Topology optimization for rarefied gas flow problems using density method and adjoint DSMC

Kaiwen Guan[a], Yuki Noguchi[a,b,*], Kei Matsushima[a], Takayuki Yamada[a,b]

[a] *Department of Mechanical Engineering, Graduate School of Engineering, The University of Tokyo, Yayoi 2-11-16, Bunkyo-ku, Tokyo 113-8656, Japan.*
[b] *Department of Strategic Studies, Institute of Engineering Innovation, Graduate School of Engineering, The University of Tokyo, Yayoi 2-11-16, Bunkyo-ku, Tokyo 113-8656, Japan.*

Abstract

In this paper, a topology optimization method based on direct simulation Monte Carlo (DSMC) is presented for rarefied gas flows. Distribution of fluid and solid in the design domain is characterized by a pseudo density. The traditional DSMC algorithm is extended to include the pseudo density by interpreting solid as an imaginary gas with fixed temperature and zero macroscopic velocity. Treating the pseudo density as the design variable, design sensitivity is obtained through the Lagrangian multiplier method and the adjoint state method. A discrete version of the adjoint equations is used, so that the adjoint variables can be evaluated using the information stored during the forward DSMC process. The information preservation (IP) method for subsonic flow is also considered in order to reduce random noise and simplify the discrete adjoint equations. The extended DSMC algorithm is verified by numerical examples, and optimized design for bent pipe is also included.

Keywords: Topology optimization, Rarefied gas flow, Direct simulation Monte Carlo method, Density method, Adjoint equations, Sensitivity analysis
2010 MSC: 00-01, 99-00

1. Introduction

In rarefied gas flows, the mean free path of the gas molecules, which is the average distance travelled by the molecules between intermolecular collisions, is not negligible compared to the characteristic length of the flow field. Under rarefied conditions, gas cannot be sufficiently described by the Navier-Stokes (NS) equation, and interesting phenomenon such as velocity slip occurs [1]. Study of rarefied gas flows plays an important role in design and analysis of spacecrafts [2, 3] and micro-electro-mechanical systems (MEMS) [4, 5].

*Corresponding author
Email address: noguchi@mech.t.u-tokyo.ac.jp (Yuki Noguchi)

To improve the performance of devices in rarefied gas flows, topology optimization is an effective approach. Developed mainly for design problems in solid mechanics, topology optimization has achieved great success since then [6, 7]. Compared to traditional optimization methods, namely, sizing optimization and shape optimization, topology optimization has the highest degree of freedom, and allows changes in topological properties of the structure. In topology optimization, structures are represented by the distribution of materials. Bendsøe and Kikuchi first proposed the homogenization design method to address material distribution in optimization [8]. The density method, which is developed later [9, 10], is another popular method. In density method, the design variable is chosen to be a normalized density which is continuous and convex. Apart from this, the use of level-set function to express material boundary as an iso-surface is also studied [11, 12].

The development of topology optimization motivated researchers to expand its domain of influence. Problems in other fields such as heat transfer [13], electromagnetics [14], and fluid dynamics are also addressed. In fluid dynamics, optimization of flow using the NS equations has been studied since the pioneering work by Borrvall and Peterrson [15]. They used density method to represent the distribution of solid and fluid, and interpreted the grayscale region as porous medium with varying permeability. This expressions satisfies non-slip boundary condition on solid-fluid interfaces implicitly. Based on their work, topology optimization for steady [16] and unsteady [17] NS flows has been proposed.

For optimization problems in fluid dynamics, the above-mentioned studies obtained the flow fields by solving the NS equation using finite element methods. While for rarefied gas flows, the governing equation is the Boltzmann equation instead, which describes gas behavior by a distribution function. Despite being very fundamental, the Boltzmann equation is difficult to solve numerically, as it includes a highly non-linear collision term. The direct simulation Monte Carlo (DSMC) method is one of the most popular numerical schemes developed to solve the Boltzmann equation [18]. In DSMC, a large number of discrete molecules are used to represent the distribution of real gas molecules, and proper collision pairs are selected randomly while the molecules move in the computational domain [19]. Despite the success of DSMC in approximating the Boltzmann equation, it is rarely used in topology optimization of rarefied gas flows because strictly speaking, it is not a deterministic process. Randomness is intentionally introduced during the simulation in order to resemble the behavior of real gas molecules. Despite being crucial to a correct simulation, randomness poses a challenge to sensitivity analysis, which evaluates the influence of design variable on the objective function that is dependent on the solution of the governing equation. As a result, deterministic numerical methods such as the lattice Boltzmann method (LBM) is often used in topology optimization instead [20, 21, 22, 23].

In 2021, Caflisch et al. proposed an inverse adjoint DSMC process in his paper [24], which throws new light on the challenge. Following a discretize-then-optimize scheme, the molecules in DSMC calculation are treated as discrete state variables, and discrete adjoint variables are assumed based on the

Lagrangian method. Recording the details of forward DSMC process leads to the fomulation of a set of adjoint equations, which generate the solutions of the adjoint variables. Design sensitivity obtained from the adjoint variables can be viewed as the total derivative of the objective function with respect to the design variable, and can be used for numerical optimization. In Caflisch's paper, only the spatially homogeneous Boltzmann equation is treated. In this paper, we shall extend the analysis to spatially inhomogeneous cases, and construct a topology optimization algorithm for general rarefied gas flows based on the adjoint DSMC method using density method.

The outline of this paper is as follows. First, the traditional DSMC algorithm, and the IP method that is used in the study to reduce random scatter will be introduced. Then a density representation of material distribution is set up, based on which the DSMC algorithm is extended to describe rarefied gas flow in the design domain. The design problem is formulated, and the discrete adjoint equations, as well as the design sensitivity, are obtained. Finally, several numerical examples are included. Namely, an analysis of two-dimensional Couette flow is used to verify the proposed density model, analysis of flow through an orifice is used to verify the design sensitivity, and finally, the optimization algorithm is applied to design of a bent pipe.

2. Solution to Boltzmann equation using DSMC

2.1. Boltzmann equation

For single-component rarefied gas flows, the governing equation is the Boltzmann equation,

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = W(f). \tag{1}$$

$f(x, v, t)$ is the distribution function for a single particle. It describes the probability of finding one specific gas molecule at position x with velocity v at time t. F is the external force per unit mass acting on the gas, and $W(f)$ is the collision term describing the change in f due to binary collisions between gas molecules. W is defined by

$$W(f) = \int_{\mathbb{R}^3} \int_{\mathcal{S}^2} q(v - v_1, \sigma)(f^* f_1^* - f f_1) d\sigma dv_1, \tag{2}$$

where $\mathcal{S}^2$ is the unit sphere, $q(v - v_1, \sigma)$ is the collision section. f_1, f^*, f_1^* are short-hand notations for $f(x, v_1, t), f(x, v^*, t), f(x, v_1^*, t)$ respectively, where v^* and v_1^* are after-collision velocity for v and v_1. $q(v - v_1, \sigma)$ characterizes the likelihood of collision between molecules with relative velocity $v - v_1$ and deflection in the direction of unit vector σ. Detailed expressions for q depend on the molecule model one chooses [1, 25].

2.2. Direct simulation Monte Carlo method

In this section, the basic DSMC algorithm used in this paper, which is based on book by Shen [1], is briefly explained. In DSMC, the continuous distribution function f is represented by a set of discrete molecules $S = \{s_1, s_2, \cdots, s_N\}$, where N is the total number of molecules. Each molecule s_i is defined by its position x_i and velocity v_i, which are vectors in $\mathbb{R}^3$. During the simulation, time is divided into small steps Δt, and the set of molecules evolve in an Eulerian way.

The key idea of DSMC is that for sufficiently small time intervals, molecule motion and molecule collisions can be decoupled. For general rarefied gas flows, each iteration in DSMC consists three steps, essentially.

- Update the positions of molecules according to their velocities. Namely, for $1 \leq i \leq N$,
$$x_i' = x_i + v_i \Delta t. \tag{3}$$
Reflection at gas-solid interface is performed when the molecule attempts to enter solid domain. For diffuse solid boundaries, after-reflection velocity of the molecules are decided in the following way:
$$v_i' = (v_{\text{wall}} w_1 \sin\varphi)\hat{t}_1 + (v_{\text{wall}} w_1 \cos\varphi)\hat{t}_2 + (v_{wall} w_2)\hat{n}. \tag{4}$$
In this expression, $v_{\text{wall}} = \sqrt{2\overline{k}T_{\text{wall}}/\overline{m}}$ is the most probable thermal speed at wall temperature T_{wall}, where $\overline{k}$ is the Boltzmann constant, and $\overline{m}$ is molecular mass. w_1 and w_2 are two random values generated by $w_1 = w_2 = \sqrt{-\ln \text{Rand}}$, where Rand is a uniformly distributed random number between 0 and 1. φ is a random angle generated by $\varphi = 2\pi\text{Rand}$. $\hat{n}$ is the unit normal vector pointing into gas domain. $\hat{t}_1$ and $\hat{t}_2$ are two perpendicular unit vectors that are tangent to the solid surface.

  Assume the molecule travelled for time $\Delta t'$ before reflection, then after calculating the reflected velocity v_i', the updated position is
$$x_i' = x_i + v_i \Delta t' + v_i'(\Delta t - \Delta t'). \tag{5}$$

- Update the velocities of the molecules according to external forces. Namely, for $1 \leq i \leq N$,
$$v_i' = v_i + F(x_i)\Delta t.$$

- Choose proper collision pairs and perform binary collisions. For each collision pair $\{s_i, s_j\}$, the velocities of the two molecules are updated according to
$$v_i' = \frac{v_i + v_j}{2} + \frac{|v_i - v_j|}{2}\hat{\sigma}.$$
$$v_j' = \frac{v_i + v_j}{2} - \frac{|v_i - v_j|}{2}\hat{\sigma}.$$
Where $\hat{\sigma}$ is a randomly generated unit vector indicating the direction of deflection.

Once the iterations are completed, flow properties can be obtained from the distribution of molecules in their positions and velocities. For steady flows, usually a number of steps are sampled, and flow properties are taken as the average value over time.

2.3. IP method for DSMC

One inherent problem with DSMC is that flow properties obtained from molecule distribution contain random scatter. For steady flows, one way of reducing the uncertainty is to include a longer sampling range after the flow reaches steady state, and calculate the time average for flow properties. However, the effect of this method is limited, as reducing uncertainty to $1/N$ requires N^2 times the number of molecules sampled. Computational cost will be too high if we want to reduce uncertainty significantly.

To tackle this problem, Fan and Shen [26, 27] proposed the IP modification for DSMC, which is further developed by Sun and Boyd [28]. In the IP method, each molecule s_i is assigned additional information properties, namely, information velocity $V_i \in \mathbb{R}^3$ and information temperature $T_i \in \mathbb{R}$. During the iterations, motion, reflection, and sampling of collision pairs are the same as in the previous section. The information variables are updated along with the original ones in the following way:

- Preserved information is unchanged during molecule motion. However, if a molecule i is reflected about a diffuse surface,

$$V_i = 0, \tag{6}$$
$$T_i = T_{\text{wall}}. \tag{7}$$

- For external forces, information velocity is also updated,

$$V_i' = V_i + F(x_i)\Delta t. \tag{8}$$

- For collision between s_i and s_j, information velocity and information temperature are updated by

$$V_i' = \frac{1 + C_\mu \cos\theta}{2} V_i + \frac{1 - C_\mu \cos\theta}{2} V_j, \tag{9}$$
$$V_j' = \frac{1 - C_\mu \cos\theta}{2} V_i + \frac{1 + C_\mu \cos\theta}{2} V_j, \tag{10}$$
$$T_i' = \frac{1 + C_k \cos\theta}{2} T_i + \frac{1 - C_k \cos\theta}{2} T_j + \frac{|V_i - V_j|^2}{4\xi R}, \tag{11}$$
$$T_j' = \frac{1 - C_k \cos\theta}{2} T_i + \frac{1 + C_k \cos\theta}{2} T_j + (1 - C_\mu^2 \cos^2\theta) \frac{|V_i - V_j|^2}{4\xi R}. \tag{12}$$

  Where C_μ and C_k are gas-specific constants, ξ is the degree of freedom of the gas molecule (in this paper $\xi = 3$ is used), R is the specific gas constant, and θ is the deflection angle, namely, the angle between vector $v_i - v_j$ and $\hat{\sigma}$.

To include pressure effects, the following flow properties are calculated and stored in each computational cell: velocity V_f, temperature T_f, pressure P, and density ρ, where V_f is a vector in $\mathbb{R}^3$ and other are scalars. Flow properties modify the molecules in the three additional steps introduced to each iteration.

- Solve the two equations simultaneously that correspond to momentum and energy transfer concerning the information velocity and information temperature of the molecules.

$$\frac{\partial V}{\partial t} = -\frac{1}{\rho}\nabla_x P, \tag{13}$$

$$\frac{\partial}{\partial t}\left(\frac{|V|^2}{2} + \frac{\xi RT}{2}\right) = -\frac{1}{\rho}\nabla_x \cdot (V_f P). \tag{14}$$

- Solve the advection equation for mass transfer,

$$\frac{\partial \rho}{\partial t} = -\nabla_x \cdot (\rho V_f). \tag{15}$$

- Calculate flow properties by information properties of the molecules in the cell. Assume for cell n, there are N_n molecules, whose indexs are m_i, where $1 \leq i \leq N_n$,

$$V_{f,n} = \frac{1}{N_n}\sum_{i=1}^{N_n} V_{m_i}, \tag{16}$$

$$T_{f,n} = \frac{1}{N_n}\sum_{i=1}^{N_n} T_{m_i}, \tag{17}$$

$$P_n = \rho_n T_{f,n} R. \tag{18}$$

In practice, equations (13) to (15) are solved by finite-volume approximations. Using the IP method, flow properties are directly obtained during the calculation from the preserved information carried by each molecule. Once steady state is reached, taking a time average over a sampling range can further reduce random error.

2.4. Inlet and outlet boundary condition

To simulate low speed rarefied gas flows, it is important to introduce new molecules at the inlet and outlet in order to maintain the balance of number of molecules in the computational domain. Following the study of Liou and Fang [29, 30], the initial conditions of the introduced molecules are determined by the information of the flow at the cells that they enter. Use a two dimensional horizontal pipe as an example, where the inlet and outlet are effectively two vertical lines. At the inlet, pressure $P = P_{\text{in}}$ and temperature $T_f = T_{\text{in}}$ of the flow is fixed, plus the condition that macroscopic velocity of the flow in the

vertical direction is zero, $V_{f,2} = 0$. While at the outlet, only the pressure is given as $P = P_{\text{out}}$.

At the inlet, macroscopic velocity of the gas is obtained by extrapolating the flow velocity inside the computational domain. Hence for the introduced molecule with index i, the information velocity and information temperature are

$$V'_{i,1} = V_{f,1}, \tag{19}$$

$$V'_{i,2} = 0, \tag{20}$$

$$T'_i = T_{in}. \tag{21}$$

At the outlet, the information velocity and information temperature of the introduced molecule i are determined using the characteristics-theory-based equations,

$$V'_{i,1} = V_{f,1} + \frac{P - P_{\text{out}}}{\rho\sqrt{\gamma R T_f}}, \tag{22}$$

$$V'_{i,2} = V_{f,2}, \tag{23}$$

$$T'_i = \frac{P_{\text{out}}}{\rho R}, \tag{24}$$

where γ is the ratio of specific heats for the gas, and R is the specific gas constant.

The original velocities of the molecules are generated randomly using the acceptance-rejection method. The exact number of molecules introduced at each step is calculated using the Maxwellian distribution. The details can be found in [29, 30].

3. Optimization problem

3.1. Topology optimization

In topology optimization, the structure optimization problem is rephrased into a material distribution problem. In the design domain $\mathcal{D}$, a characteristic function χ is used to indicate the presence of fluid or solid. Let the fluid domain be Ω and solid domain be $\mathcal{D}\backslash\Omega$, then χ should be defined as

$$\chi(x) = \begin{cases} 1 & x \in \Omega, \\ 0 & x \in \mathcal{D}\backslash\Omega. \end{cases} \tag{25}$$

To make the topology optimization problem convex, density method is often used. The characteristic function χ is replaced by a normalized density $\alpha \in \mathcal{L}^\infty(\mathcal{D}; [0, 1])$, which is chosen as the design variable. Since α can take values other than 0 and 1, it suggests the existence of some intermediate state between fluid and solid. To adapt to this view, the DSMC algorithm need to be extended.

3.2. Extension of DSMC

In traditional DSMC calculations, the boundary between solid domain and fluid (gas) domain is explicitly expressed. Molecules move in the fluid domain only, and they are reflected when they attempt to cross the gas-solid interface. No molecules are present in the solid domain. However, in topology optimization, the distinction between gas and solid is blurred by the introduction of grayscale regions, which means molecules are free to travel in the entire computational domain. However, interpreting the grayscale region as a porous medium where the pseudo density α is related to material permeability means some modifications need to be applied to the molecules.

By definition, when $\alpha = 0$, the medium is pure solid with zero permeability. Macroscopic velocity of the flow should be zero, and temperature should be equal to wall temperature T_{wall}. When $\alpha = 1$, the medium is pure gas with infinite permeability, flow properties should be the same as they are in traditional DSMC calculations. Besides, the relation between pseudo density α and permeability is expected to be continuous and monotonous. Following these criteria, we propose the following extension of DSMC.

Add another correction step where macroscopic velocities are dampened and temperature is relaxed towards wall temperature. Information velocity and temperature are corrected by

$$V' = \alpha^p V, \tag{26}$$

$$T' = \alpha^p T + (1 - \alpha^p) T_{wall}, \tag{27}$$

where p is a positive number.

For correction of true velocity v for the molecules, first the macroscopic velocity and temperature in all directions are calculated in the cell. Assuming at the correction step, there are N_n molecules in the cell, then the macroscopic velocity and temperature in each direction are calculated in the cell,

$$\overline{v}_i = \frac{1}{N_n} \sum_{j=1}^{N_n} v_{j,i}. \tag{28}$$

$$\overline{T}_i = \frac{\overline{m}}{\overline{k} N_n} \sum_{j=1}^{N_n} (\overline{v}_i - v_{j,i})^2. \tag{29}$$

A portion of the macroscopic velocity is first subtracted from the true velocities, and the temperature is rescaled.

$$v'_{j,i} = v_{j,i} - (1 - \alpha^p) \overline{v}_i. \tag{30}$$

$$v''_{j,i} = \alpha^p \overline{v}_i + (1 - \alpha^p)(v'_{j,i} - \overline{v}_i) \sqrt{\frac{T_{\text{wall}}}{\overline{T}_i}}. \tag{31}$$

Apart from this additional step, existing steps in the IP method need to be modified as well. Equations used to include pressure effects, namely equation

(13) to (15), are based on rarefied gas flow in pure fluid domain. However, in porous medium, mass, momentum, and energy transfer is hindered. Equations (13) to (15) are modified as follows to include the pseudo density α

$$\frac{\partial V}{\partial t} = -\alpha^{p'} \frac{1}{\rho} \nabla_x P, \tag{32}$$

$$\frac{\partial}{\partial t}\left(\frac{|V|^2}{2} + \frac{\xi RT}{2}\right) = -\alpha^{p'} \frac{1}{\rho} \nabla_x \cdot (PV_f), \tag{33}$$

$$\frac{\partial \rho}{\partial t} = -\alpha^{p'} \nabla_x \cdot (\rho V_f), \tag{34}$$

where p' is a positive number.

In the modified equations, when $\alpha = 1$, equations (32) to (34) are the same with equations (13) to (15). When $\alpha = 0$, equations (32) to (34) simply mean there is no transfer in solid domain. Similar to the density method in optimization of solid structures [6], the exponents $p, p' \geq 1$ help convexify the optimization problem, and penalizes against grayscale region. In this paper, their values are chosen as $p = 1$ and $p' = 2$.

In DSMC, equations (32) to (34) are usually solved using the finite volume method. Value of the variables at the surface of computational cells are approximated using neighboring cell values. For the value of α^2 at cell surfaces, we recommend approximating it as the geometric mean of α^2 at both sides of the surface, which is the product of α in neighboring cells. This scheme has another advantage apart from simplicity in expression. In the optimized shapes, α is expected to be either 0 or 1 in the design domain. Using the geometric mean ensures no direct transfer of macroscopic quantities across the cell surface, as long as one of the two sides is solid.

In traditional DSMC, molecules that get reflected are subject to reflection boundary conditions. Assuming diffuse boundaries, the reflected molecules will lose their original velocity information, and turn to follow the Maxwellian distribution at wall temperature T_{wall}. In the extended DSMC algorithm, since molecules are free to travel in the entire computational domain, they are not reflected even if they cross cell boundaries where the pseudo density α takes different values on each side. However, different values of α means different degrees of modification. In solid region where $\alpha = 0$, molecules are modified so that their macroscopic velocity is zero and their temperature is equal to T_{wall}. In gas region where $\alpha = 1$, the molecules are not modified, effectively. Therefore, regions where α is close to zero act like sources emitting molecules that follow Maxwellian distribution at T_{wall}, and regions where α is close to 1 act like sources emitting molecules that follow distribution representing gas flow. At gas-solid boundaries, molecules travelling into solid region follow distribution representing the flow, while molecules travelling into gas follow the Maxwellian distribution at T_{wall}. As a result, the diffuse reflection boundary condition is implicitly imposed in the extended DSMC algorithm by the difference in the degree of modification applied to the molecules. The only reflections that are needed in the extended algorithm are those that take place at the boundary of

the computational domain.

3.3. Summary of the extended IP-DSMC algorithm

Here, a brief outline of the extended IP-DSMC algorithm is shown, where a total of M iterations are calculated in the simulation, and properties of the steady flow is sampled starting from iteration number $M_{\text{sample}} < M$.

1. Discretize the computational domain. Initialize the molecules and the cells. Set iteration number $M_{\text{iter}} = 0$.
2. Introduce new molecules that enter the computational domain from the inlet and the outlet.
3. Move the molecules according to their velocities, and perform reflection at the boundary of computational domain when necessary.
4. Update the velocities of the molecules according to external forces.
5. Select proper pairs of molecules in each cell and perform binary collisions.
6. Modify the information properties of the molecules according to flow pressure, flow velocity, and pseudo density α.
7. Update flow density according to flow velocity and pseudo density α.
8. Correct the molecule velocities according to pseudo density α.
9. Sample flow temperature and flow velocity in each cell, update flow pressure.
10. Delete the molecules that moved out of the computational domain.
11. Let $M_{\text{iter}} \leftarrow M_{\text{iter}} + 1$. If $M_{\text{iter}} \geq M_{\text{sample}}$, record the flow properties.
12. If $M_{\text{iter}} < M$, go back to step 2.
13. Calculate the average of the recorded flow properties.

3.4. DSMC procedure as maps

In DSMC, the continuous distribution function f is represented by a finite set of molecules $\{s_j\}_{j=1}^{N}$. Each molecule j has its position x_j and velocity v_j, and carries information velocity V_j and information temperature T_j. Combining them, we can treat each molecule as a vector $s_j = (x_j, v_j, V_j, T_j)^T$. In IP method, velocity, temperature, density, and pressure of the flow are calculated in all the cells $\{u_n\}_{n=1}^{N_c}$. Similarly, flow properties in each cell can be written as a vector $u_n = (V_{f,n}, T_{f,u}, \rho_u, P_u)^T$. For convenience, the set of molecules and cells are treated as two vectors from now on: $S = (s_1, s_2, \cdots, s_N)^T$ and $U = (u_1, u_2, \cdots, u_{N_c})^T$.

During the simulation, the values associated with the molecules and the cells are updated in every step. Let superscript k mark the variables after iteration k ($0 \leq k \leq M$), where M is the total number of iterations. Then effectively, the simulation process can be regarded as applying two maps recursively:

$$S^{k+1} = L_s(S^k, U^k), \tag{35}$$

$$U^{k+1} = L_u(S^k, U^k). \tag{36}$$

During DSMC, each iteration can be further divided into several independent steps, which makes it possible to break L_s and L_u into several maps that act on S and U consecutively. Referring to the algorithm, 9 steps are identified: molecule entry, streaming, reflection, collision, pressure modification, density advection, pseudo density correction, flow property sampling, and molecule deletion. Denoting the corresponding maps by $L_{s,i}$ and $L_{u,i}, (1 \leq i \leq 9)$, respectively, and introducing fractional steps to represent intermediate results, equation (35) and (36) can be expanded by

$$S^{k+i/9} = L_{s,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9}), \tag{37}$$

$$U^{k+i/9} = L_{u,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9}). \tag{38}$$

It should be noted that the exact effects of the above-mentioned maps are subject to random numbers generated during simulation, and they cannot be regarded as completely identical. For instance, the collision step $L^k_{s,4}$ at iteration k can hardly choose the same collision pairs as $L^{k'}_{s,4}$ does at iteration k'. Nevertheless, the mechanisms they follow are the same. So despite the difference in the values of random numbers, they are grouped under the same class $L_{s,4}$.

3.5. Discrete version of the optimization problem

Using the molecules S as the representation of the distribution function f, the discrete version of the optimization problem can be formulated as

$$\inf_{\alpha} K = r(S^M), \tag{39}$$

subject to

$$V_{\text{design}} - V_{\max} \leq 0, \tag{40}$$

$$S^0 = S^I, \tag{41}$$

$$U^0 = U^I, \tag{42}$$

$$S^{k+i/9} = L_{s,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9}), \tag{43}$$

$$U^{k+i/9} = L_{u,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9}), \tag{44}$$

where $r(S^M)$ represents a sampling function over all the molecules, $V_{\text{design}} = \sum_{j=1}^{N_c} \alpha_j / N_c$ is the volume fraction of fluid in the design domain, $0 < V_{\max} < 1$ is the volume constraint, $0 \leq k < M$, and $1 \leq i \leq 9$.

The objective functional K is defined directly on the final state of the molecules S^M, which is supposed to represent the distribution function f at the end of the simulation, and contain all the information about the flow. In practice, the objective functional K is often evaluated at a number of steps in the sampling range then averaged. However, in this formulation we reduce ourselves to the case where the sampling range is equal to 1 for simplicity. The volume constraint is stated in stated in (40). Equation (41) and (42) state the initial condition. And equation (43) and (44) are the governing equations for the state variables S and U.

3.6. Sensitivity analysis

Design sensitivity $\dfrac{\partial K}{\partial \alpha}$ is obtained using the Lagrangian method. Motivated by the discrete version of the optimization problem, we formulate the discrete Lagrangian J as

$$\begin{aligned} J = & \underbrace{r(S^M)}_{K} - \underbrace{\Lambda^0 \cdot (S^0 - S^I) - \Psi^0 \cdot (U^0 - U^I)}_{J_1} \\ & \underbrace{- \sum_{k=0}^{M-1} \sum_{i=1}^{9} \Lambda^{k+i/9} \cdot (S^{k+i/9} - L_{s,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9})}_{J_2} \\ & \underbrace{- \sum_{k=0}^{M-1} \sum_{i=1}^{9} \Psi^{k+i/9} \cdot (U^{k+i/9} - L_{u,i}(S^{k+(i-1)/9}, U^{k+(i-1)/9})}_{J_3} . \end{aligned} \tag{45}$$

Here, Λ and Ψ are vectors of Lagrangian multipliers corresponding to the molecules S and the cells U. Similar to S and U, which are vectors made up by individual s_j and u_n, Λ and Ψ are also constructed as $\Lambda = (\lambda_1, \lambda_2, \cdots, \lambda_N)^T$, $\Psi = (\psi_1, \psi_2, \cdots, \psi_{N_c})^T$. Each λ_j is a vector, $\lambda_j = (\lambda_{x,j}, \lambda_{v,j}, \lambda_{V,j}, \lambda_{T,j})^T$, which are Lagrangian multipliers for the position, velocity, information velocity, and information temperature of molecule j. Similarly, $\psi_n = (\psi_{V,n}, \psi_{T,n}, \psi_{\rho,n}, \psi_{P,n})^T$ is a vector composed by Lagrangian multipliers for velocity, temperature, density, and pressure of the fluid in cell n.

Setting the Fréchet derivative of J with respect to the state variables equal to zero gives the adjoint equations,

$$\nabla_s r(S^M) - \Lambda^M = 0, \tag{46}$$

$$- \Psi^M = 0, \tag{47}$$

$$\begin{aligned} & \left[\nabla_s L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9} \\ & \quad + \left[\nabla_s L_{u,i+1}(S^{k+i/9}, c^{k+i/9})\right]^T \Psi^{k+(i+1)/9} - \Lambda^{k+i/9} = 0, \end{aligned} \tag{48}$$

$$\begin{aligned} & \left[\nabla_u L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9} \\ & \quad + \left[\nabla_u L_{u,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Psi^{k+(i+1)/9} - \Psi^{k+i/9} = 0, \end{aligned} \tag{49}$$

$$\left[\nabla_s L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9} + \left[\nabla_s L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9} - \Lambda^0 = 0, \tag{50}$$

$$\left[\nabla_c L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9} + \left[\nabla_u L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9} - \Psi^0 = 0. \tag{51}$$

Detailed derivations of the adjoint equations can be found in Appendix A. The adjoint equations form a final value problem that evolves backwards in time from the end of the simulation. Appendix B. includes the details for solving the

adjoint equations. When the adjoint variables are solved, sensitivity can be obtained by evaluating the Fréchet derivative of Lagrangian J with respect to α.

In the extended DSMC algorithm, α appears in equations (26) to (27) and equations (32) to (34), which correspond to maps $L_{s,5}, L_{u,6}$ and $L_{s,7}$. In the discrete formulation, the design variable α is represented by its values in each cell $\alpha = (\alpha_1\ \alpha_2\ \cdots\ \alpha_{N_c})^T$. Hence the sensitivity Q can be expressed in discrete form as

$$\begin{aligned} Q =& \frac{\delta J}{\delta \alpha} \\ =& -\sum_{k=0}^{M-1} [\nabla_\alpha L_{s,5}(S^{k+4/9}, U^{k+4/9})]^T \cdot \Lambda^{k+5/9} \\ & -\sum_{k=0}^{M-1} [\nabla_\alpha L_{u,6}(S^{k+5/9}, U^{k+5/9})]^T \cdot \Psi^{k+6/9} \\ & -\sum_{k=0}^{M-1} [\nabla_\alpha L_{s,7}(S^{k+6/9}, U^{k+6/9})]^T \cdot \Lambda^{k+7/9}. \end{aligned} \tag{52}$$

Detailed evaluation of Q is included in Appendix C.

4. Numerical implementation

4.1. Optimization algorithm

Optimization algorithm to maximize objective functional $K(\alpha)$ is based on the steepest gradient method with respect to α [6, 7]. The outline is as follows.

1. Initialize the design variable α and Lagrangian multiplier μ for volume constraint.
2. Perform DSMC calculations to obtain objective functional K and record all the intermediate steps. If K has converged, calculation ends.
3. Use the recorded information to solve the adjoint variables.
4. Calculate design sensitivity Q.
5. Calculate current volume $V_{\text{cur}} = \sum_{j=1}^{N_c} \alpha_j / N_c$. Update the Lagrangian multiplier by $\mu \leftarrow \mu + \sigma(V_{\text{cur}} - V_{\max})$, where σ is the penalty factor.
6. Modify the Lagrangian multiplier by $\mu \leftarrow \max(\min(\mu, \mu_{\max}), 0)$, where $\mu_{\max} > 0$ is a prescribed maximum value.
7. Update design variable by $\alpha \leftarrow \alpha + \tau(Q/|Q| - \mu)$, where $|Q|$ is the Euclidean norm of sensitivity Q, and τ is a prescribed step length.
8. Modify the design variable by $\alpha \leftarrow \max(\min(\alpha, 1), 0)$. Go back to step 2.

Convergence of objective functional K is decided by $\dfrac{|K_n - K_{n-1}|}{|K_n|} < \epsilon$ for three consecutive loops, where n counts the number of loops, and ϵ is a prescribed small value.

5. Numerical examples

5.1. Validation of the extended DSMC

Validity of the extended DSMC method is shown in the calculation of two-dimensional Couette flow. As shown in Figure 1, rarefied gas flows between two parallel plates. The plate at the bottom is fixed, and the one above at $y = y_{max}$ moves in the x-direction with velocity u_{wall}. Both plates are at the same temperature $T_{\text{wall}} = 300$ K and treated as diffuse reflectors. Since the flow field does not vary in the x-direction, a section of the infinite channel is selected as the computational domain, and periodic boundary condition is imposed on the left side and the right side. As a benchmark, calculation is first performed using traditional DSMC with the lower plate explicitly expressed as a wall at $y = 0$. Then, using the extended DSMC as we proposed, calculation is performed in an extended computational domain. In the extended set-up, distance between the plates is 1.2 times the original value, but the bottom 1/6 of the computational domain is solid, where $\alpha = 0$. In both calculations, the computational domain is discretized using square meshes. Fluid domain is divided into 100×100 squares, and solid domain is divided into 100×20 squares.

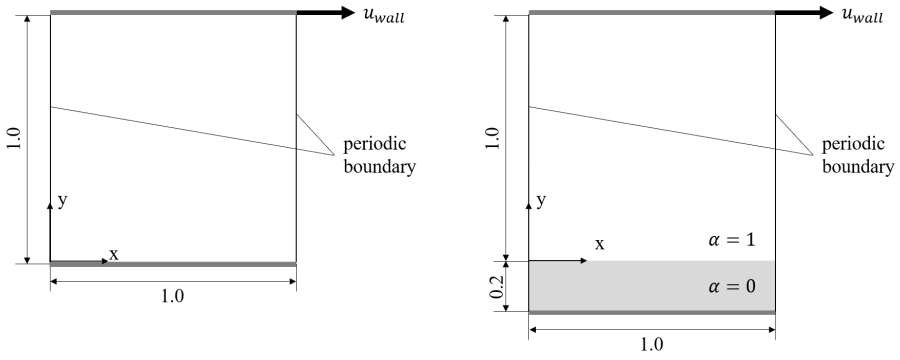
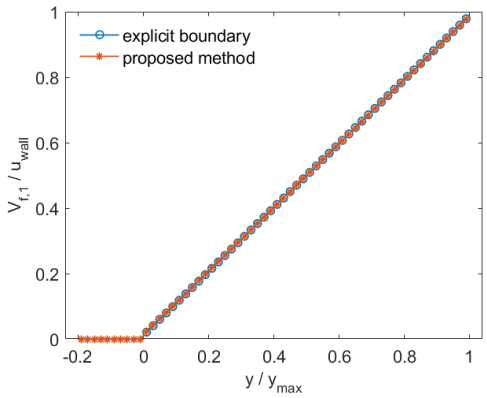


Figure 1: Geometry of the computational domain for plane Couette flow. Left: explicit solid boundary. Right: solid boundary represented by $\alpha = 0$.

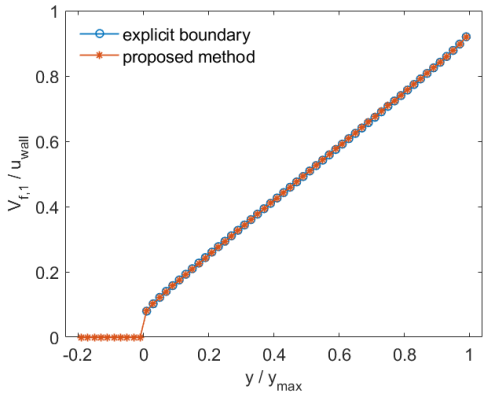
Flow fields obtained from the two methods are compared at three different Knudsen numbers: 0.01, 0.1, and 1.0. Flow velocity in the x-direction $V_{f,1}$ is calculated as the average in all the cells with the same y-coordinate. Figure 2 shows the distribution of $V_{f,1}$ with respect to y-coordinate. Results from the proposed DSMC method are in good agreement with results from traditional DSMC, which confirms the validity of the proposed method in representing the distribution of solid.

5.2. Validation of sensitivity

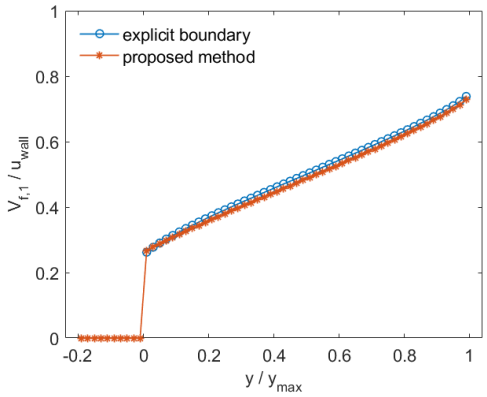
Sensitivity obtained from the adjoint equations is verified using the following two-dimensional channel flow through a neck. As illustrated in Figure 3, the



(a)



(b)



(c)

Figure 2: Comparison of flow velocity in the x-direction for the two-dimensional Couette flow at different Knudsen numbers: (a) Kn = 0.01, (b) Kn = 0.1, (c) Kn = 1.0.

channel has an aspect ratio of 3:1. The top and lower boundaries at $y = \pm h$ are solid walls at $T_{\text{wall}} = 300$ K as diffuse reflectors, while the left and right boundaries are open. Inlet boundary condition at Γ_{in} is $P_{\text{in}} = 1.5$ atm, $T_{\text{in}} = 300$ K, and $V_{f,2} = 0$ m/s. At the outlet Γ_{out}, pressure is $P_{\text{out}} = 1.0$ atm. In the middle of 1/3 of the channel, two solid blocks reduces represented by $\alpha = 0$ reduces the width of the channel to 1/3. Other parts are pure gas domain where $\alpha = 1$.

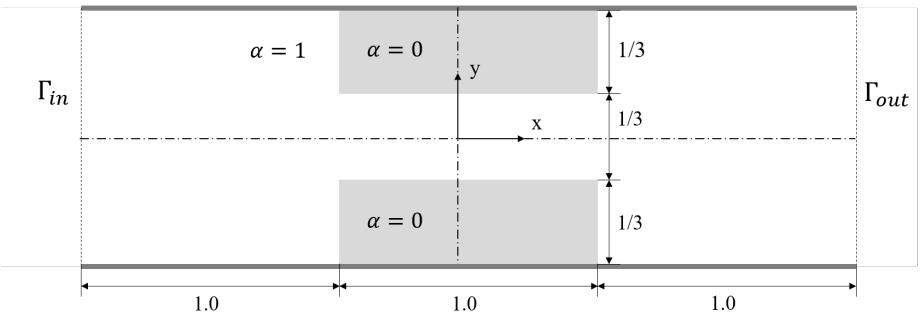


Figure 3: Geometry of the channel with a neck.

Considering the maximization of flow rate, the objective function is defined as the average flow velocity in the x-direction at the inlet, where the upwind flow density is a fixed value due to inlet boundary conditions.

$$K = \frac{\int_{\Gamma_{\text{in}}} V_{f,1} \, d\Gamma}{\int_{\Gamma_{\text{in}}} d\Gamma}. \tag{53}$$

In DSMC calculation, the objective functional is evaluated by the average information velocity in the x-direction of all the molecules that are at the inlet. Suppose there are N_{in} molecules at the inlet, and their indexes are m_i, where $1 \leq i \leq N_{\text{in}}$. K is then expressed by

$$K = \frac{1}{N_{\text{in}}} \sum_{i=1}^{N_{\text{in}}} V_{m_i,1}. \tag{54}$$

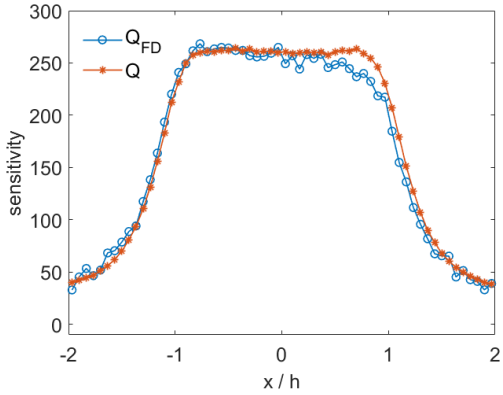
With the discrete expression for objective function, sensitivity Q can be calculated using the adjoint equations. Q is compared with the finite-difference sensitivity Q_{FD}, which is defined as

$$Q_{FD}(x') = \frac{K[\alpha + \mathbb{1}(x - x')\Delta\alpha] - K[\alpha - \mathbb{1}(x - x')\Delta\alpha]}{2\Delta\alpha}, \tag{55}$$

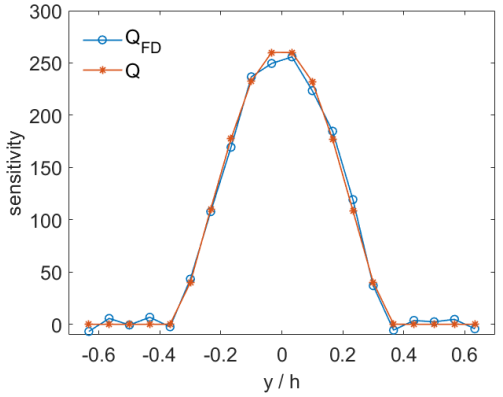
where

$$\mathbb{1}(x) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0, \end{cases} \tag{56}$$

is the Dirac's delta, and $\Delta\alpha = 0.1$.



(a) Sensitivity along the x-axis



(b) Sensitivity along the y-axis

Figure 4: Comparison of Q and Q_{FD} (a): along the x-axis. (b): along the y-axis.

Values of Q and Q_{FD} along the x and y axis at Knudsen number 0.1 are shown in Figure 4. Sensitivity calculated from the adjoint equations agrees with sensitivity calculated from finite difference of the design variable α. Therefore the validity of the derived sensitivity is confirmed.

5.3. Bent pipe optimization

Consider the optimization problem of a bent pipe as is illustrate in Figure 5. The computational domain consists of a square design domain and two smaller non-design domains, where $\alpha = 1$. Apart from the gas inlet Γ_{in} and outlet Γ_{out}, the computational domain is enclosed by diffuse wall boundaries with $T_{\text{wall}} = 300$ K. At the inlet, $P_{\text{in}} = 1.5$ atm, $T_{\text{in}} = 300$ K, and $V_{f,2} = 0$. At the outlet, $P_{\text{out}} = 1.0$ atm. The objective is to design a bent pipe that maximizes mass flow rate under the volume constraint $V_{\max} = 0.25$.

In DSMC calculation, the design domain is discretized by a 100×100 square lattice, and the two non-design domains are discretized by a 20×20 square Lattice. Since the upwind gas density is a fixed value at the inlet, given P_{in} and T_{in}, the objective function is replaced by the average flow velocity in the x-direction at the inlet.

$$K = \frac{\int_{\Gamma_{\text{in}}} V_{f,1} \, d\Gamma}{\int_{\Gamma_{\text{in}}} d\Gamma}. \tag{57}$$

In discrete formulation, K is obtained by the average information velocities in the x-direction of all the molecules that are at the inlet. Suppose there are N_{in} molecules at the inlet, and their indexes are m_i, where $1 \leq i \leq N_{\text{in}}$,

$$K = \frac{1}{N_{\text{in}}} \sum_{i=1}^{N_{\text{in}}} V_{m_i,1}. \tag{58}$$

The optimized structure of the bent pipes at different Knudsen numbers are shown in Figure 6. As the Knudsen number increases, the curvature of the bent pipe decreases, which is in agreement with the results obtained by Sato et al. using LBM [20]. To check the performance of the obtained designs, the obtained structures are tested at different Knudsen numbers, and values of the objective function are compared. The results are shown in Table 1. At each Knudsen number, the maximum objective functional is achieved by the design that is obtained at the same Knudsen number, which confirms the validity of the optimization algorithm.

Design obtained at Kn	Tested at Kn		
	0.1	1.0	10.0
0.1	**20.6**	9.62	8.16
1.0	20.1	**9.91**	8.32
10.0	20.0	9.55	**8.44**

Table 1: Comparison of objective functional for the optimized bent pipe designs at different Knudsen numbers. The maximum value at each tested Knudsen number is shown in bold.

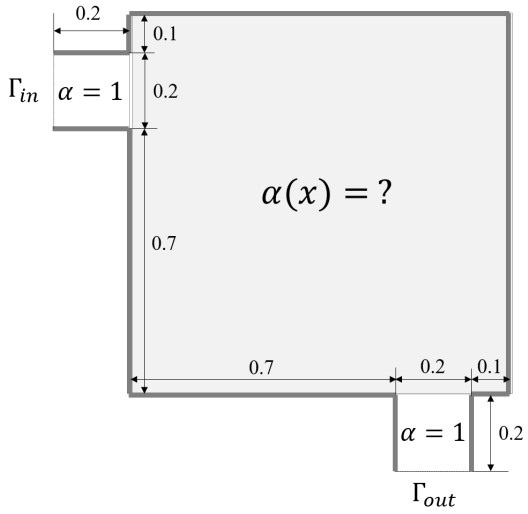


Figure 5: Geometry setting of the bent pipe optimization problem.

(a) (b) (c)

Figure 6: Optimized design for bent pipe at different Knudsen numbers: (a) Kn = 0.1, (b) Kn = 1.0, (c) Kn = 10.0. Shaded region represents solid, and white region represents gas. Only the design domain is shown.

6. Conclusion

In this paper, a topology optimization method for rarefied gas systems based on DSMC is proposed. In order to describe distribution and gas in the design domain, a pseudo density model is introduced, in which solid is treated as gas with zero macroscopic velocity and fixed wall temperature. Traditional IP-DSMC algorithm is extended to incorporate the density expression of material distribution. Topology optimization problem is formulated by treating the particles as discrete state variables and the DSMC process as a series of maps that act on the particle set consecutively.

Topology optimization algorithm is constructed by choosing the pseudo density as design variable directly. Following the discrete view of the optimization problem, discrete adjoint variables are defined for each molecule in the simulation, and adjoint equations are formulated. Design sensitivity is obtained after solving the adjoint variables. The design variable is updated by the steepest gradient method using Lagrangian multipliers.

The validity of the introduced pseudo density, as well as the proposed extension of DSMC is confirmed through numerical examples. Sensitivity obtained from the adjoint variables is also confirmed by comparison with sensitivity calculated from finite difference of the design variable. Bent pipe optimization shows the change of curvature with Knudsen number, which is in line with previous research results.

The proposed topology optimization method can assist the design of micro structures or aircrafts at high altitudes, where rarefied gas flow plays an important role. It may also be applied to other systems governed by the Boltzmann equation that can be solved by DSMC.

Appendix A. Derivation of the adjoint equations

The discrete adjoint equations are formulated by calculating the Fréchet derivatives of Lagrangian J with respect to the state variables S and U at every step of the simulation, and letting the derivatives equal zero according to first-order optimum conditions. Start by derivatives of K, which is only dependent on S^M. Since K is a scalar function, its Fréchet derivative with respect to S^M is simply the gradient of r evaluated at S^M.

$$\frac{\delta K}{\delta S^M} = \nabla_s r(S^M). \tag{59}$$

The initial condition term J_1 in equation (45) is only dependent on the initial state of the simulation, S^0 and U^0, and the derivatives are

$$\frac{\delta J_1}{\delta S^0} = \Lambda^0, \tag{60}$$

$$\frac{\delta J_1}{\delta U^0} = \Psi^0. \tag{61}$$

J_2 and J_3 in (45) have similar formulations. Generally, $S^{k+i/9}$ and $U^{k+i/9}$ appear in the dot product with $\Lambda^{k+i/9}, \Psi^{k+i/9}$, $\Lambda^{t+(i+1)/9}$, and $\Psi^{k+(i+1)/9}$, except when $k = M-1$ and $i = 9$. The derivatives are

$$\frac{\delta J_2}{\delta S^{k+i/9}} = -\left[\nabla_s L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9} + \Lambda^{k+i/9}, \tag{62}$$

$$\frac{\delta J_2}{\delta U^{k+i/9}} = -\left[\nabla_u L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9}, \tag{63}$$

$$\frac{\delta J_3}{\delta S^{k+i/9}} = -\left[\nabla_s L_{u,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Psi^{k+(i+1)/9}, \tag{64}$$

$$\frac{\delta J_3}{\delta U^{k+i/9}} = -\left[\nabla_u L_{u,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Psi^{k+(i+1)/9} + \Psi^{k+i/9}. \tag{65}$$

$k = M-1$ and $i = 9$ correspond to the final state of the simulation: S^M and U^M. They do not appear as the argument of the maps, so the derivatives with respect to them are simpler:

$$\frac{\delta J_2}{\delta S^M} = \Lambda^M, \tag{66}$$

$$\frac{\delta J_3}{\delta U^M} = \Psi^M. \tag{67}$$

Similarly, in J_2 and J_3, the initial state S^0 and U^0 only appear as arguments

of $L_{s,1}$ and $L_{u,1}$, so the corresponding derivatives are

$$\frac{\delta J_2}{\delta S^0} = -\left[\nabla_s L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9}, \tag{68}$$

$$\frac{\delta J_2}{\delta U^0} = -\left[\nabla_u L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9}, \tag{69}$$

$$\frac{\delta J_3}{\delta S^0} = -\left[\nabla_s L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9}, \tag{70}$$

$$\frac{\delta J_3}{\delta U^0} = -\left[\nabla_u L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9}. \tag{71}$$

Now, combining all four parts together, we get the discrete adjoint equations for the problem. Derivatives with respect to final state S^M and U^M being equal zero requires

$$\frac{\delta J}{\delta S^M} = \nabla_s r(S^M) - \Lambda^M = 0, \tag{72}$$

$$\frac{\delta J}{\delta U^M} = -\Psi^M = 0. \tag{73}$$

Derivative with respect to S and U at an intermediate step $k + i/9$ equaling zero requires

$$\begin{aligned} 0 = \frac{\delta J}{\delta S^{k+i/9}} = &\left[\nabla_s L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9} \\ &+ \left[\nabla_s L_{u,i+1}(S^{k+i/9}, c^{k+i/9})\right]^T \Psi^{k+(i+1)/9} - \Lambda^{k+i/9}, \end{aligned} \tag{74}$$

$$\begin{aligned} 0 = \frac{\delta J}{\delta U^{k+i/9}} = &\left[\nabla_u L_{s,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Lambda^{k+(i+1)/9} \\ &+ \left[\nabla_u L_{u,i+1}(S^{k+i/9}, U^{k+i/9})\right]^T \Psi^{k+(i+1)/9} - \Psi^{k+i/9}. \end{aligned} \tag{75}$$

Derivative with respect to initial state S^0 and U^0 equaling zero requires

$$0 = \frac{\delta J}{\delta S^0} = \left[\nabla_s L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9} + \left[\nabla_s L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9} - \Lambda^0, \tag{76}$$

$$0 = \frac{\delta J}{\delta U^0} = \left[\nabla_c L_{s,1}(S^0, U^0)\right]^T \Lambda^{1/9} + \left[\nabla_u L_{u,1}(S^0, U^0)\right]^T \Psi^{1/9} - \Psi^0. \tag{77}$$

The adjoint equations can be treated as a final value problem that evolves backward in time. The final state of the adjoint variables Λ^M, Ψ^M are given in equations (72) and (73). Then according to equations (74) and (75), values of Λ and Ψ at the previous step can be calculated from the current step. The equations can be solved one after another until Λ^0 and Ψ^0 are obtained.

In practice, DSMC calculations usually take a number of steps towards the end as a sampling range, where flow properties are sampled and averaged. However, due to the linearity of the equations in the pair (Λ, Ψ), the derivation only covers the case where the objective function depends on the final state of the simulation. The more general case, where the objective function depends on a number of sampled iterations, can be easily derived following the superposition principle.

Appendix B. Evaluating the adjoint variables

To evaluate the adjoint variables, assume that the forward DSMC process is completed, values of the state variables S and U, as well as and the random numbers generated at every step are recorded. Gradients involved in the adjoint equations are giant vectors and matrices. But fortunately, they can be simplified to make calculation easier.

In IP DSMC, flow properties are sampled from information variables of the molecules. This means $r(S^M)$ only depends on V_j^M and T_j^M. Therefore, for the adjoint variables $\lambda_{x,j}^M$ and $\lambda_{v,j}^M$, their final conditions must be zero. Besides, during the simulation, values of the information variables V_j and T_j are independent of the original variables x_j and v_j. x_j and v_j have no direct influence on U, either. Therefore, in $\nabla_s L_{s,n}$ and $\nabla_s L_{u,n}$ $(1 \leq n \leq 9)$, rows corresponding to x_j and v_j are all zero. As a result, adjoint variables $\lambda_{x,j}$ and $\lambda_{v,j}$ are zero at every step. For this reason, they are omitted for simplicity in the following parts. For two dimensional flows, the information velocity has only two components, so each λ_j can be treated as a vector $(\lambda_{V,j,1}\ \lambda_{V,j,2}\ \lambda_{T,j})^T \in \mathbb{R}^3$, and Λ as a vector in $\mathbb{R}^{3N}$ from now on.

Without loss of generality, the gradients $\nabla_s L_{s,n}$ can be written in block matrix form

$$\nabla_s L_{s,n} = A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & A_{22} & \cdots & A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & A_{NN} \end{pmatrix}. \tag{78}$$

Where each $A_{ij} =$ is a 3×3 matrix corresponding to the gradient of molecule s_i in the output with respect to input molecule $s_j = (V_{j,1}, V_{j,2}, T_j)^T$. Formally,

$$A_{ij} = \nabla_{v_j} \begin{pmatrix} V'_{i,1} \\ V'_{i,2} \\ T'_i \end{pmatrix} = \begin{pmatrix} A_{ij}^{11} & A_{ij}^{12} & A_{ij}^{13} \\ A_{ij}^{21} & A_{ij}^{22} & A_{ij}^{23} \\ A_{ij}^{31} & A_{ij}^{32} & A_{ij}^{33} \end{pmatrix}. \tag{79}$$

Gradient $\nabla_u L_{s,n}$ can be written as a $N \times N_c$ block matrix B. Each block B_{ij} is a 3×5 matrix that corresponds to the gradient of the molecule i in the output with respect to cell $u_j = (V_{f,j,1}, V_{f,j,2}, T_{f,j}, \rho_j, P_j)^T$ in the input.

$$B_{ij} = \nabla_{u_j} \begin{pmatrix} V'_{i,1} \\ V'_{i,2} \\ T'_i \end{pmatrix} = \begin{pmatrix} B_{ij}^{11} & B_{ij}^{12} & B_{ij}^{13} & B_{ij}^{14} & B_{ij}^{15} \\ B_{ij}^{21} & B_{ij}^{22} & B_{ij}^{23} & B_{ij}^{24} & B_{ij}^{25} \\ B_{ij}^{31} & B_{ij}^{32} & B_{ij}^{33} & B_{ij}^{34} & B_{ij}^{35} \end{pmatrix}. \tag{80}$$

Gradient $\nabla_s L_{u,n}$ can be written as a $N_c \times N$ block matrix C. Each block C_{ij} is a 5×3 matrix that corresponds to the gradient of cell u_i in the output with respect to molecule $s_j = (V_{j,1}, V_{j,2}, T_j)^T$ in the input.

$$C_{ij} = \nabla_{s_j} \begin{pmatrix} V'_{f,i,1} \\ V'_{f,i,2} \\ T'_{f,i} \\ \rho'_i \\ P'_i \end{pmatrix} = \begin{pmatrix} C_{ij}^{11} & C_{ij}^{12} & C_{ij}^{13} \\ C_{ij}^{21} & C_{ij}^{22} & C_{ij}^{23} \\ C_{ij}^{31} & C_{ij}^{32} & C_{ij}^{33} \\ C_{ij}^{41} & C_{ij}^{42} & C_{ij}^{43} \\ C_{ij}^{51} & C_{ij}^{52} & C_{ij}^{53} \end{pmatrix}. \tag{81}$$

Gradient $\nabla_u L_{u,n}$ can be written as a $N_c \times N_c$ block matrix D. Each block D_{ij} is a 5×5 matrix that corresponds to the gradient of cell u_i in the output with respect to cell u_j in the input.

$$D_{ij} = \nabla_{u_j} \begin{pmatrix} V'_{f,i,1} \\ V'_{f,i,2} \\ T'_{f,i} \\ \rho'_i \\ P'_i \end{pmatrix} = \begin{pmatrix} D_{ij}^{11} & D_{ij}^{12} & D_{ij}^{13} & D_{ij}^{14} & D_{ij}^{15} \\ D_{ij}^{21} & D_{ij}^{22} & D_{ij}^{23} & D_{ij}^{24} & D_{ij}^{25} \\ D_{ij}^{31} & D_{ij}^{32} & D_{ij}^{33} & D_{ij}^{34} & D_{ij}^{35} \\ D_{ij}^{41} & D_{ij}^{42} & D_{ij}^{43} & D_{ij}^{44} & D_{ij}^{45} \\ D_{ij}^{51} & D_{ij}^{52} & D_{ij}^{53} & D_{ij}^{54} & D_{ij}^{55} \end{pmatrix}. \tag{82}$$

With these notation introduced, it is easier to write out the expressions for the gradients of the functions $L_{s,n}$ and $L_{u,n}$ ($1 \leq n \leq 9$). In the following parts, the expressions for the gradients will be given in terms of block matrices A, B, C, and D.

B.1. Molecule entry step

In the molecule entry step, maps $L_{s,1}$ and $L_{u,1}$ introduce new molecules into the simulation from the inlet and the outlet. In this step, flow properties are not changed, $L_{u,1}(S^t, U^t) = U^t$. Therefore, $\nabla_s L_{u,1} = C = 0$ and $\nabla_u L_{u,1} = D = \mathbb{1}$.

For the map $L_{s,1}$, information variables of existing molecules are not changed. Therefore, $A_{ii} = \mathbb{1}$ for all i representing an existing molecule. Other parts of A are zero.

The introduced molecules get their initial values according to the boundary conditions. Use a horizontal pipe as an example, where the inlet and outlet are in the vertical direction. At the inlet, if molecule i is introduced to cell j, its initial information velocity in x-direction $V'_{i,1}$ is equal to the flow velocity in cell j, $V_{f,j,1}$. Information velocity in y-direction is zero, and information temperature is equal to prescribed value T_{in}. At the outlet, the initial value for the information variables of molecule i follow the characteristics-theory-based equations [29, 30],

$$V'_{i,1} = V_{f,j,1} + \frac{P_j - P_{\text{out}}}{\rho_j \sqrt{\gamma R T_{f,j}}}, \tag{83}$$

$$V'_{i,2} = V_{f,j,2}, \tag{84}$$

$$T'_i = \frac{P_{\text{out}}}{\rho_j R}, \tag{85}$$

where P_{out} is prescribed pressure at the outlet, γ is the ratio of specific heats for the gas, and R is the specific gas constant. From the equations (83) to (85), the related partial derivatives can be calculated.

For molecule i introduced to cell j at the inlet, matrix B_{ij} has only one non-zero component,

$$B_{ij}^{11} = 1. \tag{86}$$

For molecule i introduced to cell j at the outlet, the non-zero components of B_{ij} are

$$B_{ij}^{11} = B_{ij}^{22} = 1, \tag{87}$$

$$B_{ij}^{13} = -\frac{P_j - P_{\text{out}}}{\rho_j \sqrt{\gamma R}} \left(\frac{1}{T_{f,j}}\right)^{3/2}, \tag{88}$$

$$B_{ij}^{14} = -\frac{P_j - P_{\text{out}}}{\sqrt{\gamma R T_{f,j}}} \left(\frac{1}{\rho_j}\right)^{2}, \tag{89}$$

$$B_{ij}^{15} = \frac{1}{\rho_j \sqrt{\gamma R T_{f,j}}}, \tag{90}$$

$$B_{ij}^{34} = -\frac{P_{\text{out}}}{\rho_j^2 R}. \tag{91}$$

For other cells c_j where no new molecule is introduced, $B_{ij} = 0$.

B.2. Streaming step

In the streaming step, $L_{s,2}, L_{u,2}$ change the positions of the molecules according to their velocities. Cell variables and the information variables of the molecules do not change, hence $A = \mathbb{1}, B = 0, C = \mathbb{1}, D = 0$. As a result, adjoint variables do not change from step $k + 2/9$ to $k + 1/9$.

B.3. Reflection step

In the reflection step, $L_{p,3}$ and $L_{u,3}$ assign new velocities to molecules if they attempt to cross the wall boundary. For molecule i that get reflected, the new information variable are set according to diffuse reflection condition:

$$V_i' = 0, \tag{92}$$

$$T_j' = T_{\text{wall}}. \tag{93}$$

The information variables after reflection do not depend on the values before reflection. Hence, $A_{mm} = \mathbb{1}$ for all molecule m that is not reflected. Other parts of A are zero. Molecular Information variables do not depend on flow properties in this step, hence $B = 0$. Cell variables do not change in this step, so $C = 0$, and $D = \mathbb{1}$.

B.4. Collision step

$L_{s,4}$ and $L_{u,4}$ perform collisions between molecules. If molecule i and j collide, their information variables after collision are updated according to equations (9) to (12). From these equations we can derive the partial derivatives

with respect to pre-collision values. In A_{ii} and A_{jj}, the non-zero components are:

$$A_{ii}^{11} = A_{ii}^{22} = A_{jj}^{11} = A_{jj}^{22} = \frac{1 + C_\mu \cos\theta}{2}, \tag{94}$$

$$A_{ii}^{31} = (V_{i,1} - V_{j,1})/(2\xi R), \tag{95}$$

$$A_{ii}^{32} = (V_{i,2} - V_{j,2})/(2\xi R), \tag{96}$$

$$A_{jj}^{31} = (1 - C_\mu^2 \cos^2\theta)(V_{i,1} - V_{j,1})/(2\xi R), \tag{97}$$

$$A_{jj}^{32} = (1 - C_\mu^2 \cos^2\theta)(V_{i,2} - V_{j,2})/(2\xi R), \tag{98}$$

$$A_{ii}^{33} = A_{jj}^{33} = \frac{1 + C_k \cos\theta}{2}. \tag{99}$$

Non-zero components in A_{ij} and A_{ji} are:

$$A_{ij}^{11} = A_{ij}^{22} = A_{ji}^{11} = A_{ji}^{22} = \frac{1 - C_\mu \cos\theta}{2}, \tag{100}$$

$$A_{ij}^{31} = -(V_{i,1} - V_{j,1})/(2\xi R), \tag{101}$$

$$A_{ij}^{32} = -(V_{i,2} - V_{j,2})/(2\xi R), \tag{102}$$

$$A_{ji}^{31} = -(1 - C_\mu^2 \cos^2\theta)(V_{i,1} - V_{j,1})/(2\xi R), \tag{103}$$

$$A_{ji}^{32} = -(1 - C_\mu^2 \cos^2\theta)(V_{i,2} - V_{j,2})/(2\xi R), \tag{104}$$

$$A_{ij}^{33} = A_{ji}^{33} = \frac{1 - C_k \cos\theta}{2}. \tag{105}$$

For molecule i that did not involve in any collision, its information variables do not change, hence $A_{ii} = \mathbb{1}$. Other parts of A are zero. Flow information does not influence collision pairs, and remains the same during this step, hence $B = 0$, $C = 0$ and $D = \mathbb{1}$.

B.5. Pressure modification step

In the pressure modification step, $L_{s,5}$ and $L_{u,5}$ modify the information variables of the molecules according to cell pressure according to equations (32) and (33). During the simulation, finite volume approximations are used to solve the two equations. In the square mesh, pressure gradient is replaced by the difference in neighboring cells. For instance, x-direction information velocity of a molecule i in cell j is modified by

$$V'_{i,1} = V_{i,1} + \Delta V_1 = V_{i,1} + \frac{\Delta t}{\Delta x}\frac{1}{\rho_j}(P_a - P_b)\alpha_a\alpha_b. \tag{106}$$

Here, Δt is the time step for DSMC simulation, Δx is the size of the cell. a and b denote the index of the left cell and the right cell, respectively. The left and right cell are defined in the following way,

- If molecule i is in the left half of cell j, then the left cell is the cell to the left of cell j, and the right cell is cell j.

- If molecule i is the right half of cell j, then the left cell is cell j, and the right cell is the cell to the right of j.

Using this expression for pressure gradient avoids the central difference scheme. Using the central difference scheme creates a undesirable situation, where the pressure gradient at cell j is independent of pressure in cell j. This may result in checkerboard pattern of flow pressure.

Similarly, the y-direction information velocity of i is modified by

$$V'_{i,2} = V_{i,2} + \Delta V_2 = V_{i,2} + \frac{\Delta t}{\Delta x}\frac{1}{\rho_j}(P_c - P_d)\alpha_c\alpha_d. \tag{107}$$

In the subscript, c and d denote the index of the bottom cell and the top cell, which are defined similarly:

- If molecule i is in the lower half of cell j, the bottom cell is the cell to below cell j, and the up top is cell j.
- If molecule i is in the upper half of cell j, the bottom cell is cell j, and the top cell is the cell above cell j.

After calculating the changes in information velocity, information temperature of cell i is modified by the following approximation,

$$\Delta T = \frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}\left[(V_{f,a,1}P_a - V_{f,b,1}P_b)\alpha_a\alpha_b + (V_{f,c,2}P_c - V_{f,d,2}P_d)\alpha_c\alpha_d\right], \tag{108}$$

$$T'_i = T_i + \frac{|V_i|^2 - |V'_i|^2}{\xi R} + \Delta T. \tag{109}$$

Following the equations (106) to (109), the gradients can be calculated as follows. For molecule i modified, the non-zero components of A_{ii} are

$$A_{ii}^{11} = A_{ii}^{22} = A_{ii}^{33} = 1 \tag{110}$$

$$A_{ii}^{31} = -\Delta V_1/(\xi R) \tag{111}$$

$$A_{ii}^{32} = -\Delta V_2/(\xi R). \tag{112}$$

Modification of temperature of molecule i involves many cells, the corre-

sponding non-zero components are

$$B_{ij}^{14} = -\Delta V_1/\rho_j, \tag{113}$$

$$B_{ij}^{24} = -\Delta V_2/\rho_j, \tag{114}$$

$$B_{ij}^{34} = -\Delta T/\rho_j, \tag{115}$$

$$B_{ia}^{15} = -B_{ib}^{15} = \frac{\Delta t}{\Delta x}\frac{1}{\rho_j}\alpha_a\alpha_b, \tag{116}$$

$$B_{ia}^{35} = -B_{ib}^{35} = -\frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}\Delta V_1\alpha_a\alpha_b, \tag{117}$$

$$B_{ic}^{25} = -B_{id}^{25} = \frac{\Delta t}{\Delta x}\frac{1}{\rho_j}\alpha_c\alpha_d, \tag{118}$$

$$B_{ic}^{35} = -B_{id}^{35} = -\frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}\Delta V_2\alpha_c\alpha_d, \tag{119}$$

$$B_{ia}^{31} = \frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}P_a\alpha_a\alpha_b, \tag{120}$$

$$B_{ib}^{31} = -\frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}P_b\alpha_a\alpha_b, \tag{121}$$

$$B_{ic}^{32} = \frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}P_c\alpha_c\alpha_d, \tag{122}$$

$$B_{id}^{32} = -\frac{2}{\xi R}\frac{\Delta t}{\Delta x}\frac{1}{\rho_j}P_d\alpha_c\alpha_d. \tag{123}$$

Other parts of the matrix B are zero.

In this step, the flow properties do not change, hence $C = 0$, and $D = \mathbb{1}$.

B.6. Density advection step

In the advection step, the gas density is updated in the computational domain by $L_{s,6}$ and $L_{u,6}$ according to the modified advection equation (34).

During the simulation, the advection equation is solved by a finite volume approximation. Using the square meshes, we only need to calculate the flux across the four surfaces of the cell. Using the upwind scheme in density, and a simple average in velocity, the expression for new density is,

$$\rho_i' = \rho_i + \Delta\rho, \tag{124}$$

$$\begin{aligned}\Delta\rho = \frac{\Delta t}{\Delta x}\Bigg(&\rho_{l'}\frac{V_{f,a,1}+V_{f,i,1}}{2}\alpha_a - \rho_{r'}\frac{V_{f,b,1}+V_{f,i,1}}{2}\alpha_b \\ &+ \rho_{d'}\frac{V_{f,c,2}+V_{f,i,2}}{2}\alpha_c - \rho_{u'}\frac{V_{f,d,2}+V_{f,i,2}}{2}\alpha_d\Bigg)\alpha_i.\end{aligned} \tag{125}$$

For flow velocities V_f and pseudo density α, a, b, c and d in the subscripts stand for the index of the cell that is to the left of i, to the right of i, below i, and above i, respectively. In flow density, subscripts a', b', c', d' stand for the index of the cells where density is chosen as the upwind value. For instance, if

$V_{f,a,1} + V_{f,i,1} > 0$, a' is the index of the cell to the left of cell i, otherwise, a' is i.

In this step, the molecule variables are not changed, hence $A = \mathbb{1}$, and $B = 0$. Cell variables are independent of molecule variables, so $C = 0$. The change of density for cell i is dependent on the velocities and densities of the neighboring cells. Therefore, the following parts of D can be expressed as:

$$D_{ii}^{44} = 1, \tag{126}$$

$$D_{ii}^{41} = \frac{\Delta t}{\Delta x}\left(\frac{\rho_{l'}}{2}\alpha_a\alpha_i - \frac{\rho_{r'}}{2}\alpha_b\alpha_i\right), \tag{127}$$

$$D_{ii}^{42} = \frac{\Delta t}{\Delta x}\left(\frac{\rho_{d'}}{2}\alpha_c\alpha_i - \frac{\rho_{u'}}{2}\alpha_d\alpha_i\right), \tag{128}$$

$$D_{ia}^{41} = \frac{\Delta t}{\Delta x}\frac{\rho_{l'}}{2}\alpha_a\alpha_i, \tag{129}$$

$$D_{ia'}^{44} = \frac{\Delta t}{\Delta x}\frac{V_{f,l,1} + V_{f,i,1}}{2}\alpha_a\alpha_i, \tag{130}$$

$$D_{ib}^{41} = -\frac{\Delta t}{\Delta x}\frac{\rho_{r'}}{2}\alpha_a\alpha_i, \tag{131}$$

$$D_{ib'}^{44} = -\frac{\Delta t}{\Delta x}\frac{V_{f,r,1} + V_{f,i,1}}{2}\alpha_b\alpha_i, \tag{132}$$

$$D_{ic}^{42} = \frac{\Delta t}{\Delta x}\frac{\rho_{d'}}{2}\alpha_c\alpha_i, \tag{133}$$

$$D_{ic'}^{44} = \frac{\Delta t}{\Delta x}\frac{V_{f,d,2} + V_{f,i,2}}{2}\alpha_c\alpha_i, \tag{134}$$

$$D_{id}^{42} = -\frac{\Delta t}{\Delta x}\frac{\rho_{u'}}{2}\alpha_d\alpha_i, \tag{135}$$

$$D_{id'}^{44} = -\frac{\Delta t}{\Delta x}\frac{V_{f,u,2} + V_{f,i,2}}{2}\alpha_d\alpha_i. \tag{136}$$

Note that, due to the definition of cells a, b, c, d and a', b', c', d', it is possible for an element in matrix D to be double defined. In that case, instead of overwriting, the values should be added. Other parts of matrix D are zero.

B.7. Pseudo density correction

In the pseudo density correction step, the information variable of the molecules are correcting by $L_{s,7}$ and $L_{u,7}$ according to the following equations. For molecule i in cell j,

$$V_i' = \alpha_j V_i, \tag{137}$$

$$T_j' = \alpha_j T_j' + (1 - \alpha_j)T_{\text{wall}}. \tag{138}$$

Information variables of the molecules after the step are dependent on their values before. Hence the matrix A_{ii} has the following non-zero components:

$$A_{ii}^{11} = A_{ii}^{22} = A_{ii}^{33} = \alpha_j. \tag{139}$$

Other parts of A are zero. Since the cell properties are not changed in this step, $B = 0$, $C = 0$, and $D = \mathbb{1}$.

B.8. IP sampling step

In the IP sampling step, $L_{s,8}$ and $L_{u,8}$ updates the flow properties according to the information variables of the molecules inside. Suppose molecule s_{i_n} are in cell j, where $1 \leq n \leq N_j$. The flow properties are updated by

$$V'_{f,j} = \frac{1}{N_j} \sum_{n=1}^{N_j} V_{i_n}, \tag{140}$$

$$T'_{f,j} = \frac{1}{N_j} \sum_{n=1}^{N_j} T_{i_n}, \tag{141}$$

$$P'_j = \rho_j R T'_{f,j}. \tag{142}$$

Since the molecule variables do not change in this step, $A = \mathbb{1}$ and $B = 0$. For flow properties, velocity, temperature, and pressure are dependent on molecule variables, so for matrix C_{ji_n} the non-zero components are,

$$C^{11}_{ji_n} = C^{22}_{ji_n} = C^{33}_{ji_n} = 1/N_j, \tag{143}$$

$$C^{53}_{ji_n} = \rho_j R/N_j. \tag{144}$$

Other parts of matrix C are zero.

Density in each cell is not changed, and it also influences the new pressure in the cell, hence for D_{jj}, the non-zero components are,

$$D^{54}_{jj} = RT'_{f,j}, \tag{145}$$

$$D^{55}_{jj} = 1. \tag{146}$$

Other parts of matrix D are zero.

B.9. Molecule deletion step

In the molecule deletion step, $L_{s,9}$ and $L_{u,9}$ removes the molecule from memory if it moves out of the computational domain. For molecule i that is deleted. All its information variables are cleared to zero.

$$V'_i = T'_i = 0. \tag{147}$$

Hence for matrix A, only $A_{ii} = \mathbb{1}$ only for i such that molecule i is not deleted. Other parts of A are zero. Since the deletion of molecules is independent of flow properties, $B = 0$. Flow variables in each cell are also not changed, hence $C = 0$, $D = \mathbb{1}$.

Appendix C. Detailed formulation of sensitivity

In equation (52), sensitivity Q is defined by the gradients of functions with respect to design variable α. To get the detailed expressions, we use the similar

block matrix expression. The gradient $\nabla_\alpha L_{s,5}$ and $\nabla_\alpha L_{s,7}$ can be written as a $N \times N_c$ block matrix G.

$$\nabla_s L_{s,n} = G = \begin{pmatrix} G_{11} & G_{12} & \cdots & G_{1N_c} \\ G_{21} & G_{22} & \cdots & G_{2N_c} \\ \vdots & \vdots & \ddots & \vdots \\ G_{N1} & G_{N2} & \cdots & G_{NN_c} \end{pmatrix}. \tag{148}$$

Each block G_{ij} is a column vector corresponding to the partial derivative of $s'_j = (V'_{j,1}, V'_{j,2}, T'_j)^T$ in the output with respect to design variable α_j in cell j.

$$G_{ij} = \frac{\partial}{\partial \alpha_j} \begin{pmatrix} V'_{i,1} \\ V'_{i,2} \\ T'_i \end{pmatrix} = \begin{pmatrix} G^1_{ij} \\ G^2_{ij} \\ G^3_{ij} \end{pmatrix}. \tag{149}$$

Similarly, gradient $\nabla_\alpha L_{u,6}$ can be written as a $N_c \times N_c$ block matrix H. Each block H_{ij} is a column vector corresponding to the partial derivative of flow variables in cell i in the output, $(V'_{f,i,1}, V'_{f,i,2}, T'_{f,i}, \rho'_i, P'_i)^T$, with respect to α_j.

$$H_{ij} = \frac{\partial}{\partial \alpha_j} \begin{pmatrix} V'_{f,i,1} \\ V'_{f,i,2} \\ T'_{f,i} \\ \rho'_i \\ P'_i \end{pmatrix} = \begin{pmatrix} H^1_{ij} \\ H^2_{ij} \\ H^3_{ij} \\ H^4_{ij} \\ H^5_{ij} \end{pmatrix}. \tag{150}$$

In the pressure modification step $L_{s,5}$, the information variables of molecule i in cell j is updated according to equations (106) to (109), which gives the following non-zero components in G:

$$G^1_{ia} = \Delta V_1 / \alpha_a, \tag{151}$$

$$G^1_{ib} = \Delta V_1 / \alpha_b, \tag{152}$$

$$G^2_{ic} = \Delta V_2 / \alpha_c, \tag{153}$$

$$G^2_{id} = \Delta V_2 / \alpha_d, \tag{154}$$

$$G^3_{ia} = \frac{2}{\xi R} V'_{1,i} \Delta V_1 / \alpha_a + \frac{2}{\xi R} \frac{\Delta t}{\Delta x} \frac{1}{\rho_j} (V_{f,l,1} P_l - V_{f,r,1} P_r) \alpha_b, \tag{155}$$

$$G^3_{ib} = \frac{2}{\xi R} V'_{1,i} \Delta V_1 / \alpha_b + \frac{2}{\xi R} \frac{\Delta t}{\Delta x} \frac{1}{\rho_j} (V_{f,l,1} P_l - V_{f,r,1} P_r) \alpha_a, \tag{156}$$

$$G^3_{ic} = \frac{2}{\xi R} V'_{2,i} \Delta V_2 / \alpha_c + \frac{2}{\xi R} \frac{\Delta t}{\Delta x} \frac{1}{\rho_j} (V_{f,d,2} P_d - V_{f,u,2} P_u) \alpha_d, \tag{157}$$

$$G^3_{id} = \frac{2}{\xi R} V'_{2,i} \Delta V_2 / \alpha_d + \frac{2}{\xi R} \frac{\Delta t}{\Delta x} \frac{1}{\rho_j} (V_{f,d,2} P_d - V_{f,u,2} P_u) \alpha_c. \tag{158}$$

For density advection step $L_{u,6}$, the change in cell density i is calculated according to equations (124) and (125). Following their expressions, the non-

zero components in matrix H should be

$$H_{ii}^4 = \Delta\rho/\alpha_i, \tag{159}$$

$$H_{ia}^4 = \frac{\Delta t}{\Delta x}\rho_{a'}\frac{V_{f,a,1}+V_{f,i,1}}{2}\alpha_i, \tag{160}$$

$$H_{ib}^4 = -\frac{\Delta t}{\Delta x}\rho_{b'}\frac{V_{f,b,1}+V_{f,i,1}}{2}\alpha_i, \tag{161}$$

$$H_{ic}^4 = \frac{\Delta t}{\Delta x}\rho_{c'}\frac{V_{f,c,2}+V_{f,i,2}}{2}\alpha_i, \tag{162}$$

$$H_{id}^4 = -\frac{\Delta t}{\Delta x}\rho_{d'}\frac{V_{f,d,2}+V_{f,i,2}}{2}\alpha_i. \tag{163}$$

In the pseudo density correction step, $L_{s,7}$ changes the information variables of the molecules by the pseudo density of the cell. For molecule i in cell j, the correction result in the following non-zero components in G_{ij},

$$G_{ij}^1 = V_{i,1}, \tag{164}$$

$$G_{ij}^2 = V_{i,2}, \tag{165}$$

$$G_{ij}^3 = T_j - T_{\text{wall}}. \tag{166}$$

Other parts of matrix G are zero.

Competing interests

The authors declare no conflicts of interest.

Author contributions statement

The authors confirm contribution to the paper as follows: Kaiwen Guan: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing-Original Draft, Visualization. Yuki Noguchi: Methodology, Data curation, Writing-Review & Editing. Kei Matsushima: Methodology, Software, Writing-Review & Editing. Takayuki Yamada: Conceptualization, Methodology, Software, Resources, Data curation, Writing-Review & Editing, Project administration, Supervision.

References

[1] C. Shen, Rarefied Gas Dynamics, Springer, Berlin, Heidelberg, 2005. doi:https://doi.org/10.1007/b138784.

[2] D. Mostaza Prieto, B. P. Graziano, P. C. Roberts, Spacecraft drag modelling, Progress in Aerospace Sciences 64 (2014) 56–65. doi:https://doi.org/10.1016/j.paerosci.2013.09.001.
URL https://www.sciencedirect.com/science/article/pii/S0376042113000754

[3] B. R. Hollis, S. Borrelli, Aerothermodynamics of blunt body entry vehicles, Progress in Aerospace Sciences 48-49 (2012) 42–56, assessment of Aerothermodynamic Flight Prediction Tools. doi:https://doi.org/10.1016/j.paerosci.2011.09.005.
URL https://www.sciencedirect.com/science/article/pii/S0376042111001060

[4] Y. Wang, S. Liu, C. Zhuo, C. Zhong, Investigation of nonlinear squeeze-film damping involving rarefied gas effect in micro-electro-mechanical systems, Computers and Mathematics with Applications 114 (2022) 188–209. doi:https://doi.org/10.1016/j.camwa.2022.03.045.
URL https://www.sciencedirect.com/science/article/pii/S0898122122001420

[5] A. Tsimpoukis, D. Valougeorgis, Linear harmonic oscillatory rarefied gas flow with arbitrary frequency in comb finger blocks, Sensors and Actuators A: Physical 331 (2021) 112997. doi:https://doi.org/10.1016/j.sna.2021.112997.
URL https://www.sciencedirect.com/science/article/pii/S0924424721004623

[6] M. P. Bendsøe, Optimal shape design as a material distribution problem, Structural optimization 1. doi:10.1007/BF01650949.
URL https://doi.org/10.1007/BF01650949

[7] G. Allaire, E. Bonnetier, G. Francfort, Shape optimization by the homogenization method, Numerische Mathematik 76. doi:10.1007/s002110050253.
URL https://doi.org/10.1007/s002110050253

[8] M. P. Bendsøe, N. Kikuchi, Generating optimal topologies in structural design using a homogenization method, Computer Methods in Applied Mechanics and Engineering 71 (2) (1988) 197–224. doi:https://doi.org/10.1016/0045-7825(88)90086-2.
URL https://www.sciencedirect.com/science/article/pii/0045782588900862

[9] M. Bendsøe, O. Sigmund, Material interpolation schemes in topology optimization, Archive of Applied Mechanics 69 (9-10) (1999) 635 – 654, cited by: 1827. doi:10.1007/s004190050248.
URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-0001241926&doi=10.1007%2fs004190050248&partnerID=40&md5=92af4b739ff896e9bb13dc3442008e4b

[10] M. Bendsøe, Optimal shape design as a material distribution problem, Structural Optimization 1 (4) (1989) 193 – 202, cited by: 2768. doi:10.1007/BF01650949.
URL https://www.scopus.com/inward/record.uri?eid=2-s2.

0-0001695416&doi=10.1007%2fBF01650949&partnerID=40&md5=2686dfb745966e578620ca2ac22ca6b0

[11] M. Y. Wang, X. Wang, D. Guo, A level set method for structural topology optimization, Computer Methods in Applied Mechanics and Engineering 192 (1) (2003) 227–246. doi:https://doi.org/10.1016/S0045-7825(02)00559-5.
URL https://www.sciencedirect.com/science/article/pii/S0045782502005595

[12] G. Allaire, F. Jouve, A.-M. Toader, Structural optimization using sensitivity analysis and a level-set method, Journal of Computational Physics 194 (1) (2004) 363–393. doi:https://doi.org/10.1016/j.jcp.2003.09.032.
URL https://www.sciencedirect.com/science/article/pii/S002199910300487X

[13] T. Dbouk, A review about the engineering design of optimal heat transfer systems using topology optimization, Applied Thermal Engineering 112 (2017) 841–854. doi:https://doi.org/10.1016/j.applthermaleng.2016.10.134.
URL https://www.sciencedirect.com/science/article/pii/S135943111632645X

[14] R. T. Bonnecaze, G. J. Rodin, O. Sigmund, J. Søndergaard Jensen, Systematic design of phononic band gap materials and structures by topology optimization, Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences 361 (1806) (2003) 1001–1019. doi:10.1098/rsta.2003.1177.
URL https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2003.1177

[15] T. Borrvall, J. Petersson, Topology optimization of fluids in stokes flow, International Journal for Numerical Methods in Fluids 41 (1) (2003) 77–107. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.426, doi:https://doi.org/10.1002/fld.426.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.426

[16] A. Gersborg-Hansen, O. Sigmund, R. Haber, Topology optimization of channel flow problems, Structural and Multidisciplinary Optimization 30 (3) (2005) 181–192. doi:10.1007/s00158-004-0508-7.
URL https://doi.org/10.1007/s00158-004-0508-7

[17] Y. Deng, Z. Liu, P. Zhang, Y. Liu, Y. Wu, Topology optimization of unsteady incompressible navier–stokes flows, Journal of Computational Physics 230 (17) (2011) 6688–6708. doi:https://doi.org/10.1016/j.jcp.2011.05.004.
URL https://www.sciencedirect.com/science/article/pii/S0021999111003019

[18] G. Bird, Recent advances and current challenges for dsmc, Computers & Mathematics with Applications 35 (1) (1998) 1–14. doi:https://doi.org/10.1016/S0898-1221(97)00254-X.
URL https://www.sciencedirect.com/science/article/pii/S089812219700254X

[19] E. Oran, C. Oh, B. Cybyk, Direct simulation monte carlo: Recent advances and applications, Annual Review of Fluid Mechanics 30 (1) (1998) 403–441. arXiv:https://doi.org/10.1146/annurev.fluid.30.1.403, doi:10.1146/annurev.fluid.30.1.403.
URL https://doi.org/10.1146/annurev.fluid.30.1.403

[20] A. Sato, T. Yamada, K. Izui, S. Nishiwaki, S. Takata, A topology optimization method in rarefied gas flow problems using the boltzmann equation, Journal of Computational Physics 395 (2019) 60–84. doi:https://doi.org/10.1016/j.jcp.2019.06.022.
URL https://www.sciencedirect.com/science/article/pii/S0021999119304309

[21] G. Pingen, A. Evgrafov, K. Maute, Topology optimization of flow domains using the lattice boltzmann method, Structural and Multidisciplinary Optimization 34. doi:10.1007/s00158-007-0105-7.
URL https://doi.org/10.1007/s00158-007-0105-7

[22] T. Nguyen, H. Isakari, T. Takahashi, K. Yaji, M. Yoshino, T. Matsumoto, Level-set based topology optimization of transient flow using lattice boltzmann method considering an oscillating flow condition, Computers & Mathematics with Applications 80 (1) (2020) 82–108. doi:https://doi.org/10.1016/j.camwa.2020.03.003.
URL https://www.sciencedirect.com/science/article/pii/S0898122120301000

[23] F. Dugast, Y. Favennec, C. Josset, Y. Fan, L. Luo, Topology optimization of thermal fluid flows with an adjoint lattice boltzmann method, Journal of Computational Physics 365 (2018) 376–404. doi:https://doi.org/10.1016/j.jcp.2018.03.040.
URL https://www.sciencedirect.com/science/article/pii/S0021999118302067

[24] R. Caflisch, D. Silantyev, Y. Yang, Adjoint DSMC for nonlinear boltzmann equation constrained optimization, Journal of Computational Physics 439 (2021) 110404. doi:10.1016/j.jcp.2021.110404.
URL https://www.sciencedirect.com/science/article/pii/S0021999121002990

[25] P. Prasanth, J. K. Kakkassery, Molecular models for simulation of rarefied gas flows using direct simulation monte carlo method, Fluid Dynamics Research 40 (4) (2008) 233–252. doi:https:

//doi.org/10.1016/j.fluiddyn.2007.10.001.
URL https://www.sciencedirect.com/science/article/pii/S0169598307000834

[26] C. Shen, J. Fan, C. Xie, Statistical simulation of rarefied gas flows in micro-channels, Journal of Computational Physics 189 (2) (2003) 512–526. doi:https://doi.org/10.1016/S0021-9991(03)00231-6.
URL https://www.sciencedirect.com/science/article/pii/S0021999103002316

[27] J. Fan, C. Shen, Statistical simulation of low-speed rarefied gas flows, Journal of Computational Physics 167 (2) (2001) 393–412. doi:https://doi.org/10.1006/jcph.2000.6681.
URL https://www.sciencedirect.com/science/article/pii/S0021999100966816

[28] Q. Sun, I. D. Boyd, A direct simulation method for subsonic, microscale gas flows, Journal of Computational Physics 179 (2) (2002) 400–425. doi:https://doi.org/10.1006/jcph.2002.7061.
URL https://www.sciencedirect.com/science/article/pii/S0021999102970610

[29] Y. Fang, W. W. Liou, Computations of the Flow and Heat Transfer in Microdevices Using DSMC With Implicit Boundary Conditions , Journal of Heat Transfer 124 (2) (2001) 338–345. arXiv:https://asmedigitalcollection.asme.org/heattransfer/article-pdf/124/2/338/5749463/338_1.pdf, doi:10.1115/1.1447933.
URL https://doi.org/10.1115/1.1447933

[30] W. Liou, Y. Fang, Implicit boundary conditions for direct simulation monte carlo method in mems flow predictions, Computer Modeling in Engineering & Sciences 1 (4) (2000) 119–128. doi:10.3970/cmes.2000.001.571.
URL http://www.techscience.com/CMES/v1n4/24713