and should correspond to each other. The well-known Chapman-Enskog expansion bridges such a connection [4], where the Euler and Navier-Stokes equations can be derived from the asymptotic limits of expansion solutions of the Boltzmann equation. Although the hydrodynamic equations are based on first-principle modeling, the Chapman-Enskog ansatz provides a rigorous criterion to define their validity. In other words, the usage of hydrodynamic equations incorporates the assumption that the Chapman-Enskog solution plays a proper approximation of the particle distribution function. However, this judgment cannot be verified in a macroscopic fluid simulation since the information on particle distribution functions has already been filtered in the coarse-grained modeling. The hydrodynamic equations may be misused where they don't apply in scientific and engineering practice.

Different criteria have been proposed to predict the failure of continuum mechanics and construct the corresponding multi-scale numerical algorithms. Some typical examples are listed below. Bird [5] proposed a parameter $\mathcal{P} = D(\ln\rho)/Dt/\nu$ for the DSMC simulation of expansion flows, where ρ is gas density and ν is collision frequency, and the breakdown threshold of translational equilibrium is set as $\mathcal{P} = 0.05$. Boyd et al. [6,7] extended the above concept to a gradient-length-local Knudsen number $\mathrm{Kn}_{GLL} = \ell|\nabla I|/I$, where ℓ is the local molecular mean free path and I is a scalar of interest, with the critical value being $\mathcal{C} = 0.05$. Garcia et al. [8] proposed a breakdown parameter based on dimensionless stress and heat flux $\mathcal{B} = \max(|\tau_*|, |q_*|)$, with the switching criterion of $\mathcal{B} = 0.1$. Levermore et al. [9] developed non-dimensional matrices from the moments of the particle distribution function. The tuning parameter $\Delta\mathcal{V}$ is then defined as the deviation of the eigenvalues of this matrix from their equilibrium values of unity, with the critical value of 0.25. The idea of all the above methods is to assemble components in the Chapman-Enskog expansion. However, since the ground-truth information of particle distribution is missing in a macroscopic fluid simulation, it is virtually impossible to employ the quantitative deviation between particle distributions from the full Boltzmann solution and the Chapman-Enskog reconstruction directly. It is difficult to prove that the above criteria can be universally applied to complex systems under different conditions of flow and geometry.

The rapid development of deep learning provides us with a promising alternative for classification and regression tasks. The relevant modeling and simulation strategies have been applied in fluid mechanics, e.g., building data-to-solution mapping [10–12], constructing physics-informed neural networks [13–15], identifying sparse dynamical systems [16–18], and solving kinetic equations [19–21]. In this paper, we turn to the application of binary classification. The idea is to employ neural networks as surrogate models, which classify the most probable flow regime based on local flow conditions. The neural networks accept macroscopic quantities including velocity moments and their slopes serve as inputs, and return labels of flow regimes. Following the principle of minimal entropy distributions, a data generation strategy is developed to sample particle distributions near and out of equilibrium in the training and test sets. Based on kinetic solutions, the ground-truth labels are rigorously determined by the deviation between the particle distribution functions and the Chapman-Enskog solutions. Therefore, a data-driven parameterized function is defined implicitly by the neural network in the high-dimensional function space. Based on the neural classifier, we develop a multi-scale hybrid method, which realizes a dynamic adaptation of flow regimes and fuses the continuum and kinetic solutions seamlessly.

The paper is organized as follows. In Sec. 2 we introduce some fundamental concepts in the kinetic theory of gases and the Chapman-Enskog expansion. Sec. 3 presents the idea and design of the neural network architecture. Sec. 4 introduces the strategy for generating data in training and test set. Sec. 5 details the numerical algorithm of the hybrid solver incorporated with the neural network classifier. Sec. 6 contains several numerical experiments to validate the current method. The last section is the conclusion. The nomenclature in the paper can be found in Table 6.

2. Kinetic theory

The Boltzmann equation describes the time-space evolution of a one-particle distribution function $f(t, \mathbf{x}, \mathbf{v})$ in dilute monatomic gas, i.e.,

$$\partial_t f + \mathbf{v}\cdot\nabla_{\mathbf{x}} f = Q(f,f) = \int_{\mathbb{R}^3}\int_{\mathbb{S}^2} \left[f(\mathbf{v}')f(\mathbf{v}'_*) - f(\mathbf{v})f(\mathbf{v}_*)\right]\mathcal{B}(\cos\theta, g)d\boldsymbol{\Omega}d\mathbf{v}_*, \tag{1}$$

where $\{\mathbf{v}, \mathbf{v}_*\}$ are the pre-collision velocities of two classes of colliding particles, and $\{\mathbf{v}', \mathbf{v}'_*\}$ are the corresponding post-collision velocities. The collision kernel $\mathcal{B}(\cos\theta, g)$ measures the probability of collisions in different directions, where θ is the deflection angle and $g = |\mathbf{g}| = |\mathbf{v} - \mathbf{v}_*|$ is the magnitude of relative pre-collision velocity. The solid angle $\boldsymbol{\Omega}$ is the unit vector along the relative post-collision velocity $\mathbf{v}' - \mathbf{v}'_*$, and the deflection angle satisfies the relation $\theta = \boldsymbol{\Omega}\cdot\mathbf{g}/g$.

The Boltzmann equation depicts a physical process with increasing physical entropy. The H-theorem indicates that the entropy is a Lyapunov function for the Boltzmann equation and the logarithm of its maximizer must be a linear combination of the collision invariants $\psi = (1, \mathbf{v}, |\mathbf{v}|^2/2)^T$ [22]. The equilibrium solution related to maximal entropy is the so-called Maxwellian distribution function,

$$\mathcal{M} := \rho\left(\frac{m}{2\pi kT}\right)^{3/2}\exp(-\frac{m}{2kT}\left(\mathbf{v}-\mathbf{V}\right)^2), \tag{2}$$

where m is molecular mass, $\mathbf{V}$ is macroscopic fluid velocity, T is temperature, and k is the Boltzmann constant.

The macroscopic conservative flow variables can be obtained by taking moments from the particle distribution function over velocity space, i.e.,

$$\mathbf{W} = \begin{pmatrix}\rho\\ \rho\mathbf{V}\\ \rho E\end{pmatrix} = \int f\psi d\mathbf{v}, \tag{3}$$

where $\rho E = \rho\mathbf{V}^2/2 + \rho e$, e is the internal energy per unit mass, and ψ is the vector of collision invariants. For an ideal gas, the internal energy is related to temperature as

$$\rho e = \frac{3}{2}nkT, \tag{4}$$

where $n = \rho/m$ is the number density. Taking moments of the Boltzmann equation with respect to collision invariants yields the transport equations for conservative variables,

$$\partial_t\mathbf{W} + \int_{\mathbb{R}^3}\psi\mathbf{v}\cdot\nabla_{\mathbf{x}} f d\mathbf{v} = 0, \tag{5}$$

i.e.,

$$\begin{aligned}
&\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{V}) = 0,\\
&\frac{\partial(\rho\mathbf{V})}{\partial t} + \nabla\cdot(\rho\mathbf{V}\otimes\mathbf{V}) = \nabla\cdot\mathbf{P},\\
&\frac{\partial(\rho E)}{\partial t} + \nabla\cdot(\rho E\mathbf{V}) = \nabla\cdot(\mathbf{P}\cdot\mathbf{V}) - \nabla\cdot\mathbf{q},
\end{aligned} \tag{6}$$

where $\otimes$ denotes dyadic product, and the stress tensor $\mathbf{P}$ and heat flux $\mathbf{q}$ are defined as,

$$\mathbf{P} = \int(\mathbf{v}-\mathbf{V})(\mathbf{v}-\mathbf{V})f d\mathbf{v}, \quad \mathbf{q} = \int\frac{1}{2}(\mathbf{v}-\mathbf{V})(\mathbf{v}-\mathbf{V})^2 f d\mathbf{v}. \tag{7}$$

The flux terms in the above equations are one order higher than the leading terms, which leads to the well-known closure problem [23]. Different closure strategies, i.e., different forms of the distribution function f, result in vastly different macroscopic transport equations. In the following, we briefly show the methodology of Chapman-Enskog ansatz, where the Navier-Stokes equations can be derived from the asymptotic solution of the Boltzmann equation. With the introduction of the following dimensionless variables

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L_0},\ \tilde{t} = \frac{t}{L_0/V_0},\ \tilde{\mathbf{v}} = \frac{\mathbf{v}}{V_0},\ \tilde{f} = \frac{f}{n_0 V_0^3}, \tag{8}$$

where $V_0 = \sqrt{2kT_0/m}$ is the most probable molecular speed, the Boltzmann equation can be reformulated as

$$\partial_{\tilde{t}}\tilde{f} + \tilde{\mathbf{v}}\cdot\nabla_{\tilde{\mathbf{x}}}\tilde{f} = \frac{1}{\mathrm{Kn}}Q(\tilde{f},\tilde{f}). \tag{9}$$

The Knudsen number is defined as

$$\mathrm{Kn} = \frac{V_0}{L_0 \nu_0} = \frac{\ell_0}{L_0}, \tag{10}$$

where ℓ_0 and ν_0 are the molecular mean free path and mean collision frequency in the reference state. For brevity, we drop the tilde notation to denote dimensionless variables henceforth.

Based on a small Knudsen number $\mathrm{Kn} = \varepsilon$, the Chapman-Enskog expansion approximates the particle distribution function [4] as,

$$f \simeq f_\varepsilon = \sum_{n=0}^{\infty} \varepsilon^n f^{(n)}, \quad f^{(0)} := \mathcal{M}. \tag{11}$$

Truncating the above expansion to the first non-trivial order, substituting it into Eq. (9) and projecting the kinetic system onto the hydrodynamic level, one can derive the Navier-Stokes equations.

Here we omit the tedious mathematical derivation and refer the reader to the literature [24]. The detailed expansion solution for the Navier-Stokes regime writes

$$\begin{aligned} f_{\text{NS, Boltzmann}} = \mathcal{M} \Bigg[& 1 - \frac{2\kappa}{5Rp} \left(\frac{\mathbf{c}^2}{2RT} - \frac{5}{2} \right) \mathbf{c} \cdot \nabla_{\mathbf{x}} (\ln T) \\ & - \frac{\mu}{RTp} \left(\mathbf{c} \otimes \mathbf{c} - \frac{1}{3} \mathbf{c}^2 \mathbf{I} \right) : \nabla_{\mathbf{x}} \mathbf{V} \Bigg], \end{aligned} \tag{12}$$

where R is the gas constant and $\mathbf{c} = \mathbf{v} - \mathbf{V}$ is the peculiar velocity. The viscosity and heat conductivity are determined by specific molecule models. For example, the viscosity coefficient for hard-sphere molecules takes the form

$$\mu = \mu_0 \left(\frac{T}{T_0} \right)^{\omega}, \tag{13}$$

where μ_0 is the viscosity coefficient in the reference state, ω is the power index that needs to be calibrated for different substances, and the heat conductivity is linked by the Prandtl number $\mathrm{Pr} = c_p \mu / \kappa$ where c_p is the specific heat of the gas at a constant pressure.

3. Neural network based classification of the flow regime

The universal approximation theorem [25], as a generalization of Stone-Weierstrass theorem [26], indicates that a neural network in its simplest form can approximate continuous functions on compact subsets of $\mathbb{R}^n$, provided that there are sufficient neurons under mild assumptions on the activation function. Defined in latent space and driven by data, the neural network simplifies data representations to find patterns in supervised learning. Such surrogate models can provide an alternative for semi-empirical criteria to classify the continuum breakdown regions of a flow field.

Following the spirit of Chapman-Enskog expansion, we build the neural network model as

$$\hat{\mathcal{R}} = \mathrm{NN}_\theta(\mathbf{U}), \quad \text{with} \quad \mathbf{U} = (\mathbf{W}, \nabla_{\mathbf{x}} \mathbf{W}, \tau)^\top \tag{14}$$

where θ denotes the trainable parameters of the neural network. As shown in Fig. 1, the input of neural network $\mathbf{U}$ is a combination of macroscopic variables, their slopes, and mean collision time. The idea to constitute such function inputs is to draw on the Chapman-Enskog ansatz and provide the necessary information for the reconstruction of probable particle distribution functions. The output $\hat{\mathcal{R}}$ is set to be a scalar, which denotes the likelihood for the current cell to be in the non-equilibrium regime. The depth of the neural network is set as 4. The architecture is set as $7 \times 28 \times 56 \times 28 \times 1$ for the one-dimensional flow problem and $9 \times 36 \times 72 \times 36 \times 1$ for the two-dimensional problem. The neural network employs the sigmoid function as activation in the last layer, and thus the output satisfies $\hat{\mathcal{R}} \in [0, 1]$ naturally. With the floor function, the output takes binary values, where 1 denotes rarefied (non-equilibrium) and 0 denotes continuum (near-equilibrium) regime.

In the supervised learning task, the dataset consists of a set of inputs and ground-truth labels corresponding to the function $\mathbf{U} \mapsto \hat{\mathcal{R}}$. For a given distribution function f_{ref}, the flow regime label is defined as

$$\mathcal{R} = \begin{cases} 1, & d > \epsilon \\ 0, & d \le \epsilon \end{cases}, \quad d = \frac{||f_{\text{NS}} - f_{\text{ref}}||_2}{\rho}, \tag{15}$$

where d denotes a normalized norm between the reference particle distribution function and the reconstructed Navier-Stokes distribution, and ϵ is an acceptable error value that defines the label and needs to be defined manually. In the current work, ϵ is chosen as 1%. Following the Chapman-Enskog ansatz, the Navier-Stokes distribution function can be constructed using Eq. (12). Note that the macroscopic quantities in the above equations can be obtained by taking moments of reference distribution function as in Eq. (3), and the collision time $\tau = 1/\nu$ can be derived from kinetic theory.

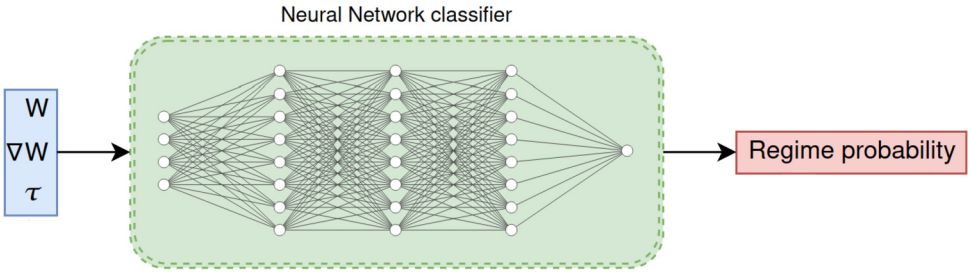


Fig. 1. The neural network-based regime classifier using macroscopic variables, their gradients, and the collision time to predict the flow regime of the current grid cell.

Given the definition of labels in the dataset, the idea of the current neural network becomes clear. The data-driven approach builds an implicit function $\mathbf{U} \mapsto \hat{\mathcal{R}}$ in the high-dimensional functional space spanned by neural network parameters.

The macroscopic flow variables, which are calculated from the reference kinetic solution f_{ref}, are inputs to the neural network, and its prediction is the flow regime. Thus one may understand the neural network's internal mechanism as an implicit reconstruction of the most probable kinetic solution, which is then compared to the Chapman-Enskog solution to determine the flow regime. The surrogate model provided by the neural network bridges macroscopic variables and flow regimes directly. Compared with classical criteria for continuum breakdown, no empirical and semi-empirical expansions are needed from asymptotic theory.

For this binary classification task, we employ the binary cross-entropy as a loss function, i.e.,

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \mathcal{R}_i \cdot \log \hat{\mathcal{R}}_i + (1-\mathcal{R}_i)\cdot \log\left(1-\hat{\mathcal{R}}_i\right), \tag{16}$$

where $\hat{\mathcal{R}}$ is the i-th scalar value in the model output, $\mathcal{R}$ is the corresponding target regime value, and N denotes the size of the training set. The cross-entropy is equivalent to fitting the model using maximum likelihood estimation. As the information entropy is constant given a defined dataset with fixed labels, minimizing the cross-entropy is equivalent to minimizing the Kullback-Leibler divergence between the empirical distribution of training data and the distribution induced by the model. The ADAM optimizer is used during all training processes. The training and testing data is produced by sampling and processing prescribed kinetic solutions of particle distribution functions, and the validation set is generated with the help of kinetic simulation data from numerical cases.

4. Data generation

As presented in Eq. (15), the information of exact particle distribution functions f is needed to compute macroscopic quantities $\mathbf{U}$ and regime labels. In the following, we consider the space of f as the sampling space under the constraint

$$f \in F = \left\{ f(\mathbf{v}) \geq 0 : \left| \int_{\mathbb{R}^3} f \psi_i d\mathbf{v} \right| < \infty, i = 0,1,2 \right\}, \tag{17}$$

i.e. the existence of the first 3 moments $\{\rho, \rho\mathbf{V}, \rho E\}$ and non-negativity of the particle distribution. A strategy to sample data from F usually creates a data-distribution p_F implicitly, which influences the training and test performance of the neural network. As the goal of the classification network is to find the separation hyperplanes between the near-equilibrium and non-equilibrium regime, we need to systematically create a data distribution p_F that generates enough samples near the boundary between regimes. A naive strategy is to sample data by performing numerical simulations and storing the required data in a post-processing fashion. The disadvantage of this is that p_F can be heavily biased toward the dynamics of the chosen test cases and might not necessarily cover enough different regions of flow regimes. Furthermore, it comes with the computational expense of a full kinetic solver, that might compute the same solutions multiple times, e.g. in the far-field of a fluid simulation. In the following, we demonstrate a sampling strategy to generate balanced data near and out of equilibrium.

4.1. Sampling of particle distribution functions

The sampling of data leverages the entropy closure of the Boltzmann moment system. We briefly introduce the principle here and refer to [23,27,28] for details. A general closure aims to reconstruct the particle distribution function f from a vector of moments

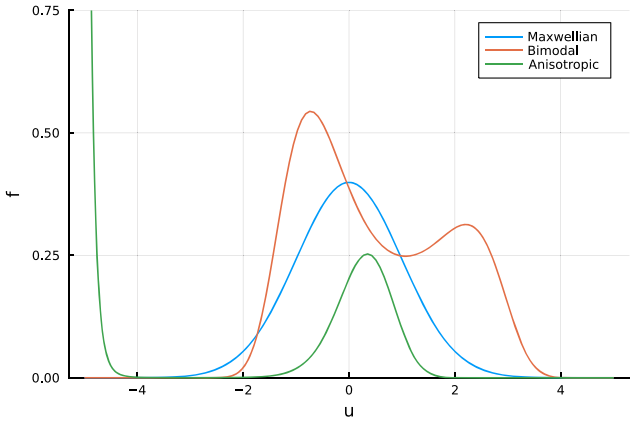


Fig. 2. Sampling of particle distribution functions. The more a function deviates from the Maxwellian, the higher the condition number of the corresponding entropy problem.

$$\mathbf{u} = \int_{\mathbb{R}^3} f \mathbf{m} d\mathbf{v} \in \mathbb{R}^{N_m}, \tag{18}$$

under the constraint

$$f \in F_M = \left\{ f(\mathbf{v}) \geq 0 : \left| \int_{\mathbb{R}^3} f m_i d\mathbf{v} \right| < \infty, i = 0, \dots, N_m \right\}, \tag{19}$$

where $\mathbf{m}(\mathbf{v}) \in \mathbb{R}^{N_m}$ is a vector of velocity dependent basis functions. We choose the basis in a way that the first three moments coincide with the conservative variables of the Navier-Stokes equations in Eq. (3). We thus rewrite $\mathbf{m}(\mathbf{v})$ in the following form,

$$\mathbf{m}(\mathbf{v}) = (\psi_0, \psi_1, \psi_2, \tilde{\mathbf{m}}(\mathbf{v}))^T, \tag{20}$$

where $\tilde{\mathbf{m}}(\mathbf{v})$ can be arbitrary monomials and mixed polynomials up to degree N_m and ψ_i are the collision invariants of the Boltzmann equation. In the current work, we choose $N_m = 4$, thus $\mathbf{m}$ resembles Levermore's 14 moment system [29], i.e.

$$\mathbf{m}(\mathbf{v}) = \left(1, \mathbf{v}, |\mathbf{v}|^2/2, \mathbf{v} \otimes \mathbf{v} - |\mathbf{v}|^2 \mathbf{I}, |\mathbf{v}|^2 \mathbf{v}, |\mathbf{v}|^4\right)^T \tag{21}$$

The minimal entropy closure employs an optimization problem to ensure the uniqueness of the solution of the closure problem. The objective function of the optimization problem is denoted by the integrated mathematical entropy density η. For the choice of entropy from Maxwell-Boltzmann statistics $\eta(f) = f \log(f) - f$ [30], the minimal entropy closure problem reads

$$\min_{g \in F_m} \int_{\mathbb{R}^3} g \log(g) - g d\mathbf{v} \quad \text{s.t. } \mathbf{u} = \int_{\mathbb{R}^3} \mathbf{m} g d\mathbf{v}. \tag{22}$$

If a solution of this optimization problem exists,[1] it can be represented as

$$f_u(\mathbf{v}) = \exp(\boldsymbol{\alpha}_u \mathbf{m}(\mathbf{v})), \tag{23}$$

where $\boldsymbol{\alpha}_\mathbf{u} \in \mathbb{R}^{N_m}$ is the vector of Lagrange multipliers of the dual formulation of the optimization problem, which reads

$$\boldsymbol{\alpha}_\mathbf{u} = \underset{\boldsymbol{\alpha} \in \mathbb{R}^{N_m}}{\operatorname{argmin}} \left\{ \int_{\mathbb{R}^3} \exp(\boldsymbol{\alpha} \cdot \mathbf{m}) d\mathbf{v} - \boldsymbol{\alpha} \cdot \mathbf{u} \right\}. \tag{24}$$

The idea is to generate distribution functions which are solutions of the minimal entropy closure using Eq. (23). Specifically, we sample the corresponding Lagrange multipliers $\boldsymbol{\alpha}$. For the sake of simplicity, we drop the subscript notation in the following. For example, the Maxwellian in Eq. (2) can be expressed with the following choice of $\boldsymbol{\alpha}$,

[1] Even if $\mathbf{u} \in \mathscr{R}$, there may not exist a solution to Eq. (22) [31].

$$\begin{aligned} &\mathcal{M} = \exp(\boldsymbol{\alpha}\cdot\mathbf{m}),\ \boldsymbol{\alpha} = (\alpha_0, \alpha_1, \alpha_2, \dots, \alpha_n)^T, \\ &\alpha_0 = \ln(\rho/(2\pi kT)^{3/2}) - \mathbf{V}^2/(2kT),\ \alpha_1 = \frac{\mathbf{V}}{kT},\ \alpha_2 = -\frac{1}{2kT},\ \alpha_n = 0 \text{ for all } n > 2. \end{aligned} \tag{25}$$

The disturbance from the equilibrium state can be controlled for example by choosing $\alpha_n \neq 0$ for $n > 2$. For a fixed length N_m of the Lagrange multiplier vector $\boldsymbol{\alpha}$, we sample α_n for $n > 0$ normally distributed with a prescribed standard deviation. The sampling mean is chosen according to the Lagrange multiplier, which recovers the Maxwellian above. Without loss of generality, we assume that u_0 which corresponds to ρ in terms of conservative variables equals one. For $\alpha_n \neq 0$, $n > 2$, in general, the computed particle density $\rho \neq 1$. To enforce the assumption, that $u_0 = \rho = 1$ we use for a given set of sampled coefficients $\boldsymbol{\alpha}$ the ansatz

$$u_0 = \rho = 1 = \int_{\mathbb{R}^3} \exp(\boldsymbol{\alpha}\cdot\mathbf{m}) d\mathbf{v}. \tag{26}$$

Applying the natural logarithm to both sides of the equation, we get

$$\alpha_0 := \vartheta(\boldsymbol{\beta}) = -\ln\left(\int_{\mathbb{R}^3} \exp(\boldsymbol{\beta}\cdot(m_1, m_2, \dots, m_{N_m})^T) d\mathbf{v}\right) \tag{27}$$

with $\boldsymbol{\beta} = (\alpha_1, \alpha_2, \dots, \alpha_{N_m})^T$- The set of all moments u for which the minimal entropy problem in Eq. (22) has a solution is called the realizable set

$$\mathscr{R} = \left\{ \mathbf{u} : \mathbf{u} = \int_{\mathbb{R}^3} g\mathbf{m} d\mathbf{v},\ g \in F_m \right\}. \tag{28}$$

Further, for $m_i = \psi_i$, the corresponding moment u_i equals a conservative variable W_i of Eq. (3). It should be noted that the minimal entropy problem has no solution at the boundary $\partial\mathscr{R}$ of the realizable set and its condition number σ_H increases when approaching the boundary. The condition number of the minimal entropy closure at a moment $\mathbf{u}$ can be computed via the positive semi-definite Hessian of the dual problem (24), i.e.

$$H(\boldsymbol{\alpha}) = \int_{\mathbb{R}^3} \mathbf{m}\otimes\mathbf{m}\exp(\boldsymbol{\alpha}\cdot\mathbf{m}) d\mathbf{v}. \tag{29}$$

Using the condition number of H_u we can control the sampling of reference densities in near-equilibrium and non-equilibrium regimes. In this work, we use the condition number threshold as $\tau_{\text{cond}} = 10^{-3}$ for the basis (21).

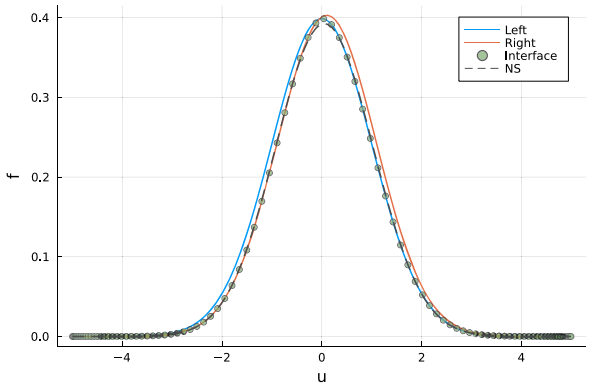
Reconstructed particle distributions with moments for which the minimal entropy closure has a low condition number are typically similar to the Maxwellian. Distribution functions corresponding to moments near $\partial\mathscr{R}$, where the minimal entropy problem is ill-conditioned, are highly anisotropic and have a high distance to a Maxwellian, which is illustrated in Fig. 2. Further insights into the realizability condition have been studied in detail [30,32–36]. The resulting sampling strategy is summarized in Algorithm 1.

Algorithm 1: Minimum entropy sampling of kinetic densities.

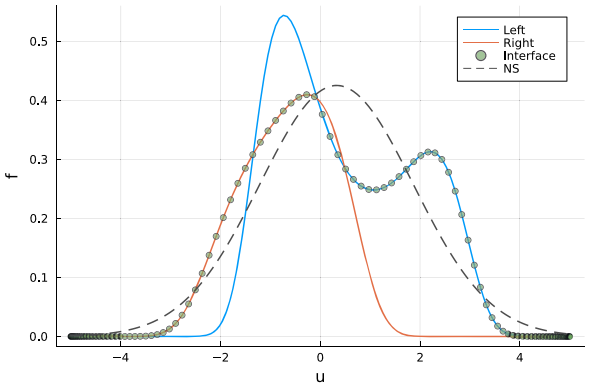
Input: $\boldsymbol{V}_{tr}$: Truncated velocity domain
$\sigma_{\boldsymbol{\alpha}}$: Sampling std. deviation for $\boldsymbol{\alpha}$
T, $\boldsymbol{V}$: Temperature and bulk velocity
τ_{cond}: Condition number threshold

Result: F_T: Set of sampled kinetic densities

for $i = 0$ *to* $i = T$ **do**
  $\boldsymbol{\beta}_{\text{mean}} \leftarrow \left[\frac{\mathbf{V}^\top}{kT}, -\frac{1}{2kT}, 0, \dots\right]^\top$ /* Compute sampling mean */
  do
    $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\beta}_{\text{mean}}, \sigma_{\boldsymbol{\alpha}})$ /* Sample reduced multipliers */
    $\boldsymbol{\alpha} \leftarrow \left[\vartheta(\boldsymbol{\beta}), \boldsymbol{\beta}^\top\right]^\top$ /* Reconstruct normalized multipliers */
  while $\sigma_H < \tau_{cond}$
  $f_i \leftarrow \exp(\boldsymbol{\alpha}\cdot\mathbf{m})$ /* Compute kinetic density */
  Append f_i to F_T.
end

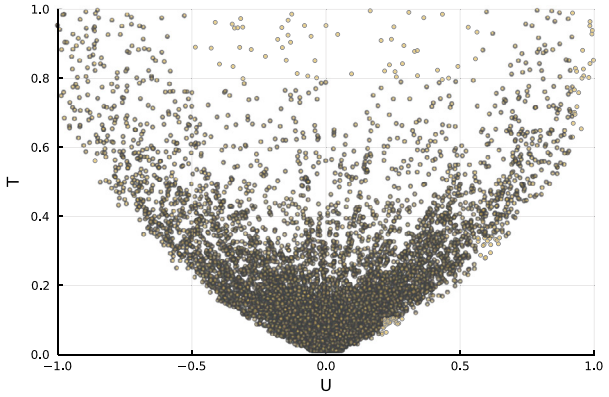


(a) Equilibrium solutions in ghost cells, $\tau = 0.0012$

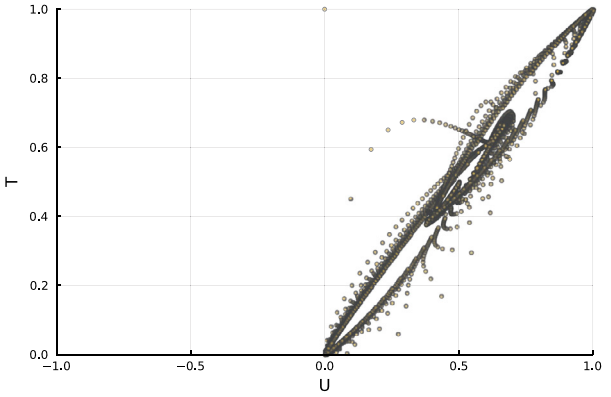


(b) Non-equilibrium solutions in ghost cells, $\tau = 0.000511$

Fig. 3. Sampling of reference solution at the interface of two neighboring ghost cells and Chapman-Enskog reconstruction for $Kn = 0.001$ and $dx = 0.01$ in 1 spatial dimension.



(a) Algorithmic



(b) Solver

Fig. 4. Distributions of data points in U-T phase diagram from the current algorithmic generator and sampled from Sod shock tube solution.

Note that Eq. (23) requires a velocity mesh to record the generated distribution function. In the current work, we truncate the velocity domain as $\mathbf{v} \in \boldsymbol{V}_{tr} = \left[-4\sqrt{RT_0},\ 4\sqrt{RT_0}\right]$ where $T_0 = 10$, and employ 200 quadrature points in each velocity direction.

4.2. Assembly of the training data

The input of the neural network contains not only a set of conservative variables but also their gradients and local collision time. The idea for data generation is to combine two sampled distribution functions $\{f_L, f_R\}$ with two adjacent ghost cells, of which the positions $\{\mathbf{x}_L, \mathbf{x}_R\}$ as well as the unit normal vector $\mathbf{n}$ are randomly sampled. Therefore, the reference particle distribution function at the interface can be approximated via an upwind reconstruction,

$$f_{\text{ref}}(\mathbf{v}) = f_L(\mathbf{v}) H(\mathbf{n}\cdot\mathbf{v}) + f_R(\mathbf{v})\,(1 - H(\mathbf{n}\cdot\mathbf{v}))\,, \tag{30}$$

where H is the Heaviside step function. The conservative variables $\{\mathbf{W}, \mathbf{W}_L, \mathbf{W}_R\}$ are obtained by taking moments of f_{ref}, and the gradients $\nabla_x \mathbf{W}$ are computed with a finite difference formula. Fig. 3a) displays the upwind approximation and Chapman-Enskog reconstruction in Eq. (12) from the corresponding conservative variables at the interface of two ghost cells with near equilibrium distributions and Fig. 3b) the reconstruction of two non-equilibrium solutions. One sees, that in Fig. 3a) the Chapman-Enskog reconstruction is close to the upwind approximation, whereas in Fig. 3b) the respective distributions have a very different shapes.

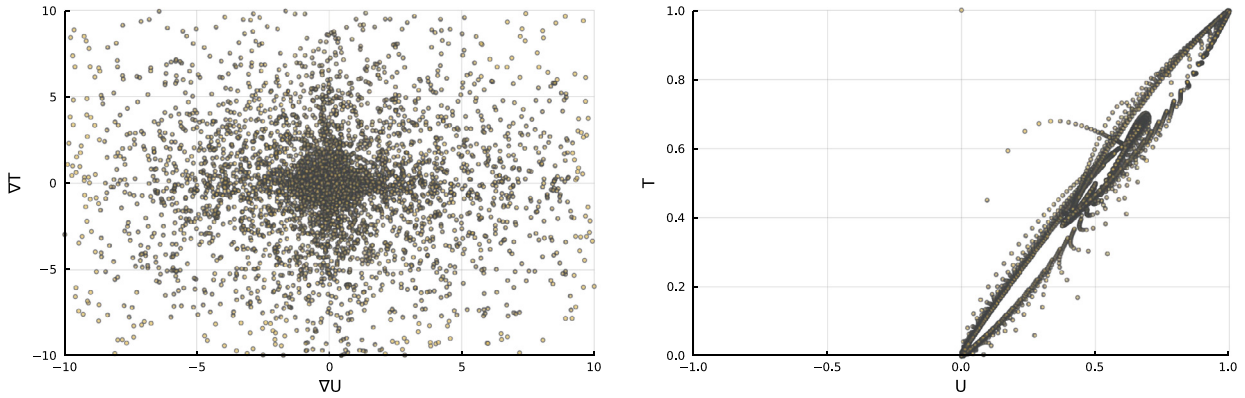
Using a randomly sampled Knudsen-number Kn from a predefined range, we can compute the local collision time $\tau = 1/\nu$ and obtain a completely assembled training data point $\mathbf{U} = (\mathbf{W}, \nabla_{\mathbf{x}}\mathbf{W}, \tau)$. Finally, we compute the label of the training data point by first computing f_{NS} using Eq. (12) and then calculating the distance to the sampled reference solution f_{ref} using Eq. (15). The resulting sampling strategy is displayed in Algorithm 2.

Algorithm 2: Sampling of labeled training data.

Input: $[\mathrm{Kn}_{min}, \mathrm{Kn}_{max}]$: Range of Knudsen numbers
$[\rho_{min}, \rho_{max}]$: Range of particle densities
$[\boldsymbol{x}_{min}, \boldsymbol{x}_{max}]$: Range of cell-center distances

Result: $X_T = \left\{(\mathbf{Q}_i, \mathcal{R}_i)\right\}_{i \in T}$: Training data-set

$F_T \leftarrow$ `Algorithm 1` `/* Compute reference kinetic density functions */`
for $i = 0$ *to* $i = T$ **do**
  $f_L,\ f_R \sim F_T$ `/* Sample` f `of left and right cell */`
  $\rho_L,\ \rho_R \sim \mathtt{uniform}([\rho_{min}, \rho_{max}])$ `/* Sample macroscopic density to` f_L `and` f_R `*/`
  $f_L,\ f_R \leftarrow \rho_L f_L,\ \rho_R f_R$ `/* Scale` f_L `and` f_R `*/`
  $\mathbf{u}_L, \mathbf{u}_R \leftarrow \int \boldsymbol{\phi} f_L d\mathbf{v}, \int \boldsymbol{\phi} f_R d\mathbf{v}$ `/* Compute macroscopic variables */`
  $\mathbf{x}_L,\ \mathbf{x}_R \sim \mathtt{uniform}[\boldsymbol{x}_{min}, \boldsymbol{x}_{max}]^d$ `/* Sample ghost cell centroids */`
  $\mathbf{n} \leftarrow (\mathbf{x}_L - \mathbf{x}_R)/\left\|\mathbf{x}_L - \mathbf{x}_R\right\|$ `/* Compute cell interface normal */`
  $f_{\mathrm{ref}}(\mathbf{v}) \leftarrow f_L(\mathbf{v})\mathcal{H}(\mathbf{n}\cdot\mathbf{v}) + f_R(\mathbf{v})\left(1 - \mathcal{H}(\mathbf{n}\cdot\mathbf{v})\right)$ `/* Upwind scheme */`
  $\mathbf{W}_i \leftarrow \int \boldsymbol{\phi} f_{\mathrm{ref}} d\mathbf{v}$ `/* Compute macroscopic variables */`
  $\nabla_{\mathbf{x}}\mathbf{W}_i \leftarrow (\mathbf{W}_L - \mathbf{W}_R)/\left\|\mathbf{x}_L - \mathbf{x}_R\right\|$ `/* Compute slope of macro. var. */`
  $\tau_i \leftarrow \mu/p$ `/* Hard-sphere model for` τ `*/`
  $f_{\mathrm{NS}} \leftarrow$Eq. (12) `/* BGK reconstruction of` f_{NS} `*/`
  $\mathcal{R}_i \leftarrow$Eq. (15) `/* Determine local regime */`
  $\mathbf{U}_i \leftarrow \left[\mathbf{W}_i^\top, \nabla_{\mathbf{x}}\mathbf{W}_i^\top, \tau_i\right]^\top$
  Append $(\mathbf{U}_i, r_i)$ to X_T.
end



(a) Algorithmic generator (b) Solver sampler

Fig. 5. Distributions of data points in ∇U-∇T phase diagram from the current algorithmic generator and sampled from Sod shock tube solution.

To illustrate the superiority of the current data generation strategy, we compare the data distributions resulting from Algorithm 2 to the data gathered from the simulation results of the standard Sod shock tube problem with a full Boltzmann simulation. Details of the setup can be found in Sec. 6.1. Fig. 4(a) shows the macroscopic variables generated by the data generator using Algorithm 2 and Fig. 4(b) displays the generation from the simulation results. The results have been normalized via $\tilde{\mathbf{W}} = \mathbf{W}/\rho$ is displayed. It is evident, that the samples from the kinetic solver have a strong bias toward positive bulk velocity. Temperature and velocity are strongly correlated. In contrast, the algorithmic sampler generates a wide range of macroscopic variables with different ranges of $\mathbf{U}$ and T. Besides, the generated gradients of the macroscopic variables $\nabla_x\mathbf{W}$ are shown in Fig. 5. The data sampled by the generator is shown in Fig. 5(a) and exhibits a distribution that is concentrated around the origin, without strong bias towards a specific direction, whereas the data generated by the solver in Fig. 5(b) displays again a strong bias and fails to cover most parts of the domain. Furthermore, the presented sampling strategy does not require the computational expense of full simulations, possibly with multiple initial conditions. Computational resources for the data sampler can be found in [1].

5. Solution algorithm

In this section, we present the numerical implementation of the adaptive scheme based on the neural classifier. The solution algorithm is built on top of a finite volume method.

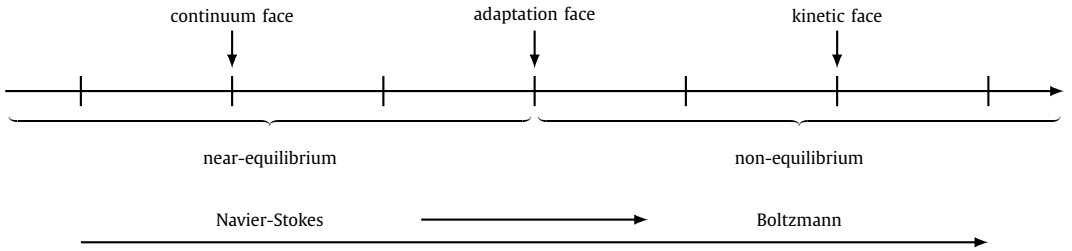


Fig. 6. Schematic of the adaptive scheme for multi-scale flow.

5.1. Kinetic solver

Given the notation of cell-averaged particle distribution function in the physical element $\mathbf{\Omega}_i$ and velocity element $\mathbf{\Omega}_j$,

$$f_{i,j}^n = \frac{1}{\mathbf{\Omega}_i(\mathbf{x})\mathbf{\Omega}_j(\mathbf{v})} \int_{\mathbf{\Omega}_i} \int_{\mathbf{\Omega}_j} f(t^n, \mathbf{x}, \mathbf{v}) d\mathbf{x} d\mathbf{v}, \tag{31}$$

the update algorithm of the finite volume scheme writes

$$f_{i,j}^{n+1} = f_{i,j}^n - \frac{1}{\mathbf{\Omega}_i} \sum_{r \in \partial \mathbf{\Omega}_i} \int_{t^n}^{t^{n+1}} \mathbf{F}_{r,j}^f \cdot \boldsymbol{n}_r S_r dt + \int_{t^n}^{t^{n+1}} Q_j(f_i, f_i) dt, \tag{32}$$

where $\mathbf{n}_r$ is the unit normal vector of surface r that points outside of the element $\mathbf{\Omega}_i$, and S_r is the surface area. The interface flux of distribution function $\mathbf{F}^f$ can be computed via an upwind reconstruction,

$$\mathbf{F}_{i+1/2,j}^f = \mathbf{v}_j \left(f_L H(\mathbf{v}_j \cdot \mathbf{n}_{i+1/2}) + f_R \left(1 - H(\mathbf{v}_j \cdot \mathbf{n}_{i+1/2})\right)\right), \tag{33}$$

where H is the Heaviside step function, and the status on the left and right sides of the interface are reconstructed via

$$\begin{aligned} f_L &= f_{i,j} + \nabla f_{i,j} \cdot (\mathbf{x}_{i+1/2} - \mathbf{x}_i), \\ f_R &= f_{i+1,j} + \nabla f_{i+1,j} \cdot (\mathbf{x}_{i+1/2} - \mathbf{x}_{i+1}). \end{aligned} \tag{34}$$

Inside each element, the collision term $Q(f, f)$ is computed by the fast spectral method [2]. The discrete Fourier transform is employed to solve the convolution in the spectral domain efficiently. We refer to [37] for a detailed formulation of this method.

5.2. Navier-Stokes solver

We define the average conservative flow variables in an element as

$$\mathbf{W}_i^n = \frac{1}{\mathbf{\Omega}_i(\mathbf{x})} \int_{\mathbf{\Omega}_i} \mathbf{W}(t^n, \mathbf{x}) d\mathbf{x}, \tag{35}$$

and the finite volume algorithm writes

$$\mathbf{W}_i^{n+1} = \mathbf{W}_i^n - \frac{1}{\mathbf{\Omega}_i} \sum_{r \in \partial \mathbf{\Omega}_i} \int_{t^n}^{t^{n+1}} \mathbf{F}_r^W \cdot \boldsymbol{n}_r S_r dt. \tag{36}$$

A key step for solving conservation laws is to compute the fluxes $\mathbf{F}^W$ of conservative variables. Here, we employ the Chapman-Enskog solution from the BGK-type relaxation model [38] to construct numerical fluxes. The relaxation model writes

$$\partial_t f + \mathbf{v} \cdot \nabla_{\mathbf{x}} f = \nu(\mathcal{E} - f). \tag{37}$$

The equilibrium distribution $\mathcal{E}$ can be chosen as the Maxwellian in Eq. (2) or its variants [39,40], and ν is the collision frequency. The above equation can be written in the following successive form

$$f = \mathcal{E} - \tau D_t \mathcal{E} + \tau D_t(\tau D_t \mathcal{E}) + \cdots, \tag{38}$$

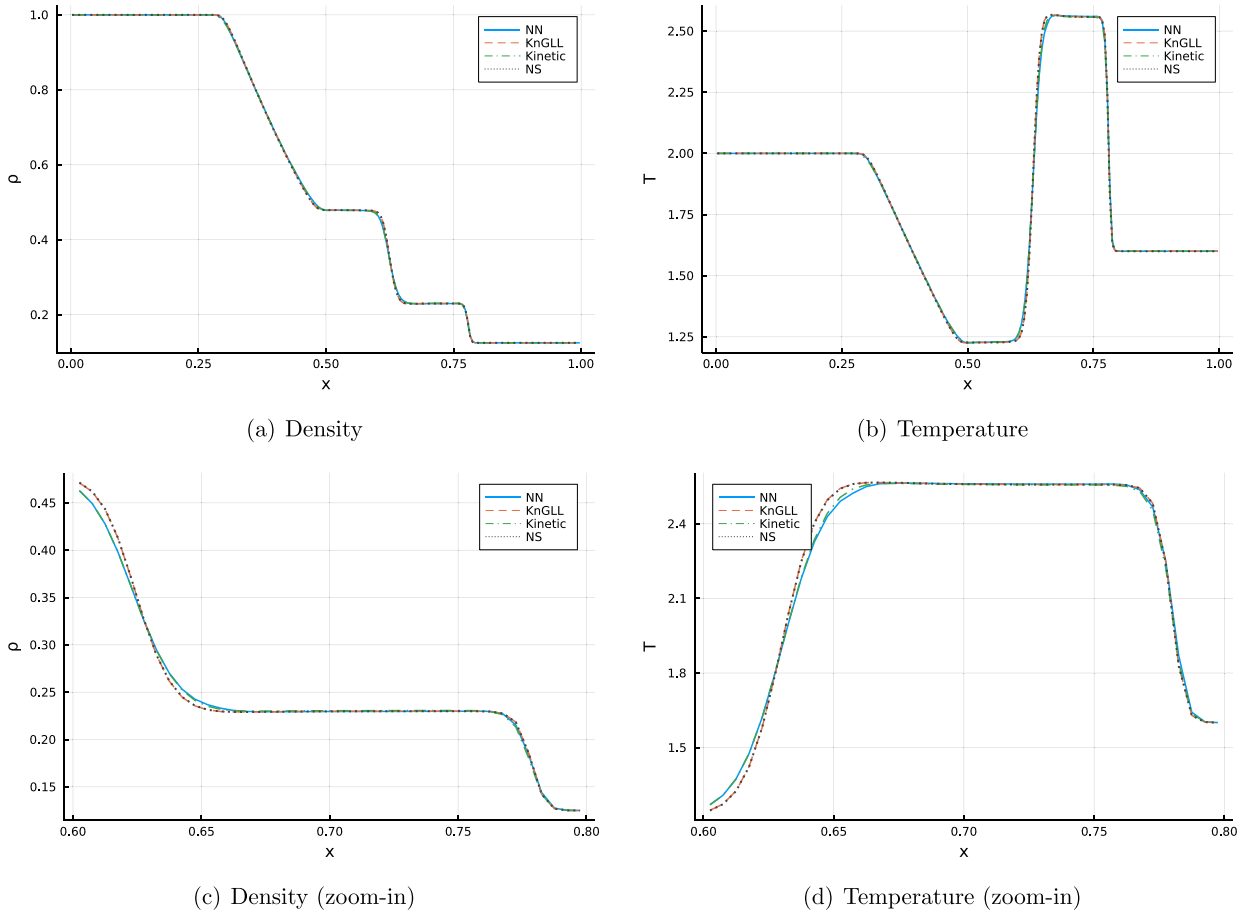


Fig. 7. Prediction of flow regimes from fully kinetic solutions at $t = 0.15$ in the Sod shock tube with different criteria (0 denotes near-continuum, 1 denotes non-equilibrium).

where D_t denotes total derivative operator and $\tau = 1/\nu$. The above equation has the same structure as Eq. (11), and thus the first-order truncation of Chapman-Enskog expansion writes [41],

$$f \simeq \mathcal{E} - \tau(\partial_t \mathcal{E} + \mathbf{v} \cdot \nabla_{\mathbf{x}} \mathcal{E}). \tag{39}$$

In the solution algorithm, we follow the Chapman-Enskog expansion and construct the particle distribution function at interface $\mathbf{x}_{i+1/2}$ with an upwind approach,

$$\begin{aligned} f_L &= \mathcal{E}_L \left(1 - \tau \left(\mathbf{a}_L \cdot \mathbf{v} + b_L\right)\right), \\ f_R &= \mathcal{E}_R \left(1 - \tau \left(\mathbf{a}_R \cdot \mathbf{v} + b_R\right)\right), \end{aligned} \tag{40}$$

where $\{\mathcal{E}_L, \mathcal{E}_R\}$ are the equilibrium distributions computed from reconstructed macroscopic variables, i.e.,

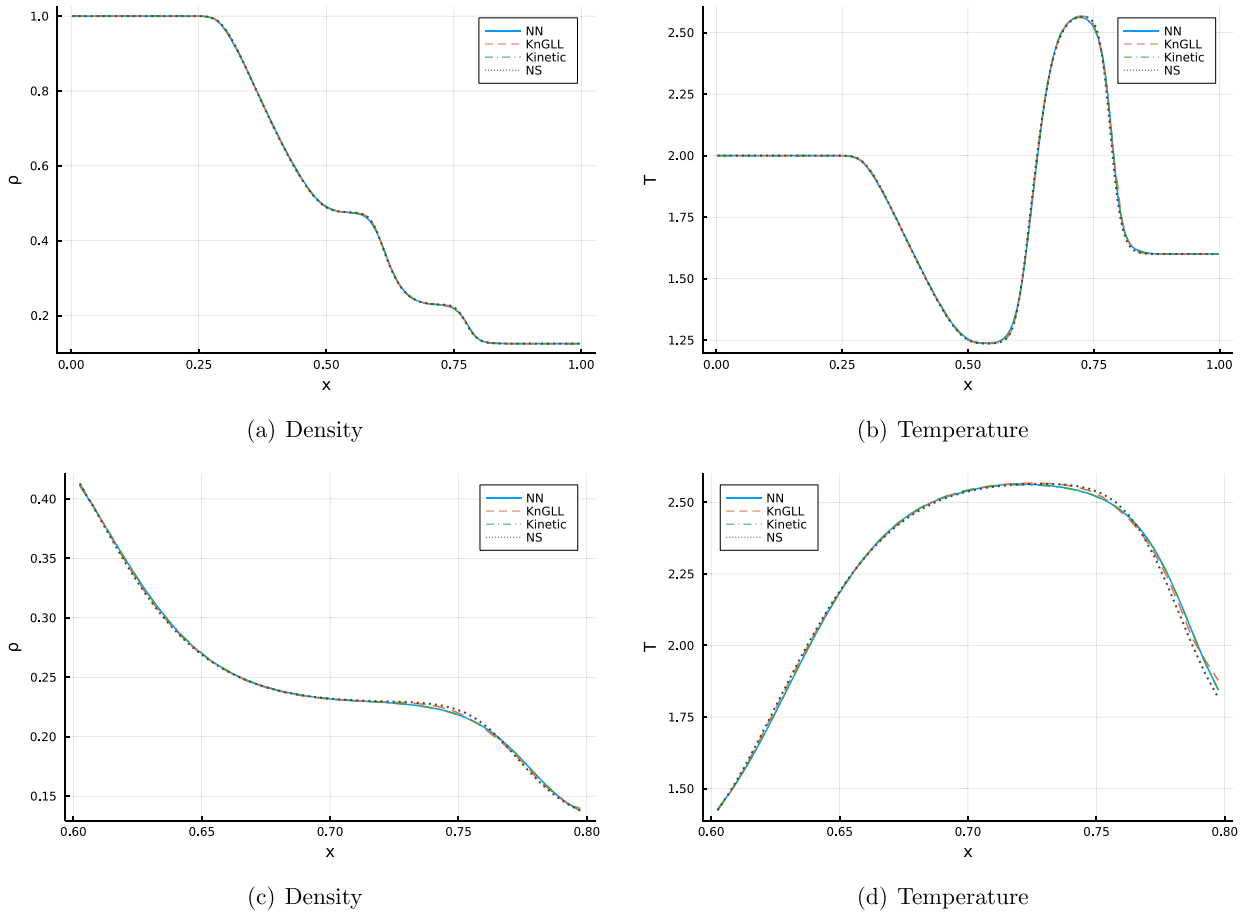
$$\begin{aligned} \mathbf{W}_L &= \mathbf{W}_{i,j} + \nabla \mathbf{W}_{i,j} \cdot (\mathbf{x}_{i+1/2} - \mathbf{x}_i), \\ \mathbf{W}_R &= \mathbf{W}_{i+1,j} + \nabla \mathbf{W}_{i+1,j} \cdot (\mathbf{x}_{i+1/2} - \mathbf{x}_{i+1}). \end{aligned} \tag{41}$$

In a well-resolved region, the relation $\mathbf{W}_L = \mathbf{W}_R$ holds, and Eq. (40) deduces to standard Chapman-Enskog expansion naturally. The spatial derivatives of the particle distribution function $\mathbf{a}_{L,R}$ are related to macroscopic slopes via

$$\int \mathbf{a}_{L,R} \mathcal{E}_{L,R} \psi d\mathbf{v} = \nabla_{\mathbf{x}} \mathbf{W}_{L,R}, \tag{42}$$

where $\psi = (1, \mathbf{v}, \mathbf{v}^2/2)^T$ are the collision invariants. Then $\mathbf{a}_{L,R}$ can be obtained by solving a linear system [42]. Then the time derivative $b_{L,R}$ can be obtained through the compatibility condition of the BGK model, i.e.,

$$\int \nu(\mathcal{E} - f)\psi d\mathbf{v} = 0, \tag{43}$$



(a) Density (b) Temperature

(c) Density (d) Temperature

Fig. 8. Profiles of density and temperature in the shock tube at $t = 0.15$ under $\mathrm{Kn} = 0.0001$.

which yields

$$\int b_{L,R}\mathcal{E}_{L,R}\psi d\mathbf{v} = -\int (a_{L,R}\cdot\mathbf{v})\mathcal{E}_{L,R}\psi d\mathbf{v}. \tag{44}$$

After the coefficients for spatial and time variations are determined, the interface fluxes for macroscopic variables can be obtained by taking moments over particle velocity space, i.e.,

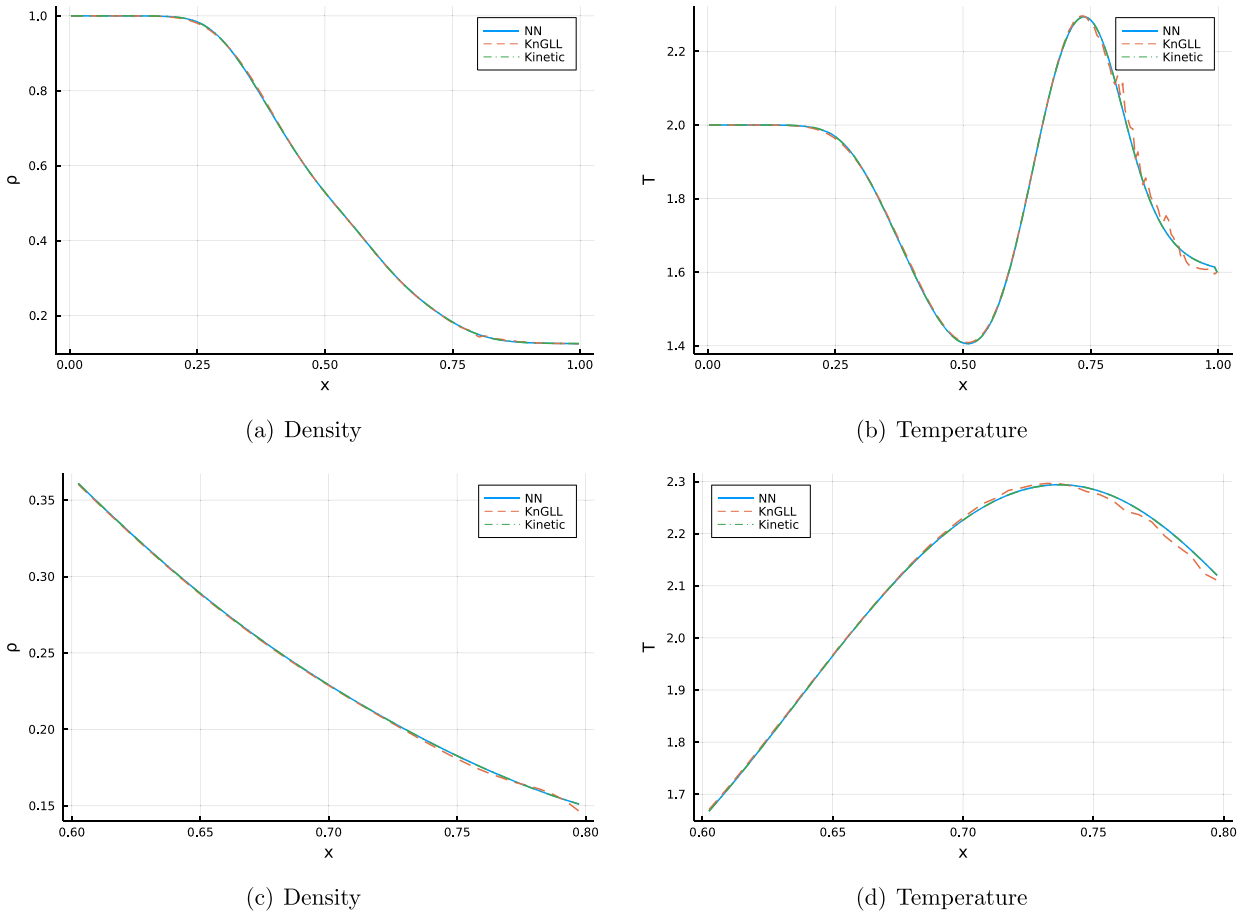
$$\mathbf{F}^{W}_{i+1/2,j} = \int \mathbf{v}\left(f_L H(\mathbf{v}\cdot\mathbf{n}_{i+1/2}) + f_R\left(1 - H(\mathbf{v}\cdot\mathbf{n}_{i+1/2})\right)\right)\psi d\mathbf{v}, \tag{45}$$

where H is the Heaviside step function. Since the equilibrium state is based on Gaussian distribution, the above integral can be evaluated analytically. Remarkably, the above numerical method can be understood as a simplification of the gas-kinetic scheme [42,43].

5.3. Adaptation strategy

The Boltzmann and Navier-Stokes solvers can be combined to solve multi-scale flow problems efficiently with an adaptive continuous-discrete velocity transformation. The work paradigm is shown in Fig. 6. For a near-equilibrium flow region, the particle distribution function is formulated analytically from the Chapman-Enskog expansion. Therefore, only the macroscopic flow variables are needed to store and iterate by the Navier-Stokes solver in Eq. (36). For non-equilibrium flows, the solution algorithm allocates the localized velocity quadrature to track the evolution of the particle distribution function in Eq. (32).

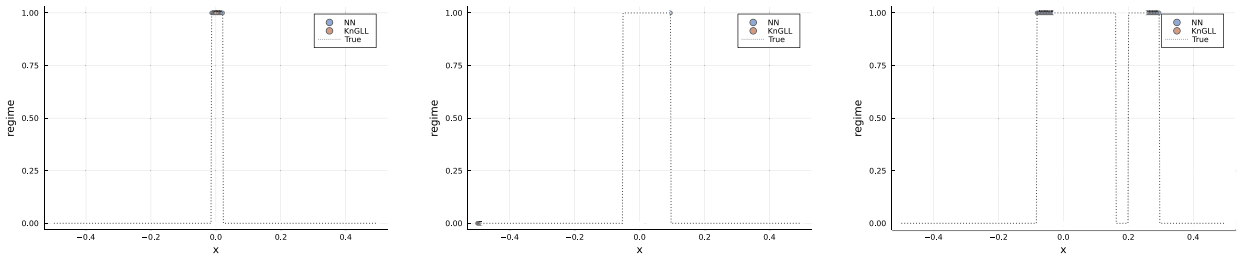
A core task of the hybrid solver lies in the dynamic adaptation of time-varying flow regimes at different locations. At every time step t^n, the spatial derivatives of the updated macroscopic variables are evaluated via $\nabla_{\mathbf{x}}\mathbf{W} = (\nabla_{\mathbf{x}}\mathbf{W}_L + \nabla_{\mathbf{x}}\mathbf{W}_R)/2$, where $\nabla_{\mathbf{x}}\mathbf{W}_{L,R}$ are the difference values between to neighboring cells. The collision time is evaluated by $\tau = \mu/p$. Therefore, the complete information needed for the neural network to predict the flow regime has been obtained. As shown in Fig. 6,



(a) Density (b) Temperature

(c) Density (d) Temperature

Fig. 9. Profiles of density and temperature in the shock tube at $t = 0.15$ under $\mathrm{Kn} = 0.001$.



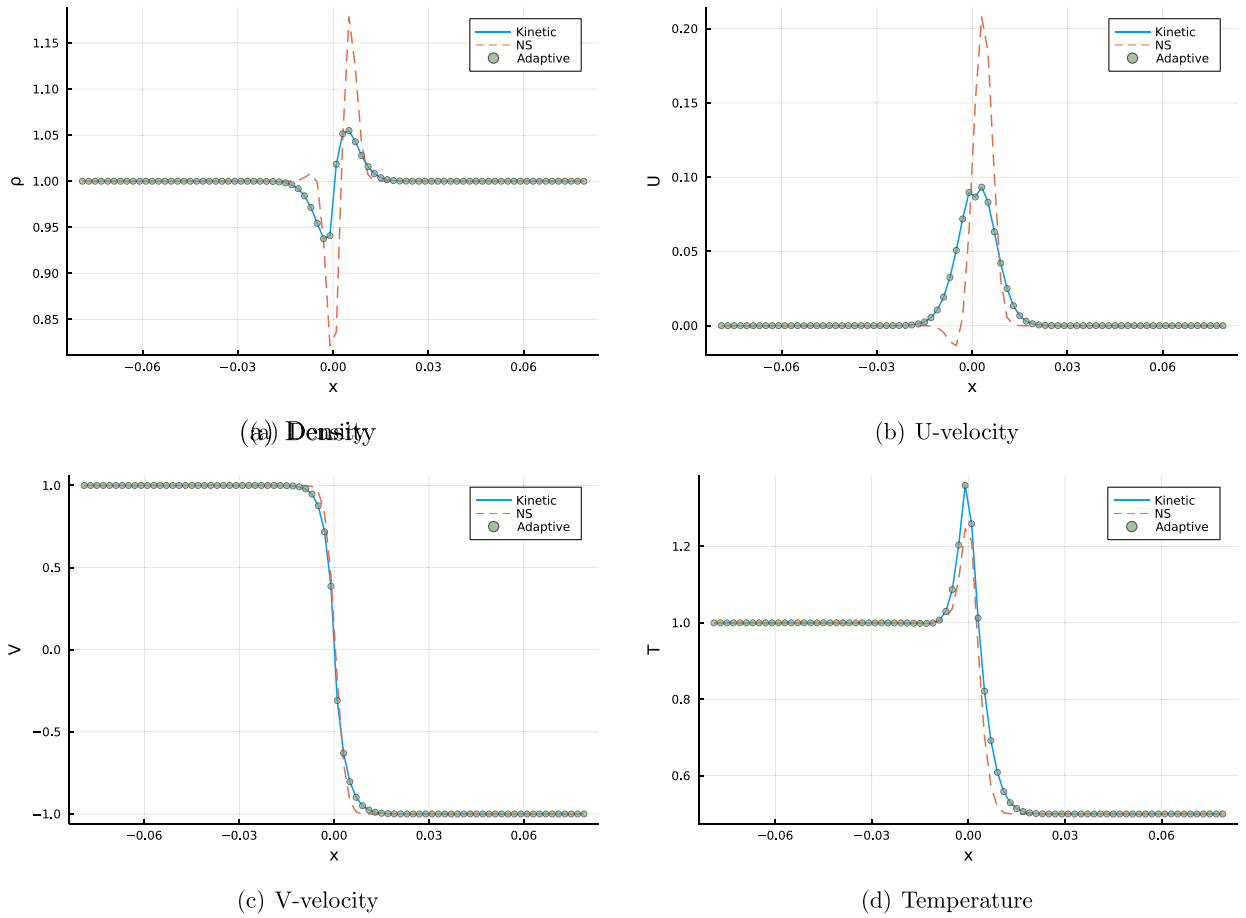
(a) $t = \tau_0$ (b) $t = 10\tau_0$ (c) $t = 50\tau_0$

Fig. 10. Profiles of density and temperature in the shock tube at $t = 0.15$ under $\mathrm{Kn} = 0.01$.

we have two types of cells, i.e. the non-equilibrium one holding discrete solution of the distribution function and the near-equilibrium one with Navier-Stokes variables, and three types of cell interfaces based on the flow regimes, i.e.,

- kinetic face: two neighboring cells are in a non-equilibrium flow regime;
- continuum face: two neighboring cells of the face are in near-equilibrium flow regime;
- adaptation face: two neighboring cells of the face lie in different flow regimes.

The solution algorithm in type 1/2 cells is straightforward following the section 5.1 and 5.2. At the adaptation face, both macroscopic and microscopic fluxes are evaluated to update the solutions in the left and right cells. This is uniformly done by computing the kinetic flux in Eq. (33), where its velocity moments result in macroscopic fluxes, i.e.,



(a) Density (b) U-velocity

(c) V-velocity (d) Temperature

Fig. 11. Prediction of flow regimes from fully kinetic solutions at different time instants in the shear layer with different criteria (0 denotes near-equilibrium, 1 denotes non-equilibrium).

$$\mathbf{F}^W = \int \mathbf{F}^f \psi d\mathbf{v} \simeq \sum_j^{N_q} w_j \mathbf{F}_j \psi, \tag{46}$$

where N_q is the number of quadrature points and w_j the quadrature weights. To utilize the above equation, a local velocity mesh is generated within

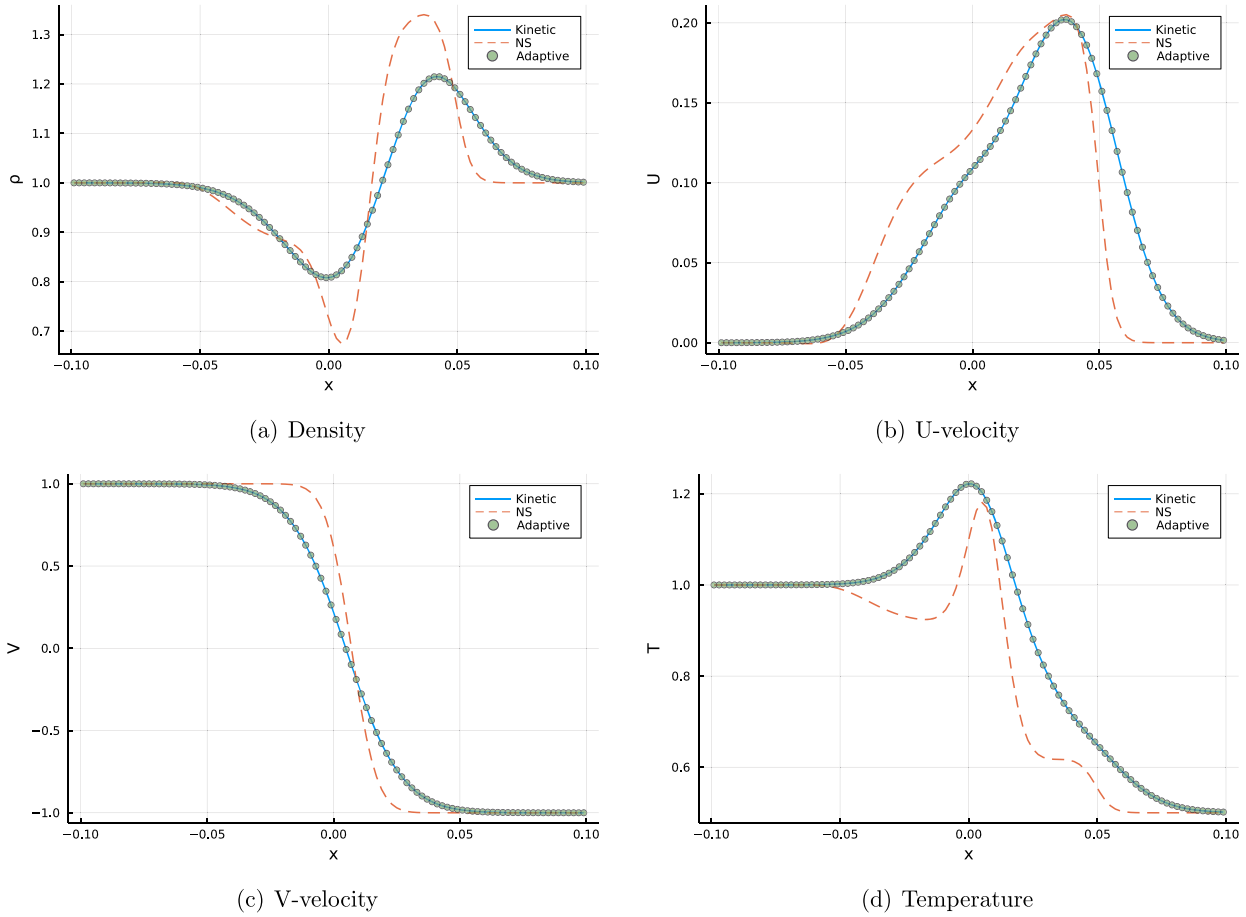
$$\mathbf{v} \in \left[-|\mathbf{V}_0| - 4\sqrt{RT_0},\ |\mathbf{V}_0| + 4\sqrt{RT_0} \right], \tag{47}$$

where $\{\mathbf{V}_0, T_0\}$ are reference velocity and temperature, and R is the gas constant. The velocity grid is chosen such that more than 99% of values of the Maxwellian distribution fall into its range. In a continuum cell at t^n which has a discrete solution of distribution function at t^{n-1}, the memory can be freed by deallocations in static languages, e.g., C and Fortran, and by setting to be None type in dynamic languages, e.g., Python and Julia. In a kinetic cell with no former record of the discretized distribution function, the solution is reconstructed from the Chapman-Enskog expansion in Eq. (12) in the continuum cell and then used for flux evaluation. This way, a hybrid continuum-kinetic solver has been set up, where no buffer zone is required to transit solutions.

6. Numerical experiments

In this section, we conduct numerical experiments of several multi-scale flow problems to validate the neural classifier and the corresponding adaptive solver. All the variables are nondimensionalized following the paradigm introduced in Sec. 2.

The hard-sphere gas model is employed in all cases. We choose the gradient-length-local Knudsen number Kn_{GLL} [6] as a reference and provide some quantitative comparisons to predict continuum breakdown. It is worth mentioning that we are not here to censure this methodology, but rather to choose a widely accepted criterion as a benchmark to point out potential possibilities of our new method. The computational resources of the hybrid solver can be found in [2].



(a) Density (b) U-velocity

(c) V-velocity (d) Temperature

Fig. 12. Profiles of flow variables in the shear layer at $t = \tau$.

Table 1
Computational setup of Sod shock tube problem.

t	x	N_x	$\mathbf{v}$	N_u	N_v	N_w
$[0, 0.15]$	$[0, 1]$	200	$[-8, 8]^3$	64	32	32
Quadrature	Kn	CFL	Integrator	Boundary		
Rectangular	$[0.0001, 0.01]$	0.5	Euler	Dirichlet		

6.1. Sod shock tube

The first numerical experiment is the Sod shock tube, where the longitudinal processes dominate the flow motion in the one-dimensional Riemann problem. The particle distribution function is initialized as a Maxwellian, which corresponds to the following macroscopic variables

$$\begin{pmatrix} \rho \\ U \\ T \end{pmatrix}_{t=0,L} = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}, \quad \begin{pmatrix} \rho \\ U \\ T \end{pmatrix}_{t=0,R} = \begin{pmatrix} 0.125 \\ 0 \\ 1.6 \end{pmatrix}.$$

To test the capability of the current scheme to solve multi-scale flow problems, simulations are performed with different reference Knudsen numbers ranging in $\mathrm{Kn} = [0.0001, 0.01]$. The detailed computation setup is listed in Table 1.

We first conduct a fully kinetic simulation with the Boltzmann equation. Based on the kinetic solution, the partition of flow regimes based on different criteria is shown in Fig. 7. The ground-truth regime is obtained from the L^2 error between the particle distribution and its Chapman-Enskog reconstructed value in Eq. (15). It is clear that localized flow structures, including rarefaction wave, contact discontinuity, and shock wave, contribute as sources of non-equilibrium effects. In the remaining near-equilibrium regions the Chapman-Enskog expansion can approximate real particle distributions. With the increasing Knudsen number, the kinetic regime enlarges due to the increasing rarefied gas effect.

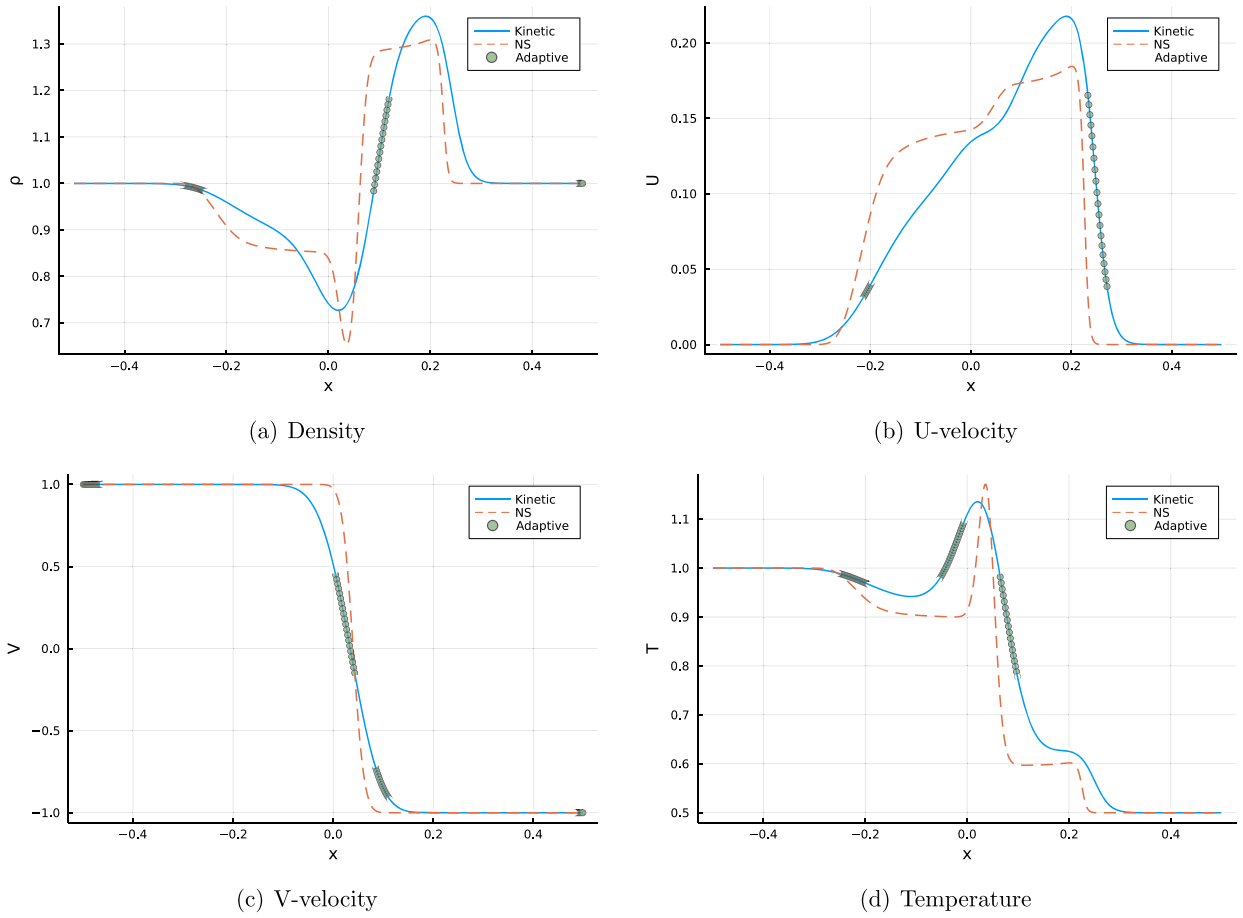


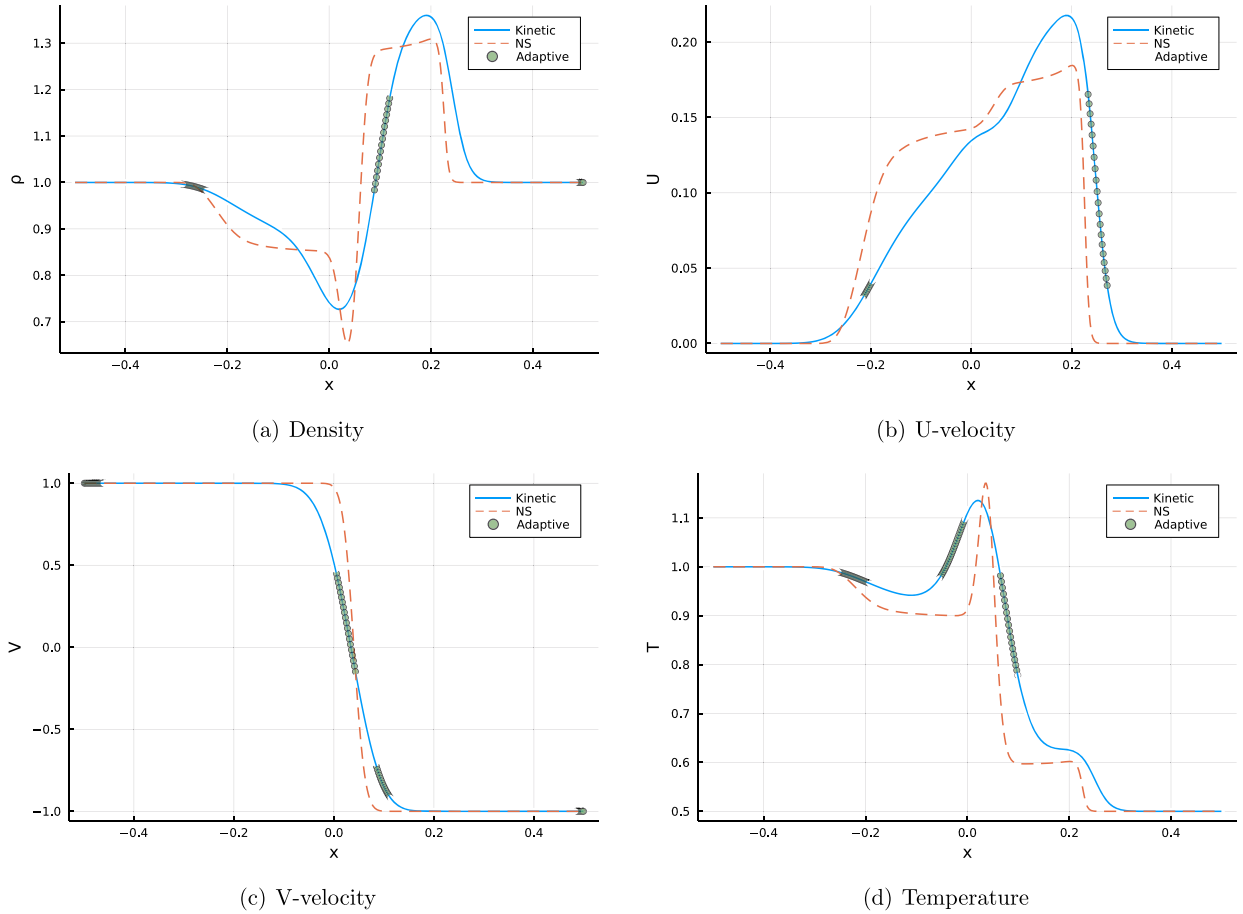
Fig. 13. Profiles of flow variables in the shear layer at $t = 10\tau$.

Table 2
Computational cost of the sod shock tube problem.

	time	total allocations	total allocated memory
Navier-Stokes	1.39 s	2.16×10^7	1.82 GB
Kinetic	1649.02 s	1.65×10^8	7.35 TB
Adaptive (Kn=0.0001)	97.50 s	2.72×10^7	121.04 GB
Adaptive (Kn=0.001)	514.90 s	3.60×10^7	713.92 GB
Adaptive (Kn=0.01)	1209.60 s	9.88×10^7	3.88 TB

From the results, we can see that the gradient-length-local Knudsen number criterion underestimates the influence of wave structures and makes inaccurate predictions. On the contrary, the neural network predicts equivalent flow regimes as the benchmark. Then, we employ the adaptive solver to conduct complete simulations based on the criteria from the neural network and Kn_{GLL}. The profiles of density and temperature inside the shock tube at the time instant $t = 0.15$ under different Knudsen numbers are presented in Fig. 8, 9 and 10. The kinetic and Navier-Stokes solutions are plotted as a benchmark. As shown, although all the results are qualitatively similar, the zoom-in view demonstrates that the hybrid method based on Kn_{GLL} provides the Navier-Stokes solution, while the neural network method accurately predicts the local flow regime and recovers the Boltzmann solution. Therefore, we have a basis to believe that the neural network classifier works better than the Kn_{GLL} method in this case. At $Kn = 0.01$, the Chapman-Enskog expansion yields negative values in the particle distribution function where the spatial slopes are large, resulting in the failure of Navier-Stokes solutions. In this case, the inaccurate prediction of flow regimes from Kn_{GLL} results in unreasonable oscillations of macroscopic solutions, which is overcome by the neural network classifier.

Table 2 provides the computational cost of all these three solvers. As can be seen, the adaptive scheme accelerates the simulation significantly in the continuum and transition flow regimes, and reduce the memory load.



(a) Density (b) U-velocity

(c) V-velocity (d) Temperature

Fig. 14. Profiles of flow variables in the shear layer at $t = 50\tau$.

Table 3
Computational setup of shear layer problem.

t	x	N_x	$\mathbf{v}$	N_u	N_v	N_w
$[0, 50\tau_0]$	$[-0.5, 0.5]$	500	$[-6, 6]^3$	64	28	28
Quadrature	Kn	CFL	Integrator	Boundary		
Rectangular	0.005	0.5	Euler	Dirichlet		

6.2. Shear layer

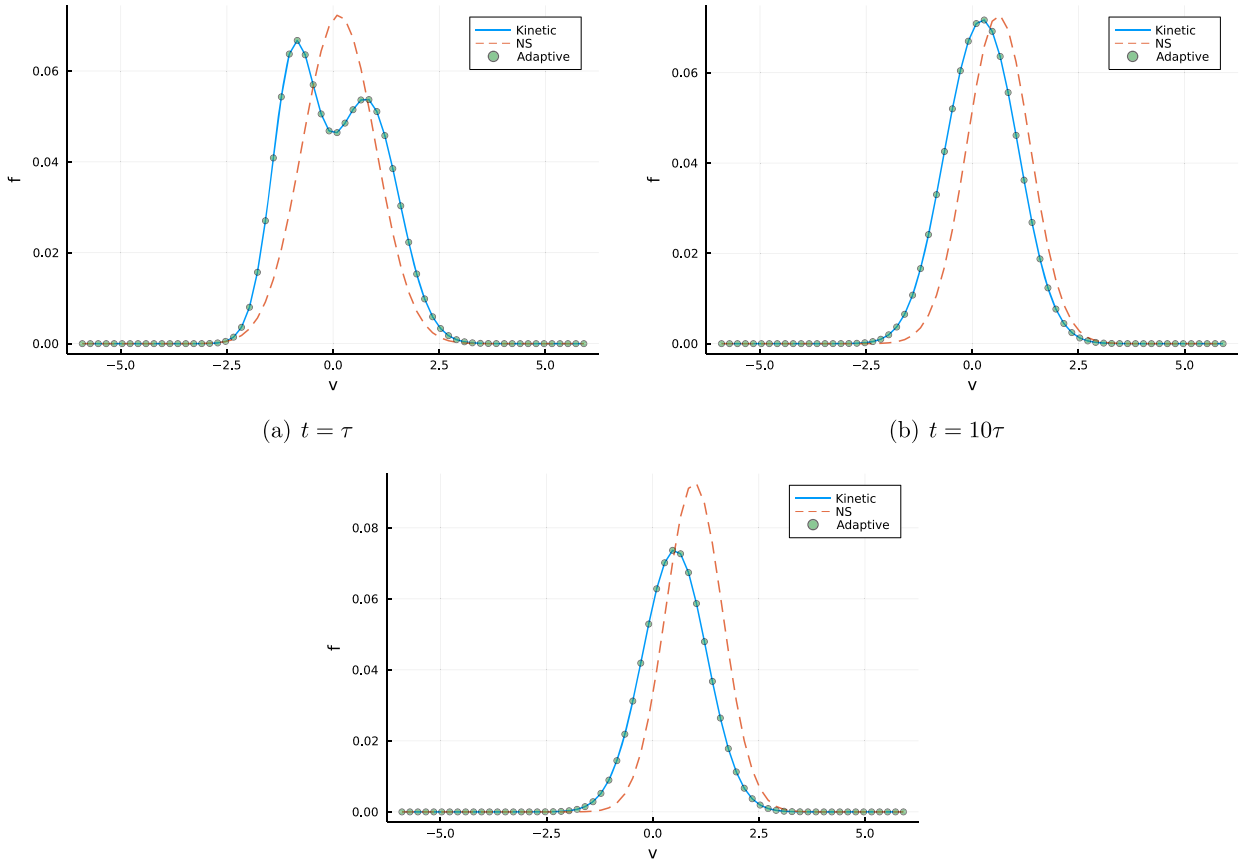
In the second numerical experiment, let us turn to a shear layer in the transition regime where the transverse processes dominate the fluid motion. The particle distribution function is initialized as Maxwellian, which corresponds to the following macroscopic variables,

$$\begin{pmatrix} \rho \\ V_x \\ V_y \\ T \end{pmatrix}_{t=0,L} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} \rho \\ V_x \\ V_y \\ T \end{pmatrix}_{t=0,R} = \begin{pmatrix} 1 \\ 0 \\ -1 \\ 0.5 \end{pmatrix}.$$

The simulation is performed till $50\tau_0$, where $\tau_0 = \mu_0/p_0$ denotes the mean collision time in the left half of the initial domain, and the viscosity μ_0 can be evaluated from the hard-sphere model,

$$\mu_0 = \frac{15\sqrt{\pi}}{48}\mathrm{Kn}.$$

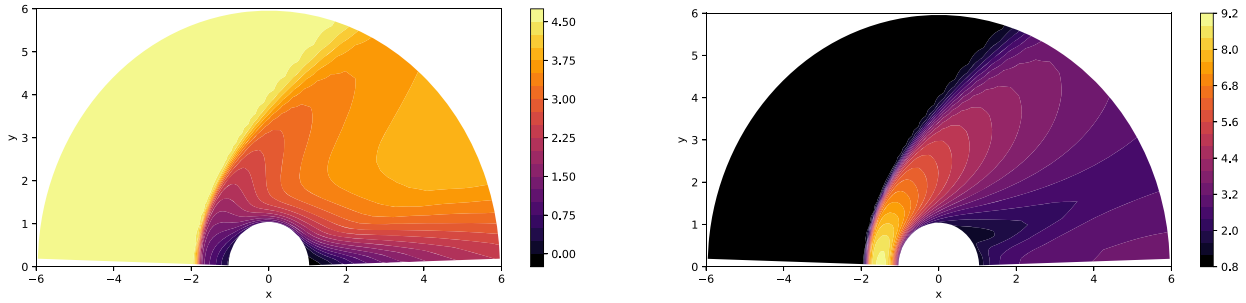
The detailed computation setup is listed in Table 3.



(a) $t = \tau$ (b) $t = 10\tau$

(c) $t = 50\tau$

Fig. 15. Particle distribution functions at the domain center at different time instants.

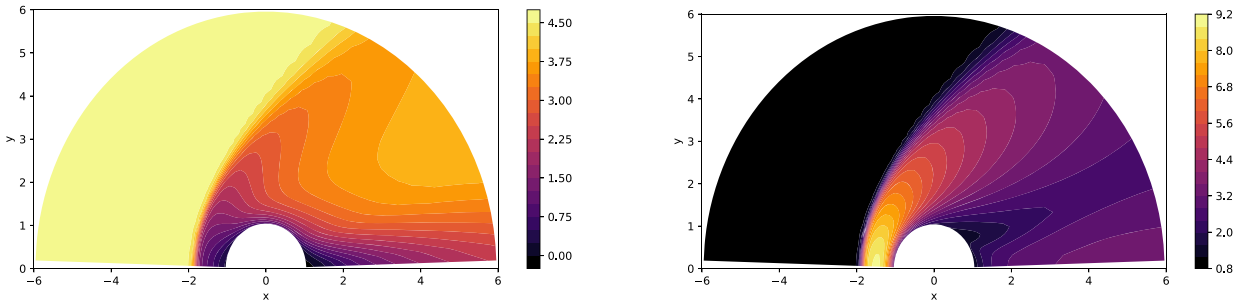


(a) U-velocity (b) Temperature

Fig. 16. Profiles of density and temperature in the cylinder flow under Kn = 0.001. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

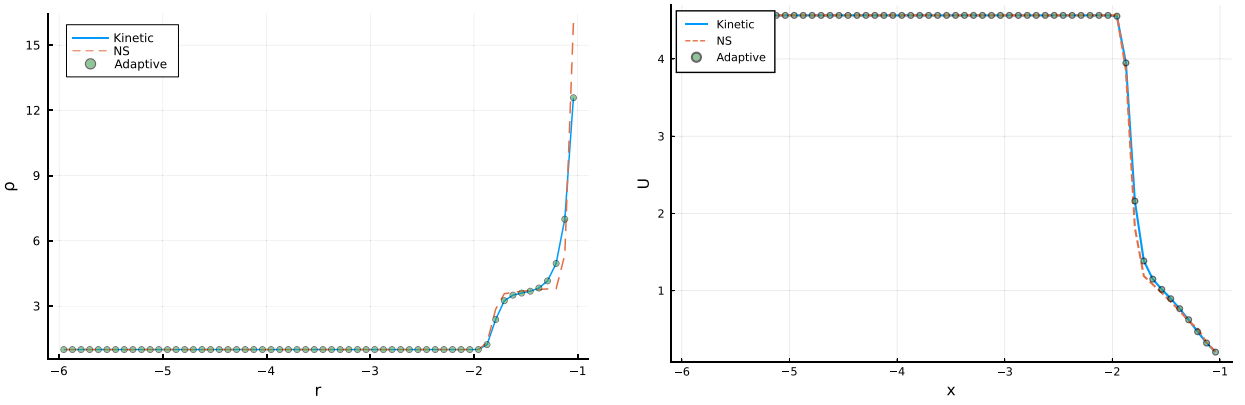
We first conduct a fully kinetic simulation with the Boltzmann equation. Based on the kinetic solution, the partition of flow regimes based on different criteria is shown in Fig. 11. With time evolution, it is clear that the non-equilibrium region expands due to the strong shearing effect. The neural network predicts equivalent flow regimes as ground truth, while the gradient-length-local Knudsen number criterion underestimates the non-equilibrium effect.

Then we employ the adaptive solver to conduct the simulation. The profiles of density, velocity, and temperature at different time instants are presented in Fig. 12, 13, and 14. The kinetic and Navier-Stokes solutions are plotted as a benchmark. As is shown, for this highly dissipative problem with a strong shearing effect, the kinetic and Navier-Stokes equations present distinct solutions. Fig. 15 presents the evolution of the particle distribution function at the domain center. Due to the accumulating effect of intermolecular collisions, the particle distribution function transforms gradually into Maxwellian from the initial bi-modal distribution. During the evolution process, the adaptive scheme provides equivalent solutions as

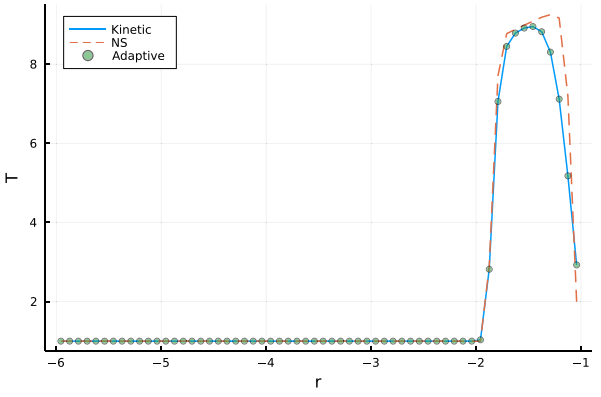


(a) U-velocity (b) Temperature

Fig. 17. Profiles of density and temperature in the cylinder flow under Kn = 0.01.



(a) Density (b) U-velocity

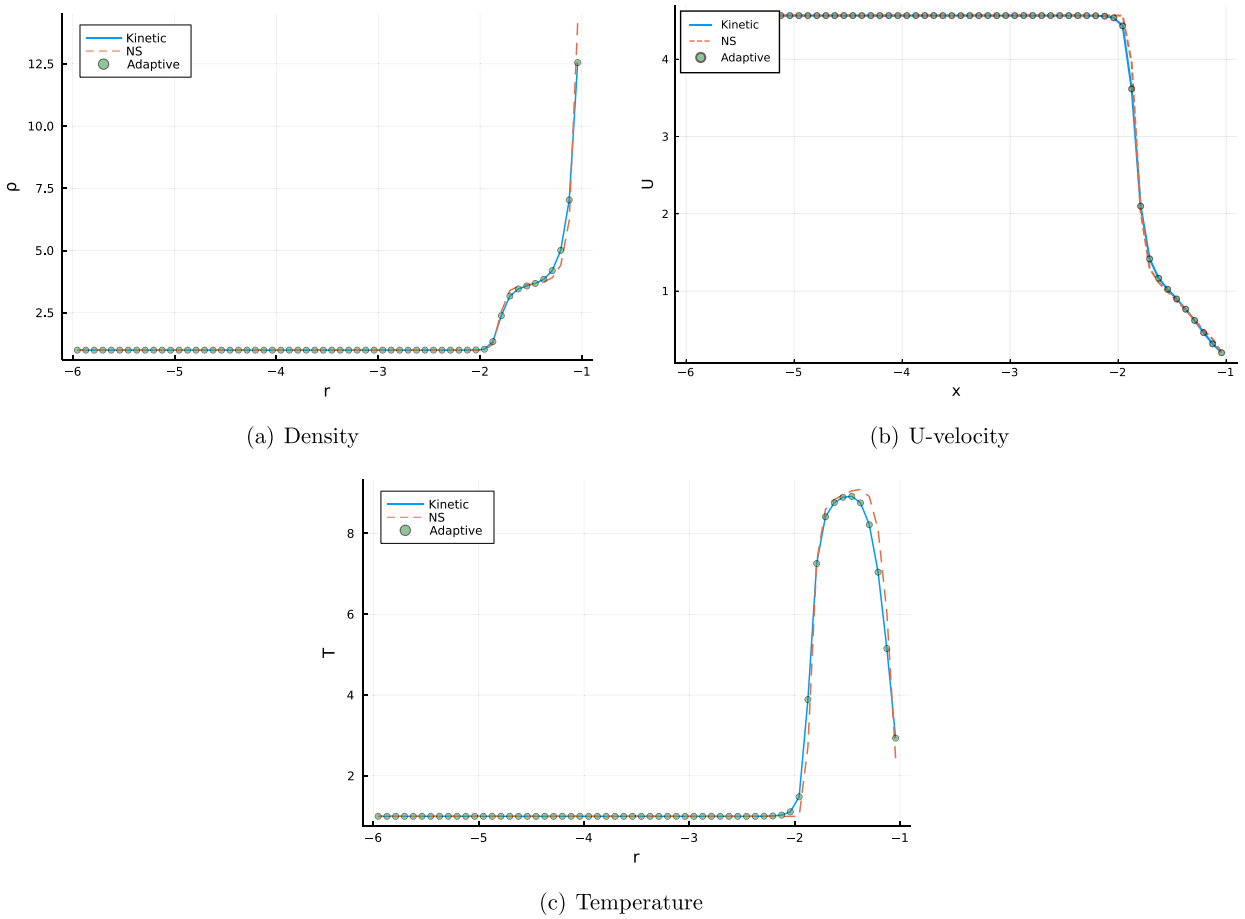


(c) Temperature

Fig. 18. Solutions along the horizontal center line in front of the cylinder at Kn = 0.001.

Table 4
Computational cost of shear layer problem.

	time	total allocations	total allocated memory
Navier-Stokes	11.07 s	7.87×10^7	5.63 GB
Kinetic	1985.34 s	4.00×10^8	15.39 TB
Adaptive	623.05 s	1.35×10^8	1.36 TB



(a) Density (b) U-velocity

(c) Temperature

Fig. 19. Solutions along the horizontal central line in front of the cylinder at Kn = 0.01.

Table 5
Computational setup of flow around a circular cylinder.

r	N_r	θ	N_θ	$\mathbf{v}$	N_u	N_v
$[1, 6]$	60	$[0, \pi]$	50	$[-10, 10]^3$	48	48
N_w	Quadrature	Kn	CFL	Integrator	Wall	Edge
32	Rectangular	[0.001, 0.01]	0.5	Euler	Maxwell	Symmetry

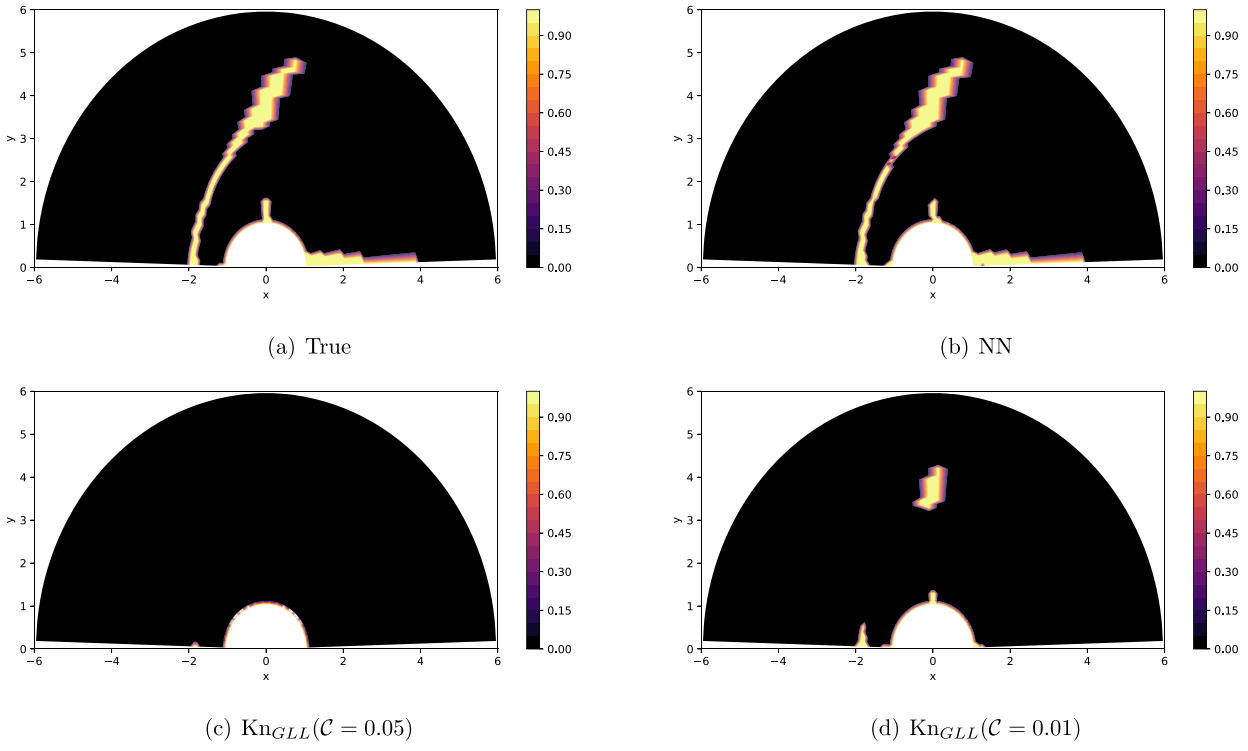
the kinetic benchmark, which confirms the validity of the neural network classifier. Table 4 provides the computational cost of all these three solvers. As can be seen, the adaptive scheme accelerates the simulation by 69% and saves 66% unnecessary allocations.

6.3. Flow around circular cylinder

In the last numerical experiment, we present the two-dimensional hypersonic flow around a circular cylinder, where longitudinal and transverse processes coexist in the domain. The particle distribution function is initialized as Maxwellian everywhere corresponding to the Mach number $Ma = 5$. The detailed computation setup is listed in Table 5.

In this steady-state problem, the computation can be accelerated with the help of the NS solver. A convergent coarse flow field can be first obtained by the NS solver, and then reconstructed as the initial state in the subsequent adaptive method. The workflow for the computation of steady flow is described as follows.

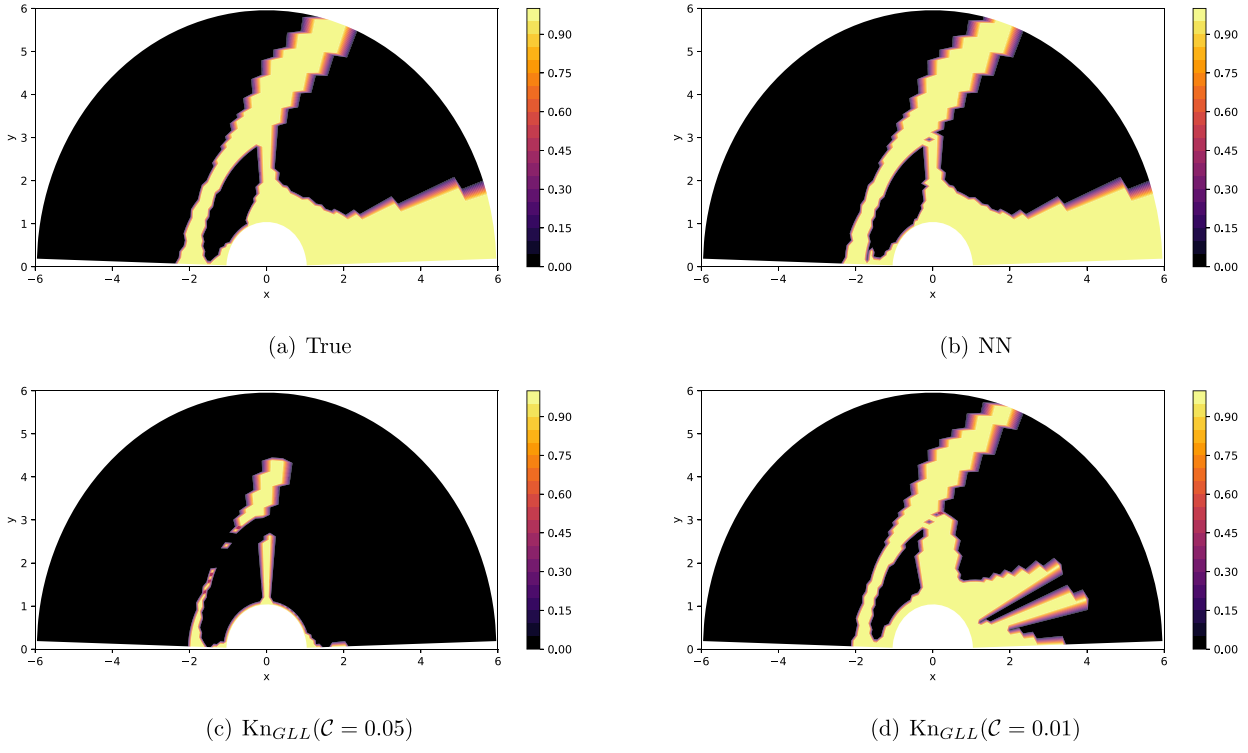
Fig. 16 and 17 present the contours of U-velocity and temperature produced by the adaptive solver at Kn = 0.001 and 0.01. As shown, the bow shock and the expansion cooling region behind the cylinder are well captured. Fig. 18 and 19 present the quantitative comparison of solutions produced by the kinetic, NS, and the current adaptive solver respectively. At Kn = 0.001, the cell size and time step in the computation are much larger than the particle mean free path and collision time, and all three methods deduce to the shock-capturing scheme. When the reference Knudsen number gets to Kn = 0.01, a larger particle mean free path leads to a wide shock structure. Due to the non-equilibrium gas dynamics in the shock



(a) True (b) NN

(c) $\mathrm{Kn}_{GLL}(\mathcal{C} = 0.05)$ (d) $\mathrm{Kn}_{GLL}(\mathcal{C} = 0.01)$

Fig. 20. Prediction of flow regimes at convergent state in the circular cylinder flow with different criteria under $\mathrm{Kn} = 0.001$ (0 denotes near-equilibrium, 1 denotes non-equilibrium).



(a) True (b) NN

(c) $\mathrm{Kn}_{GLL}(\mathcal{C} = 0.05)$ (d) $\mathrm{Kn}_{GLL}(\mathcal{C} = 0.01)$

Fig. 21. Prediction of flow regimes at convergent state in the circular cylinder flow with different criteria under $\mathrm{Kn} = 0.01$ (0 denotes near-equilibrium, 1 denotes non-equilibrium).

Algorithm 3: Workflow of steady flow problems.

Converge the flow-field with the Navier-Stokes solver
Classify cell-wise the flow-regimes
Reconstruct the solution of kinetic cells
Converge the flow-field with the adaptive solver

Table 6
Nomenclature.

Kn	Knudsen number
f	particle distribution function
Q	collision operator in the Boltzmann equation
ψ	collision invariants
$\mathcal{M}$	Maxwellian distribution function
k	Boltzmann constant
$\mathbf{W}$	macroscopic conservative variables
ρ	density
$\mathbf{V}$	bulk velocity
T	temperature
R	gas constant
μ	viscosity coefficient
κ	heat conductivity coefficient
$\mathbf{c}$	peculiar velocity
$\mathbf{I}$	identity tensor
ω	power index of hard-sphere model
θ	parameters of neural network
$\mathbf{U}$	input of neural network
$\hat{\mathcal{R}}$	output of neural network
$\mathcal{R}$	ground-truth label
$\mathcal{L}$	loss function
F	sampling space of f
$\mathbf{u}$	moment variables
$\mathbf{m}$	moment basis
$\boldsymbol{\alpha}$	Lagrange multipliers of dual problem
$\mathscr{R}$	realizable set of $\mathbf{u}$
H	Hessian of the dual problem
$\mathbf{n}$	unit normal vector
H	Heaviside step function
$\mathbf{\Omega}_i(\mathbf{x})$	control volume of physical space
$\mathbf{\Omega}_j(\mathbf{v})$	control volume of velocity space
$\mathbf{F}$	numerical flux
S	surface area
$\mathcal{E}$	equilibrium distribution function
$\mathbf{a}$	spatial derivatives of particle distribution function
b	time derivatives of particle distribution function

wave and gas-surface interaction, a slight difference can be observed in the solutions provided by kinetic and NS solvers, where the continuum scheme provides a narrower shock profile than the kinetic solution. The current adaptive method provides equivalent solutions as the kinetic benchmark, which confirms the validity of the neural network classifier in a two-dimensional case. Based on the convergent solution, the partition of flow regimes based on different criteria is shown in Fig. 20 and 21. Note that different critical values $\mathcal{C}$ are tested for the gradient-length-local Knudsen number. For the commonly adopted value $\mathcal{C} = 0.05$, Kn_{GLL} underestimates the non-equilibrium effect and makes inaccurate predictions. After we reset it as $\mathcal{C} = 0.01$, the predictions are still not precise enough. On the contrary, the neural network predicts more accurate flow regimes as the benchmark under different Knudsen numbers.

7. Conclusion

Gaseous flow is intrinsically a cross-scale problem due to the possible large variations of density and local Knudsen number. A quantitative criterion of continuum breakdown is crucial for developing sound flow theories and multi-scale solution algorithms. In this paper, we have built the first neural network for the binary classification of near-equilibrium and non-equilibrium flow regimes. This data-driven surrogate model provides an alternative to classical semi-empirical criteria and shows superiority in numerical experiments. Based on the minimal entropy closure of the Boltzmann moment system, an algorithmic strategy is designed to generate a dataset with a balanced distribution near and out of equilibrium state for model training and testing. A hybrid Boltzmann-Navier-Stokes flow solver is developed, which can dynamically adapt to local flow regimes using the neural network classifier. The current method provides an accurate and efficient tool for the study of cross-scale and non-equilibrium flow phenomena. It shows the potential to be extended to other complex systems, such as multi-component flows [44] and plasma physics [45].

CRediT authorship contribution statement

Tianbai Xiao: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Visualization, Writing – original draft, Writing – review & editing. **Steffen Schotthöfer:** Conceptualization, Formal analysis, Investigation, Methodology, Resources, Software, Visualization, Writing – original draft, Writing – review & editing. **Martin Frank:** Conceptualization, Formal analysis, Funding acquisition, Methodology, Project administration, Resources, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that there is no conflict of interest to this work.

Data availability

Data will be made available on request.

References

[1] Jonas Kusch, Steffen Schotthöfer, Pia Stammer, Jannick Wolters, Tianbai Xiao, Kit-rt: An Extendable Framework for Radiative Transfer and Therapy, 2022.
[2] Tianbai Xiao, Kinetic.jl: a portable finite volume toolbox for scientific and neural computing, J. Open Sour. Softw. 6 (62) (2021) 3060.
[3] Hsue-Shen Tsien, Superaerodynamics, mechanics of rarefied gases, J. Aeronaut. Sci. 13 (12) (1946) 653–664.
[4] Sydney Chapman, Thomas George Cowling, David Burnett, The Mathematical Theory of Non-uniform Gases: an Account of the Kinetic Theory of Viscosity, Thermal Conduction and Diffusion in Gases, Cambridge University Press, 1990.
[5] Graeme Austin Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Clarendon, 1994.
[6] Iain D. Boyd, Gang Chen, Graham V. Candler, Predicting failure of the continuum fluid equations in transitional hypersonic flows, Phys. Fluids 7 (1) (1995) 210–219.
[7] Quanhua Sun, Iain D. Boyd, Graham V. Candler, A hybrid continuum/particle approach for modeling subsonic, rarefied gas flows, J. Comput. Phys. 194 (1) (2004) 256–277.
[8] Alejandro L. Garcia, Berni J. Alder, Generation of the Chapman–Enskog distribution, J. Comput. Phys. 140 (1) (1998) 66–70.
[9] C. David Levermore, William J. Morokoff, B.T. Nadiga, Moment realizability and the validity of the Navier–Stokes equations for rarefied gas dynamics, Phys. Fluids 10 (12) (1998) 3214–3226.
[10] Maziar Raissi, Alireza Yazdani, George Em Karniadakis, Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations, Science 367 (6481) (2020) 1026–1030.
[11] Yingzhou Li, Jianfeng Lu, Anqi Mao, Variational training of neural network approximations of solution maps for physical models, J. Comput. Phys. 409 (2020) 109338.
[12] Yuehaw Khoo, Lexing Ying, Switchnet: a neural network model for forward and inverse scattering problems, SIAM J. Sci. Comput. 41 (5) (2019) A3182–A3201.
[13] Maziar Raissi, Paris Perdikaris, George E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707.
[14] Zhiping Mao, Ameya D. Jagtap, George Em Karniadakis, Physics-informed neural networks for high-speed flows, Comput. Methods Appl. Mech. Eng. 360 (2020) 112789.
[15] Luning Sun, Han Gao, Shaowu Pan, Jian-Xun Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, Comput. Methods Appl. Mech. Eng. 361 (2020) 112732.
[16] Steven L. Brunton, Joshua L. Proctor, J. Nathan Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. USA 113 (15) (2016) 3932–3937.
[17] Samuel H. Rudy, Steven L. Brunton, Joshua L. Proctor, J. Nathan Kutz, Data-driven discovery of partial differential equations, Sci. Adv. 3 (4) (2017) e1602614.
[18] Jun Zhang, Wenjun Ma, Data-driven discovery of governing equations for fluid dynamics based on molecular simulation, J. Fluid Mech. 892 (2020).
[19] Tianbai Xiao, Martin Frank, Using neural networks to accelerate the solution of the Boltzmann equation, J. Comput. Phys. (2021) 110521.
[20] Steffen Schotthöfer, Tianbai Xiao, Martin Frank, Cory D. Hauck, A structure-preserving surrogate model for the closure of the moment system of the Boltzmann equation using convex deep neural networks, arXiv preprint, arXiv:2106.09445, 2021.
[21] Qin Lou, Xuhui Meng, George Em Karniadakis, Physics-informed neural networks for solving forward and inverse flow problems via the Boltzmann-bgk formulation, J. Comput. Phys. (2021) 110676.
[22] François Bouchut, François Golse, Mario Pulvirenti, Kinetic Equations and Asymptotic Theory, Elsevier, 2000.
[23] C. Levermore, Moment closure hierarchies for kinetic theories, J. Stat. Phys. 83 (1996) 1021–1065.
[24] N. Mikhail Kogan, Rarefied Gas Dynamics, Plenum Press, 1969.
[25] Balázs Csanád Csáji, et al., Approximation with artificial neural networks, Fac. Sci., Etvs Lornd Univ., Hungary 24 (48) (2001) 7.
[26] Louis De Branges, The Stone-Weierstrass theorem, Proc. Am. Math. Soc. 10 (5) (1959) 822–824.
[27] Steffen Schotthöfer, Tianbai Xiao, Martin Frank, Cory Hauck, Structure preserving neural networks: a case study in the entropy closure of the Boltzmann equation, in: Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, Sivan Sabato (Eds.), Proceedings of the 39th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 162, PMLR, 17–23 Jul 2022, pp. 19406–19433.
[28] Steffen Schotthöfer, Tianbai Xiao, Martin Frank, Cory D. Hauck, A Structure-Preserving Surrogate Model for the Closure of the Moment System of the Boltzmann Equation Using Convex Deep Neural Networks, 2021.
[29] Michael Junk, Andreas Unterreiter, Maximum entropy moment systems and Galilean invariance, Contin. Mech. Thermodyn. 14 (2002) 563–576.
[30] Michael Junk, Maximum entropy for reduced moment problems, Math. Models Methods Appl. Sci. 10 (07) (2000) 1001–1025.
[31] C. Hauck, C.D. Levermore, A. Tits, Convex duality and entropy-based moment closures: characterizing degenerate densities, in: 2008 47th IEEE Conference on Decision and Control, 2008, pp. 5092–5097.
[32] C. David Levermore, Entropy-based moment closures for kinetic equations, Transp. Theory Stat. Phys. 26 (4–5) (1997) 591–606.
[33] Raul E. Curto, Lawrence A. Fialkow, Recursiveness, positivity, and truncated moment problems, Houst. J. Math. 17 (4) (1991) 603–635.
[34] Michael Junk, Domain of definition of Levermore's five-moment system, J. Stat. Phys. 93 (5–6) (December 1998) 1143–1167.

[35] V. Pavan, General entropic approximations for canonical systems described by kinetic equations, J. Stat. Phys. 142 (2011) 792–827.
[36] Steffen Schotthöfer, Tianbai Xiao, Martin Frank, Cory D. Hauck, Neural network-based, structure-preserving entropy closures for the Boltzmann moment system, arXiv preprint, arXiv:2201.10364, 2021.
[37] Clément Mouhot, Lorenzo Pareschi, Fast algorithms for computing the Boltzmann collision operator, Math. Comput. 75 (256) (2006) 1833–1852.
[38] Prabhu Lal Bhatnagar, Eugene P. Gross, Max Krook, A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems, Phys. Rev. 94 (3) (1954) 511.
[39] E.M. Shakhov, Generalization of the Krook kinetic relaxation equation, Fluid Dyn. 3 (5) (1968) 95–96.
[40] H. Lowell Holway Jr., New statistical models for kinetic theory: methods of construction, Phys. Fluids (1958–1988) 9 (9) (1966) 1658–1673.
[41] Taku Ohwada, Kun Xu, The kinetic scheme for the full-Burnett equations, J. Comput. Phys. 201 (1) (2004) 315–332.
[42] Kun Xu, Direct Modeling for Computational Fluid Dynamics: Construction and Application of Unified Gas-Kinetic Schemes, World Scientific, 2015.
[43] Tianbai Xiao, Chang Liu, Kun Xu, Qingdong Cai, A velocity-space adaptive unified gas kinetic scheme for continuum and rarefied flows, J. Comput. Phys. 415 (2020) 109535.
[44] Tianbai Xiao, Kun Xu, Qingdong Cai, A unified gas-kinetic scheme for multiscale and multicomponent flow transport, Appl. Math. Mech. 40 (3) (2019) 355–372.
[45] Tianbai Xiao, Martin Frank, A stochastic kinetic scheme for multi-scale plasma transport with uncertainty quantification, J. Comput. Phys. 432 (2021) 110139.