

Article

A Meshless Algorithm for Modeling the Gas-Dynamic Interaction between High-Inertia Particles and a Shock Layer

Andrey Sposobin and Dmitry Reviznikov *

Moscow Aviation Institute, National Research University, Volokolamskoye sh. 4, 125993 Moscow, Russia

* Correspondence: reviznikov@mai.ru

Abstract: This paper is devoted to numerical modeling of a supersonic flow around a blunt body by a viscous gas with an admixture of relatively large high-inertia particles that, after reflection from the surface, may go beyond the shock layer and change the flow structure dramatically. To calculate the gas-dynamic interaction of moving particles with the shock layer, it is important to take into account the large difference in scales of the flow around the particles and around the body. To make the computations effective, we use a meshless method to solve non-stationary Navier–Stokes equations. The algorithm is based on the approximation of partial derivatives by the least squares method on a set of nodes distributed in the calculation area. Each moving particle is surrounded by a cloud of calculation nodes belonging to its domain and moving with it in space. The algorithm has been tested on the problem of the motion of a single particle and a pair of particles in a supersonic flow around a sphere.

Keywords: heterogenous supersonic flow; gas-dynamic particle—shock layer interaction; numerical simulation



Citation: Sposobin, A.; Reviznikov, D. A Meshless Algorithm for Modeling the Gas-Dynamic Interaction between High-Inertia Particles and a Shock Layer. *Fluids* **2023**, *8*, 53. <https://doi.org/10.3390/fluids8020053>

Academic Editors: Mehrdad Massoudi, Maša Knez Marevci and Darija Čör Andrejč

Received: 31 December 2022

Revised: 26 January 2023

Accepted: 30 January 2023

Published: 2 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A dispersed admixture in a supersonic flow may have a multifactorial effect on a streamlined body: direct impacts with the surface, contributing to energy transfer from particles to the body surface and erosive destruction [1], the reverse effect of particles on the gas flow [2,3] and convective heat flux enhancement [4,5], and radiative heat transfer between particles and the body surface [6]. A detailed review of publications on dusty flows around bodies is given in [7,8].

One of the important problems connected with supersonic heterogeneous flows is the gas dynamical interaction of high-inertia particles and a shock layer. By high-inertia particle, we mean relatively large particles that, having reflected from the body surface, can overcome the shock layer and go beyond the bow shock. As experiments have shown [9–11], when moving towards a supersonic oncoming flow, such particles can significantly rearrange the flow pattern, change the form and position of the bow shock, and contribute to a multiple increase in the convective heat flux to the body surface. In our previous work, we studied this process numerically. A detailed description of the resulting wave and vortex patterns is given in the papers [12–14].

It should be noted that the process of gas-dynamic interaction of particles with a shock layer is extremely difficult for numerical simulation. This is due to the mobility of particles and a significant difference in scales—the linear dimensions of the streamlined body and even relatively large particles differ by several orders of magnitude. In [12–14], we used the finite volume method on adaptive Cartesian grids. It allowed us to perform numerical simulation in both the axisymmetric and planar two-dimensional formulations. However, the use of Cartesian grids and the need for a detailed resolution of the boundary layer make it difficult to carry out three-dimensional calculations due to the very large number of computational cells.

There are several effective numerical approaches for solving gas dynamic problems with moving objects. Chimera-type overlapping grid technology [15,16], which uses finite volume or finite difference methods, is based on combining several high-resolution grids related to each of the moving objects and adapted them to their geometry into a single computational grid. The implementations of this approach on structured and unstructured grids are known [17,18]. Another approach involves the use of sliding grids [19–21]. It is based on the interpolation of gas flow parameters obtained by calculation on one grid in order to transfer them to another grid, moving relative to the first one, in overlapping areas. This approach has been applied for modeling rotating turbine blades, helicopter propellers, and solving other problems with known trajectories of moving objects [22,23]. In [14], we used sliding grids to calculate the single particle interaction with the shock layer. A serious problem with the considered approaches is the complexity of modeling the interaction between the objects. We face such situations when reproducing collisions between particles or their reflections from the surface.

Therefore, when constructing a three-dimensional computational model, a significantly different approach was chosen, which is based on a meshless method for solving a system of gas dynamics equations [24]. Unlike the finite volume method, which divides the entire calculation area into closed cells, the meshless method uses a finite set of points—calculation nodes. Nodes can be distributed in space in an anisotropic manner, which significantly saves computational resources [24]. Two methods for approximating derivatives have become widespread: based on radial basis functions [25–27] and the least squares method [28–30]. The latest one is used in this article. This method, despite its relative simplicity, showed a satisfactory agreement between the calculation results and the reference data when solving model problems of supersonic viscous and inviscid flows around a body.

In Section 2, we give a detailed description of the meshless algorithm. To calculate the gas-dynamic interaction of particles with the shock layer, it is important to take into account the large difference in scales of flow around the particles and around the body. For this purpose, we propose the following approach: Along with the stationary main set of nodes, each moving particle is surrounded by its own local cloud of points belonging to its domain. Clouds of computing nodes moving in space together with particles interact with each other, forming a single connected cloud of nodes. When calculating the fluxes, the nodes' belonging to different domains and their relative velocities are taken into account. The algorithm verification is carried out. The results of calculating the motion of a particle in a gas at rest and the free flow around a stationary particle at the same relative velocity turned out to be identical. In Section 3, we present the results of the numerical simulation of the motion of a single particle and a pair of particles in a supersonic flow around a sphere.

2. Materials and Methods

Governing equations

The model of the viscous heat-conducting gas flow in three-dimensional space includes a system of non-stationary Navier–Stokes equations in combination with the equation of state for ideal gas:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{q})}{\partial y} + \frac{\partial \mathbf{H}(\mathbf{q})}{\partial z} = \frac{\partial \mathbf{F}^v(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{G}^v(\mathbf{q})}{\partial y} + \frac{\partial \mathbf{H}^v(\mathbf{q})}{\partial z} \tag{1}$$

$$p = \rho RT \tag{2}$$

$$\mathbf{q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{pmatrix}, \mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho uH \end{pmatrix}, \mathbf{G} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ \rho vH \end{pmatrix}, \mathbf{H} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ \rho wH \end{pmatrix} \tag{3}$$

$$\mathbf{F}^v = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ \tau_{xx}u + \tau_{xy}v + \tau_{xz}w - q_x \end{pmatrix}, \mathbf{G}^v = \begin{pmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ \tau_{yx}u + \tau_{yy}v + \tau_{yz}w - q_y \end{pmatrix} \tag{4}$$

$$\mathbf{H}^v = \begin{pmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ \tau_{zx}u + \tau_{zy}v + \tau_{zz}w - q_z \end{pmatrix}$$

$$H = e + \frac{p}{\rho}, e = \frac{p}{\rho(\gamma - 1)} + \frac{1}{2}(u^2 + v^2 + w^2) \tag{5}$$

where t is the time, ρ is the density, p is the pressure, T is the temperature, $u, v,$ and w are the components of the gas velocity vector along the coordinate axes $x, y,$ and z, R, γ is the heat capacity ratio H is the total gas $\mathbf{F}, \mathbf{G},$ and \mathbf{H} are the inviscid flux $\mathbf{F}^v, \mathbf{G}^v,$ and \mathbf{H}^v are the viscous flux vectors along the coordinate axes.

Components of the viscous stress tensor are:

$$\begin{aligned}
 \tau_{xx} &= \frac{2}{3}\mu \left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z} \right), \tau_{yy} = \frac{2}{3}\mu \left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z} \right), \tau_{zz} = \frac{2}{3}\mu \left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right) \\
 \tau_{xy} = \tau_{yx} &= \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), \tau_{xz} = \tau_{zx} = \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right), \tau_{yz} = \tau_{zy} = \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)
 \end{aligned} \tag{6}$$

The heat flux density is determined by a vector with components:

$$q_x = -\lambda \frac{\partial T}{\partial x}, q_y = -\lambda \frac{\partial T}{\partial y}, q_z = -\lambda \frac{\partial T}{\partial z} \tag{7}$$

The value of dynamic viscosity is calculated using the well-known Sutherland formula:

$$\mu = \mu^* \left(\frac{T}{T^*} \right)^{\frac{3}{2}} \frac{T^* + C}{T + C} \tag{8}$$

$$\mu^* = 0.0000178 \frac{H \cdot c}{M^2}, T^* = 273.15 K, C^* = 110.4 K \text{ for air.}$$

The thermal conductivity coefficient is proportional to the dynamic viscosity:

$$\lambda = \frac{C_p \mu}{Pr} \tag{9}$$

where C_p is the specific heat capacity of the gas at constant pressure and Pr is the Prandtl number.

At the entrance to the computational domain, the Dirichlet boundary conditions are set, which determine the incoming supersonic flow with a fixed temperature $T_\infty,$ pressure, and velocity. At the output—the Neumann conditions $\frac{\partial \mathbf{q}}{\partial \mathbf{n}} = 0$ are set, where \mathbf{n} is the outer normal to the boundary.

The body surface is considered isothermal with a known temperature $T_w.$ No-slip and impermeability conditions $\mathbf{v} = 0$ are set here, as well as the condition $\frac{\partial p}{\partial \mathbf{n}} = 0.$

Meshless Method

To represent the field of gas-dynamic quantities in the computational domain, a finite set of discrete points with a fixed location in space is formed. Near the body surface, the points cluster in the normal direction in order to resolve the flow in the boundary layer in detail (see Figure 1).

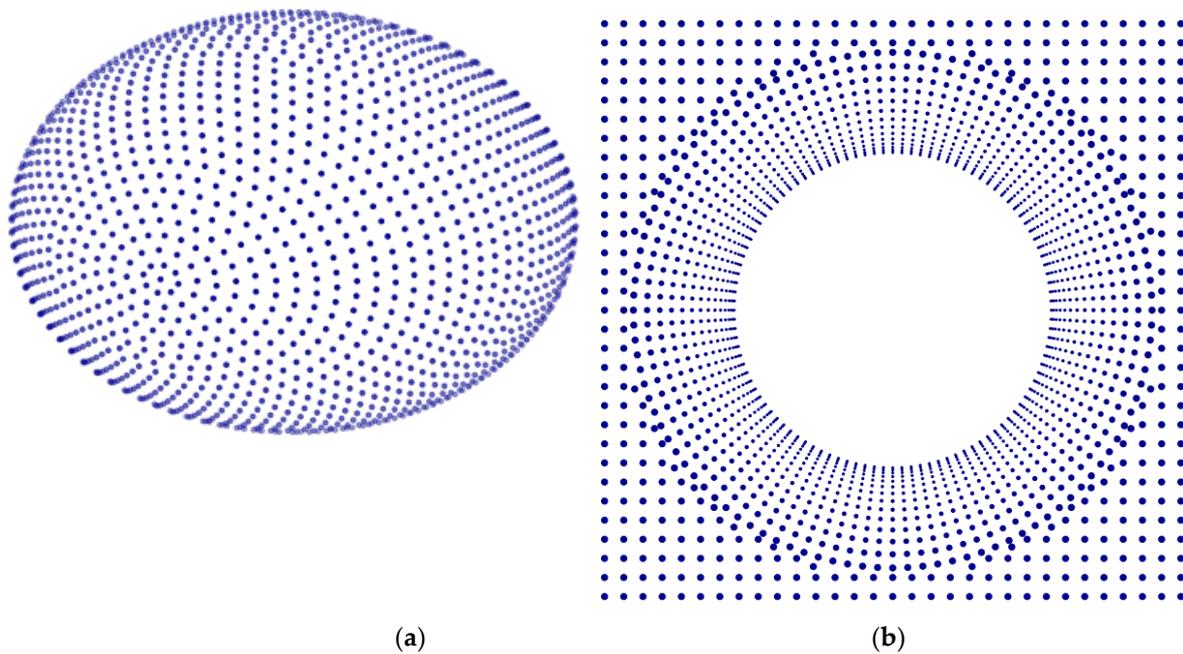


Figure 1. Location of computing nodes: on the body surface (a), in the section of the computational domain (b).

The least square method serves as the basis for approximating a scalar function $\varphi = \varphi(x, y, z)$ while having the values in a discrete set of points. For each computational node i and the cloud of neighboring nodes surrounding it $j \in C_i$ (see Figure 2), approximate equalities are written:

$$\varphi_j = \varphi_i + \Delta x_{ij} \left. \frac{\partial \varphi}{\partial x} \right|_i + \Delta y_{ij} \left. \frac{\partial \varphi}{\partial y} \right|_i + \Delta z_{ij} \left. \frac{\partial \varphi}{\partial z} \right|_i + O(h^2) \tag{10}$$

$$\Delta x_{ij} = x_j - x_i, \Delta y_{ij} = y_j - y_i, \Delta z_{ij} = z_j - z_i, \Delta \varphi_{ij} = \varphi_j - \varphi_i \tag{11}$$

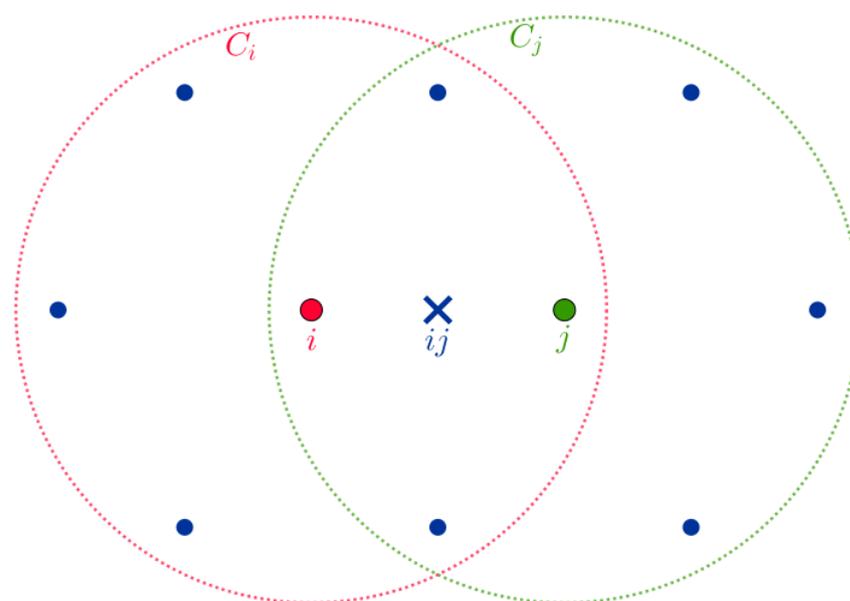


Figure 2. Clouds of computing points surrounding nodes i and j .

According to the least square method, the optimal approximation of the partial derivatives $\left. \frac{\partial \varphi}{\partial x} \right|_i, \left. \frac{\partial \varphi}{\partial y} \right|_i, \left. \frac{\partial \varphi}{\partial z} \right|_i$ is achieved by minimizing the functional

$$\sum_{j \in C_i} \omega_{ij} \left(\Delta \varphi_{ij} - \Delta x_{ij} \left. \frac{\partial \varphi}{\partial x} \right|_i - \Delta y_{ij} \left. \frac{\partial \varphi}{\partial y} \right|_i - \Delta z_{ij} \left. \frac{\partial \varphi}{\partial z} \right|_i \right)^2 \rightarrow \min \tag{12}$$

where the weight coefficients ω_{ij} are inversely proportional to the distance d_{ij} between a node and its neighbors:

$$\omega_{ij} = \frac{1}{d_{ij}}, \quad d_{ij} = \sqrt{\Delta x_{ij}^2 + \Delta y_{ij}^2 + \Delta z_{ij}^2} \tag{13}$$

The values of the linear combination coefficients can be obtained by solving a system of linear equations:

$$\left. \frac{\partial \varphi}{\partial x} \right|_i = \sum_{j \in C_i} \alpha_{ij} \Delta \varphi_{ij}, \quad \left. \frac{\partial \varphi}{\partial y} \right|_i = \sum_{j \in C_i} \beta_{ij} \Delta \varphi_{ij}, \quad \left. \frac{\partial \varphi}{\partial z} \right|_i = \sum_{j \in C_i} \gamma_{ij} \Delta \varphi_{ij} \tag{14}$$

$$\begin{bmatrix} \alpha_{ij} \\ \beta_{ij} \\ \gamma_{ij} \end{bmatrix} = \begin{bmatrix} \sum_{k \in C_i} \omega_{ik} \Delta x_{ik}^2 & \sum_{k \in C_i} \omega_{ik} \Delta x_{ik} \Delta y_{ik} & \sum_{k \in C_i} \omega_{ik} \Delta x_{ik} \Delta z_{ik} \\ \sum_{k \in C_i} \omega_{ik} \Delta x_{ik} \Delta y_{ik} & \sum_{k \in C_i} \omega_{ik} \Delta y_{ik}^2 & \sum_{k \in C_i} \omega_{ik} \Delta y_{ik} \Delta z_{ik} \\ \sum_{k \in C_i} \omega_{ik} \Delta x_{ik} \Delta z_{ik} & \sum_{k \in C_i} \omega_{ik} \Delta y_{ik} \Delta z_{ik} & \sum_{k \in C_i} \omega_{ik} \Delta z_{ik}^2 \end{bmatrix}^{-1} \begin{bmatrix} \omega_{ij} \Delta x_{ij} \\ \omega_{ij} \Delta y_{ij} \\ \omega_{ij} \Delta z_{ij} \end{bmatrix} \tag{15}$$

In the numerical solution of the system of gas dynamics equations, the components of the velocity vector, pressure, temperature, gas density, and complex functions containing them, as well as components of viscous and convective fluxes, act as function φ . The system of Navier–Stokes equations in a semi-discrete form takes the form:

$$\begin{aligned} & \frac{\partial \mathbf{q}_i}{\partial t} + 2 \sum_{j \in C_i} [\alpha_{ij} (\mathbf{F}_{ij} - \mathbf{F}_i) + \beta_{ij} (\mathbf{G}_{ij} - \mathbf{G}_i) + \gamma_{ij} (\mathbf{H}_{ij} - \mathbf{H}_i)] \\ & = 2 \sum_{j \in C_i} [\alpha_{ij} (\mathbf{F}_{ij}^v - \mathbf{F}_i^v) + \beta_{ij} (\mathbf{G}_{ij}^v - \mathbf{G}_i^v) + \gamma_{ij} (\mathbf{H}_{ij}^v - \mathbf{H}_i^v)] \end{aligned} \tag{16}$$

$$\mathbf{F}_i = \mathbf{F}(\mathbf{q}_i), \quad \mathbf{G}_i = \mathbf{G}(\mathbf{q}_i), \quad \mathbf{H}_i = \mathbf{H}(\mathbf{q}_i) \tag{17}$$

$$\mathbf{F}_i^v = \mathbf{F}^v(\mathbf{q}_i), \quad \mathbf{G}_i^v = \mathbf{G}^v(\mathbf{q}_i), \quad \mathbf{H}_i^v = \mathbf{H}^v(\mathbf{q}_i) \tag{18}$$

The calculation of the convective flux vectors $\mathbf{F}_{ij}, \mathbf{G}_{ij}, \mathbf{H}_{ij}$ in the middle of the ij segment connecting the nodes i and j , according to the AUSMPW+ scheme [31], requires two passes over the set of points. At the first stage, the vectors of conservative variables $\mathbf{q}^- = \mathbf{q}(\Psi_{ij}^-), \mathbf{q}^+ = \mathbf{q}(\Psi_{ij}^+)$ are reconstructed by applying the MUSCL scheme with the van Albada 2 limiter to the vector of primary variables Ψ [32]:

$$\Psi = (\rho, u, v, w, p)^T \tag{19}$$

$$\psi_{ij}^+ = \psi_i + \frac{s_i}{4} \left[(1 - ks_i) \Delta_{ij}^- + (1 + ks_i) (\psi_j - \psi_i) \right], \quad \Delta_{ij}^- = 2 \Delta \mathbf{r}_{ij} \bullet \nabla \psi_i - (\psi_j - \psi_i) \tag{20}$$

$$\psi_{ij}^- = \psi_j - \frac{s_j}{4} \left[(1 - ks_j) \Delta_{ij}^+ + (1 + ks_j) (\psi_j - \psi_i) \right], \quad \Delta_{ij}^+ = 2 \Delta \mathbf{r}_{ij} \bullet \nabla \psi_j - (\psi_j - \psi_i) \tag{21}$$

$$s_i = \max \left(0, \frac{2 \Delta_{ij}^- (\psi_j - \psi_i) + \varepsilon}{\Delta_{ij}^{-2} + (\psi_j - \psi_i)^2 + \varepsilon} \right), \quad s_j = \max \left(0, \frac{2 \Delta_{ij}^+ (\psi_j - \psi_i) + \varepsilon}{\Delta_{ij}^{+2} + (\psi_j - \psi_i)^2 + \varepsilon} \right) \tag{22}$$

$$\mathbf{r}_{ij} = \begin{pmatrix} x_j - x_i \\ y_j - y_i \\ z_j - z_i \end{pmatrix}, \nabla \psi_n = \begin{pmatrix} \frac{\partial \psi}{\partial x} \\ \frac{\partial \psi}{\partial y} \\ \frac{\partial \psi}{\partial z} \end{pmatrix}_n = \begin{pmatrix} \sum_{m \in C_n} \alpha_{mn} (\psi_m - \psi_n) \\ \sum_{m \in C_n} \beta_{mn} (\psi_m - \psi_n) \\ \sum_{m \in C_n} \gamma_{mn} (\psi_m - \psi_n) \end{pmatrix} \quad (23)$$

where the coefficient k determines the approximation order of the reconstruction scheme, at $k = \frac{1}{3}$ the third order is realized, and at $k = 0$ or $k = \pm 1$ the second order takes place. In this work, we use $k = \frac{1}{3}, \epsilon = 10^{-13}$.

For each pair of nodes, i and j , the minimum pressure value $p_{\min} = \min_{k \in \{i,j\}, m \in C_k} (\min(p_{km}^-, p_{km}^+))$ and the coefficient $f_p = \min_{k \in \{i,j\}, m \in C_k} \left(\min \left(\frac{p_{km}^+}{p_{km}^-}, \frac{p_{km}^-}{p_{km}^+} \right) \right)$ at all interfaces between the nodes i and j and their immediate neighbors are calculated. Here p^- and p^+ are the pressure values for \mathbf{q}^- and \mathbf{q}^+ , respectively.

At the second stage, using the values p_{\min} and f_p , and the state vectors \mathbf{q}^+ and \mathbf{q}^- for a pair of nodes i and j , the calculation of the convective fluxes vectors is performed. To generalize the presented method, we introduce the vectors $\boldsymbol{\eta}$ and \mathbf{N} , depending on the flux vector direction:

$$\mathbf{N} = (0 \quad \eta_x \quad \eta_y \quad \eta_z \quad 0)^T, \boldsymbol{\eta} = \begin{cases} (1 \ 0 \ 0)^T & \partial_{\text{ЛЯ}} \mathbf{F}_{ij} \\ (0 \ 1 \ 0)^T & \partial_{\text{ЛЯ}} \mathbf{G}_{ij} \\ (0 \ 0 \ 1)^T & \partial_{\text{ЛЯ}} \mathbf{H}_{ij} \end{cases} \quad (24)$$

Then the normal projection of the velocity is $\vartheta = \mathbf{v} \cdot \boldsymbol{\eta}$. The procedure for calculating the convective flux vector \mathbf{F}_{ij} along the x axis is described in detail below. In this case, $\vartheta = u$. Algorithms for calculating vectors $\mathbf{G}_{ij}, \mathbf{H}_{ij}$ are similar to the one given below, taking into account the vector $\boldsymbol{\eta}$ and velocity projections ($\vartheta = v$ и $\vartheta = w$).

Depending on the direction of the vector, $\mathbf{r}_{ij}, \mathbf{q}_L,$ and \mathbf{q}_R are selected:

$$\mathbf{q}_L = \begin{cases} \mathbf{q}^+, x_i \leq x_j \\ \mathbf{q}^-, x_i > x_j \end{cases}, \mathbf{q}_R = \begin{cases} \mathbf{q}^-, x_i \leq x_j \\ \mathbf{q}^+, x_i > x_j \end{cases} \quad (25)$$

The enthalpy, speed of sound, and Mach numbers are calculated:

$$H_{\text{normal}} = \frac{1}{2} \left(H_L - \frac{|\mathbf{v}_{\tau L}|^2}{2} + H_R - \frac{|\mathbf{v}_{\tau R}|^2}{2} \right), c_s = \sqrt{2 \frac{\gamma - 1}{\gamma + 1} H_{\text{normal}}} \quad (26)$$

$$c_{1/2} = \begin{cases} \frac{c_s^2}{\max(|\vartheta_L|, c_s)}, \vartheta_L + \vartheta_R \geq 0 \\ \frac{c_s^2}{\max(|\vartheta_R|, c_s)}, \vartheta_L + \vartheta_R < 0 \end{cases}, M_L = \frac{\vartheta_L}{c_{1/2}}, M_R = \frac{\vartheta_R}{c_{1/2}} \quad (27)$$

where $H = e + \frac{p}{\rho}$ is the total enthalpy and $\mathbf{v}_{\tau} = \mathbf{v} - \vartheta \boldsymbol{\eta}$ is the tangential velocity component.

Coefficients are calculated as follows:

$$M_L^+ = \begin{cases} \frac{(M_L + 1)^2}{4M_L}, & |M_L| \leq 1 \\ \frac{M_L + |M_L|}{2}, & |M_L| > 1 \end{cases}, M_R^- = \begin{cases} -\frac{(M_R - 1)^2}{4M_R}, & |M_R| \leq 1 \\ \frac{M_R - |M_R|}{2}, & |M_R| > 1 \end{cases} \quad (28)$$

$$m_{1/2} = M_L^+ + M_R^- \quad (29)$$

$$P_L^+ |_{\alpha=\frac{3}{16}} = \begin{cases} \frac{1}{4} (M_L + 1)^2 (2 - M_L) + \alpha M_L (M_L^2 - 1)^2, & |M_L| \leq 1 \\ \frac{1}{2} (1 + \text{sign}(M_L)), & |M_L| > 1 \end{cases} \quad (30)$$

$$P_R^- |_{\alpha=\frac{3}{16}} = \begin{cases} \frac{1}{4} (M_R - 1)^2 (2 + M_R) - \alpha M_R (M_R^2 - 1)^2, & |M_R| \leq 1 \\ \frac{1}{2} (1 - \text{sign}(M_R)), & |M_R| > 1 \end{cases} \quad (31)$$

$$p_s = P_L^+ p_L + P_R^- p_R \quad (32)$$

$$f_L = \begin{cases} 0, & p_s = 0 \\ \left(\frac{p_L}{p_s} - 1\right) \min\left(1, \frac{p_{\min}}{\min(p_L, p_R)}\right)^2, & p_s \neq 0 \end{cases} \quad (33)$$

$$f_R = \begin{cases} 0, & p_s = 0 \\ \left(\frac{p_R}{p_s} - 1\right) \min\left(1, \frac{p_{\min}}{\min(p_L, p_R)}\right)^2, & p_s \neq 0 \end{cases} \quad (34)$$

$$\omega = 1 - f_p^3 \quad (35)$$

$$\overline{M}_L^+ = \begin{cases} M_L^+ + M_R^- \cdot [(1 - \omega)(1 + f_R) - f_L], & m_{1/2} \geq 0 \\ M_L^+ \cdot \omega \cdot (1 + f_L), & m_{1/2} < 0 \end{cases} \quad (36)$$

$$\overline{M}_R^- = \begin{cases} M_R^- \cdot \omega \cdot (1 + f_R), & m_{1/2} \geq 0 \\ M_R^- + M_L^+ \cdot [(1 - \omega)(1 + f_L) - f_R], & m_{1/2} < 0 \end{cases} \quad (37)$$

The expression for calculating the convective flux \mathbf{F}_{ij} has the form

$$\mathbf{F}_{ij} = \overline{M}_L^+ \cdot c_{1/2} \cdot \Phi_L + \overline{M}_R^- \cdot c_{1/2} \cdot \Phi_R + p_s \cdot \mathbf{N}, \quad \Phi = (\rho \quad \rho u \quad \rho v \quad \rho w \quad \rho H)^T \quad (38)$$

Partial spatial derivatives of temperature and velocity components are required to calculate the elements of the viscous stress tensor and the heat flux vector:

$$\frac{\partial u}{\partial x}\Big|_i = \sum_{j \in C_i} \alpha_{ij}(u_j - u_i), \quad \frac{\partial u}{\partial y}\Big|_i = \sum_{j \in C_i} \beta_{ij}(u_j - u_i), \quad \dots, \quad \frac{\partial T}{\partial z}\Big|_i = \sum_{j \in C_i} \gamma_{ij}(T_j - T_i) \quad (39)$$

$$\tau_{xx}|_i = \frac{2}{3}\mu_i \left(2 \frac{\partial u}{\partial x}\Big|_i - \frac{\partial v}{\partial y}\Big|_i - \frac{\partial w}{\partial z}\Big|_i\right), \quad \tau_{xy}|_i = \mu_i \left(\frac{\partial u}{\partial y}\Big|_i + \frac{\partial v}{\partial x}\Big|_i\right), \quad \dots, \quad q_z|_i = -\lambda_i \frac{\partial T}{\partial z}\Big|_i \quad (40)$$

The obtained values are directly used to calculate the viscous flux vectors $\mathbf{F}_i^v = \mathbf{F}^v(\mathbf{q}_i)$, $\mathbf{G}_i^v = \mathbf{G}^v(\mathbf{q}_i)$, $\mathbf{H}_i^v = \mathbf{H}^v(\mathbf{q}_i)$ in the node i according to the above expressions.

The reconstruction of the gradient vectors of physical variables u, v, w, T is required to calculate the viscous fluxes $\mathbf{F}_{ij}^v, \mathbf{G}_{ij}^v, \mathbf{H}_{ij}^v$ in the middle of the ij segment, connecting the nodes i and j . It is performed according to [28]:

$$\nabla \varphi|_{ij} = \overline{\nabla \varphi|_{ij}} - \left(\overline{\nabla \varphi|_{ij}} \cdot \frac{\mathbf{r}_{ij}}{|\mathbf{r}_{ij}|} - \frac{\varphi_j - \varphi_i}{|\mathbf{r}_{ij}|}\right) \frac{\mathbf{r}_{ij}}{|\mathbf{r}_{ij}|}, \quad \overline{\nabla \varphi|_{ij}} = \frac{\nabla \varphi|_i + \nabla \varphi|_j}{2} \quad (41)$$

Let us give some expressions:

$$\begin{aligned} \frac{\partial u}{\partial y}\Big|_{ij} &= \overline{\frac{\partial u}{\partial y}\Big|_{ij}} - \left(\overline{\frac{\partial u}{\partial x}\Big|_{ij}}(x_j - x_i) + \overline{\frac{\partial u}{\partial y}\Big|_{ij}}(y_j - y_i) + \overline{\frac{\partial u}{\partial z}\Big|_{ij}}(z_j - z_i) - \frac{u_j - u_i}{|\mathbf{r}_{ij}|}\right) \frac{y_j - y_i}{|\mathbf{r}_{ij}|}, \\ \frac{\partial u}{\partial x}\Big|_{ij} &= \frac{1}{2} \left(\frac{\partial u}{\partial x}\Big|_i + \frac{\partial u}{\partial x}\Big|_j\right), \quad \frac{\partial u}{\partial y}\Big|_{ij} = \frac{1}{2} \left(\frac{\partial u}{\partial y}\Big|_i + \frac{\partial u}{\partial y}\Big|_j\right), \quad \frac{\partial u}{\partial z}\Big|_{ij} = \frac{1}{2} \left(\frac{\partial u}{\partial z}\Big|_i + \frac{\partial u}{\partial z}\Big|_j\right), \\ \frac{\partial v}{\partial x}\Big|_{ij} &= \overline{\frac{\partial v}{\partial x}\Big|_{ij}} - \left(\overline{\frac{\partial v}{\partial x}\Big|_{ij}}(x_j - x_i) + \overline{\frac{\partial v}{\partial y}\Big|_{ij}}(y_j - y_i) + \overline{\frac{\partial v}{\partial z}\Big|_{ij}}(z_j - z_i) - \frac{v_j - v_i}{|\mathbf{r}_{ij}|}\right) \frac{x_j - x_i}{|\mathbf{r}_{ij}|}, \\ \frac{\partial v}{\partial x}\Big|_{ij} &= \frac{1}{2} \left(\frac{\partial v}{\partial x}\Big|_i + \frac{\partial v}{\partial x}\Big|_j\right), \quad \frac{\partial v}{\partial y}\Big|_{ij} = \frac{1}{2} \left(\frac{\partial v}{\partial y}\Big|_i + \frac{\partial v}{\partial y}\Big|_j\right), \quad \frac{\partial v}{\partial z}\Big|_{ij} = \frac{1}{2} \left(\frac{\partial v}{\partial z}\Big|_i + \frac{\partial v}{\partial z}\Big|_j\right) \end{aligned} \quad (42)$$

Due to minor changes, the viscosity is determined by averaging the values in nodes [28]:

$$\mu_{ij} = \frac{\mu_i + \mu_j}{2} \quad (43)$$

then

$$\tau_{xy}|_{ij} = \mu_{ij} \left(\frac{\partial u}{\partial y}\Big|_{ij} + \frac{\partial v}{\partial x}\Big|_{ij}\right) \quad (44)$$

The remaining components of the viscous stress tensor are calculated in a similar way.

Numerical integration with respect to time is performed by the third-order explicit Runge–Kutta method [32]:

$$\frac{\partial \mathbf{q}_i}{\partial t} + \mathbf{R}(\mathbf{q}_i) = \mathbf{R}^v(\mathbf{q}_i) \tag{45}$$

$$\mathbf{R}(\mathbf{q}_i) = 2 \sum_{j \in \tilde{C}_i} [\alpha_{ij}(\mathbf{F}_{ij} - \mathbf{F}_i) + \beta_{ij}(\mathbf{G}_{ij} - \mathbf{G}_i) + \gamma_{ij}(\mathbf{H}_{ij} - \mathbf{H}_i)] \tag{46}$$

$$\mathbf{R}^v(\mathbf{q}_i) = 2 \sum_{j \in \tilde{C}_i} [\alpha_{ij}(\mathbf{F}_{ij}^v - \mathbf{F}_i^v) + \beta_{ij}(\mathbf{G}_{ij}^v - \mathbf{G}_i^v) + \gamma_{ij}(\mathbf{H}_{ij}^v - \mathbf{H}_i^v)] \tag{47}$$

$$\begin{aligned} \mathbf{q}_i^{(1)} &= \mathbf{q}_i^n - \Delta t \mathbf{R}(\mathbf{q}_i^n), \quad \mathbf{q}_i^{(2)} = \frac{3}{4} \mathbf{q}_i^n + \frac{1}{4} \mathbf{q}_i^{(1)} - \frac{1}{4} \Delta t \mathbf{R}(\mathbf{q}_i^{(1)}), \\ \mathbf{q}_i^{n+1} &= \frac{1}{3} \mathbf{q}_i^n + \frac{2}{3} \mathbf{q}_i^{(2)} - \frac{2}{3} \Delta t \mathbf{R}(\mathbf{q}_i^{(2)}) \end{aligned} \tag{48}$$

The time step for explicit integration is determined according to the Courant criterion:

$$\Delta t = \min \left(\frac{\text{CFL}}{\max_i |\lambda_i|}, \frac{\text{VNN}}{\max_i |\lambda_i^v|} \right) \tag{49}$$

$$\lambda_i = \sum_{j \in \tilde{C}_i} (\alpha_{ij} \hat{u}_{ij} + \beta_{ij} \hat{v}_{ij} + \gamma_{ij} \hat{w}_{ij} + \hat{c}_{ij} \sqrt{\alpha_{ij}^2 + \beta_{ij}^2 + \gamma_{ij}^2}) \tag{50}$$

$$\lambda_i^v = \max \left(\frac{4}{3}, \frac{\gamma_{ij}^2}{\text{Pr}} \right) \sum_{j \in \tilde{C}_i} \frac{2 \bar{\mu}_{ij} (\alpha_{ij}^2 + \beta_{ij}^2 + \gamma_{ij}^2)}{\rho_i + \rho_j} \tag{51}$$

$$\bar{\mu}_{ij} = \mu \left(\frac{T_i + T_j}{2} \right) \tag{52}$$

$$\text{CFL} = 0.5, \text{VNN} = 0.4 \tag{53}$$

by Roe-averaging vectors of physical variables for \mathbf{q}^+ and \mathbf{q}^-

$$\begin{aligned} \hat{\rho} &= \sqrt{\rho^- \rho^+}, \quad \hat{H} = \frac{H^- \sqrt{\rho^-} + H^+ \sqrt{\rho^+}}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad \hat{c} = \sqrt{\left\{ \hat{H} - \frac{\hat{u}^2 + \hat{v}^2 + \hat{w}^2}{2} \right\} (\gamma - 1)}, \\ \hat{u} &= \frac{u^- \sqrt{\rho^-} + u^+ \sqrt{\rho^+}}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad \hat{v} = \frac{v^- \sqrt{\rho^-} + v^+ \sqrt{\rho^+}}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad \hat{w} = \frac{w^- \sqrt{\rho^-} + w^+ \sqrt{\rho^+}}{\sqrt{\rho^-} + \sqrt{\rho^+}} \end{aligned} \tag{54}$$

The implementation of the Neumann boundary conditions is also based on the approximation of the derivative $\frac{\partial \varphi}{\partial \mathbf{n}}$ by the least squares method [28]:

$$\left. \frac{\partial \varphi}{\partial \mathbf{n}} \right|_i = \mathbf{n} \cdot \nabla \varphi = n_x \sum_{j \in \tilde{C}_i} \alpha_{ij} (\varphi_j - \varphi_i) + n_y \sum_{j \in \tilde{C}_i} \beta_{ij} (\varphi_j - \varphi_i) + n_z \sum_{j \in \tilde{C}_i} \gamma_{ij} (\varphi_j - \varphi_i) \tag{55}$$

$$\eta_{ij} = \alpha_{ij} n_x + \beta_{ij} n_y + \gamma_{ij} n_z \tag{56}$$

$$\left. \frac{\partial \varphi}{\partial \mathbf{n}} \right|_i = \sum_{j \in \tilde{C}_i} \eta_{ij} (\varphi_j - \varphi_i) = \sum_{j \in \tilde{C}_i} \eta_{ij} \varphi_j - \varphi_i \sum_{j \in \tilde{C}_i} \eta_{ij} \tag{57}$$

$$\varphi_i = \frac{\sum_{j \in \tilde{C}_i} \eta_{ij} \varphi_j - \left. \frac{\partial \varphi}{\partial \mathbf{n}} \right|_i}{\sum_{j \in \tilde{C}_i} \eta_{ij}} \tag{58}$$

where n_x, n_y, n_z are the components of the outward normal vector \mathbf{n} at node i on the surface boundary and \tilde{C}_i is the set of its neighboring nodes that do not belong to the boundary (see Figure 3).

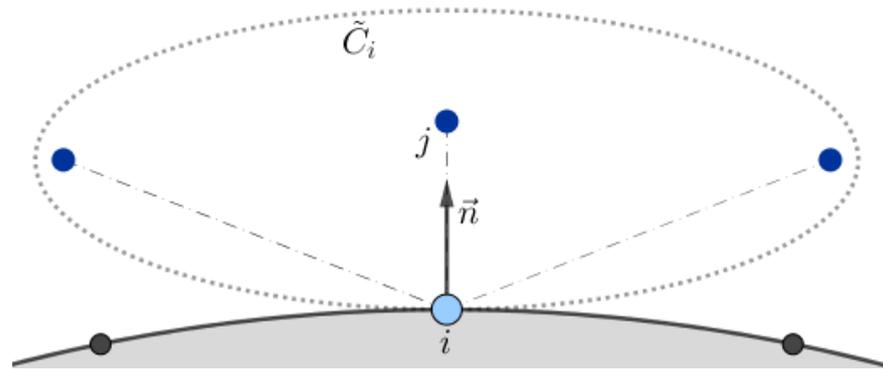


Figure 3. Implementation of the boundary conditions on the body surface.

The calculation of the components of the gas state vector at node i at the isothermal wall with temperature T_w is carried out according to the expressions:

$$T_i = T_w, p_i = \frac{\sum_{j \in \tilde{C}_i} \eta_{ij} p_j}{\sum_{j \in \tilde{C}_i} \eta_{ij}}, \rho_i = \frac{p_i}{R T_i}, u_i = v_i = w_i = 0, e_i = \frac{p_i}{\rho_i(\gamma - 1)} \quad (59)$$

Modeling the motion of particles in a gas flow

The developed computational model makes it possible to study the supersonic flow of a viscous gas in the presence of one or several relatively large moving particles. The flow around each object (the body and the particles) is calculated by solving the system of gas dynamics equations in its own coordinate system on a selected set of calculation nodes belonging to its domain. A body in a supersonic flow is considered immovable along with its coordinate system, the domain, and the cloud of nodes, which are further referred to as the “main”. On the inlet boundary of the main computational domain, the parameters of the oncoming flow are set (\mathbf{n} —the normal vector to the boundary):

$$\rho = \rho_\infty, T = T_\infty, \mathbf{v} = \mathbf{v}_\infty, \frac{\partial \rho}{\partial \mathbf{n}} = 0, \frac{\partial p}{\partial \mathbf{n}} = 0 \quad (60)$$

The Neumann boundary conditions are set at the exit from the main calculation area:

$$\frac{\partial \rho}{\partial \mathbf{n}} = 0, \frac{\partial \mathbf{v}}{\partial \mathbf{n}} = 0, \frac{\partial p}{\partial \mathbf{n}} = 0 \quad (61)$$

The aerodynamic drag force, \mathbf{f}_D , which determines the change in particle velocity, is calculated from the action of the viscous friction force and the gas pressure at the computational nodes on its surface:

$$\frac{d\mathbf{r}_p}{dt} = \mathbf{v}_p, m_p \frac{d\mathbf{v}_p}{dt} = \mathbf{f}_D, \mathbf{f}_D = \sum_{i \in S_p} \left(-p_i \mathbf{n}_i + \mu_i \frac{\partial \mathbf{v}_\tau}{\partial \mathbf{n}} \Big|_i \right) S_i \quad (62)$$

where m_p is the particle mass, \mathbf{v}_p is the particle velocity vector, and \mathbf{r}_p is the particle position vector in the central coordinate system. The boundary nodes lying on the particle surface S_p correspond to the surface elements with area S_i , and the outer normal vector \mathbf{n}_i , \mathbf{v}_τ is the tangential component of the gas velocity near the surface.

In the computational domain, a stationary set of computing nodes belonging to the main domain, adapted to the boundaries of the domain and the geometry of the object, is formed. Each particle is surrounded by a cloud of computational nodes that belong to its domain and move with it in space. The solution of the gas dynamics equations at the nodes connected with the particle is carried out in the local moving coordinate system, whereas at the nodes of the main domain—it is in the central fixed coordinate system. As the

particle moves in space, some of the computational nodes are temporarily excluded from the calculation, and the outer boundary nodes of the particle domain become neighbors with the nodes of the main domain or the domain of another particle (see Figure 4). When numerically integrating the equations using a meshless method, the calculation of viscous and convective fluxes between nodes belonging to different domains requires the transformation of the conservative variables' vectors, \mathbf{q} , as well as their gradients, $\nabla \mathbf{q}$, into the local coordinate system of the node for which the calculation is performed.

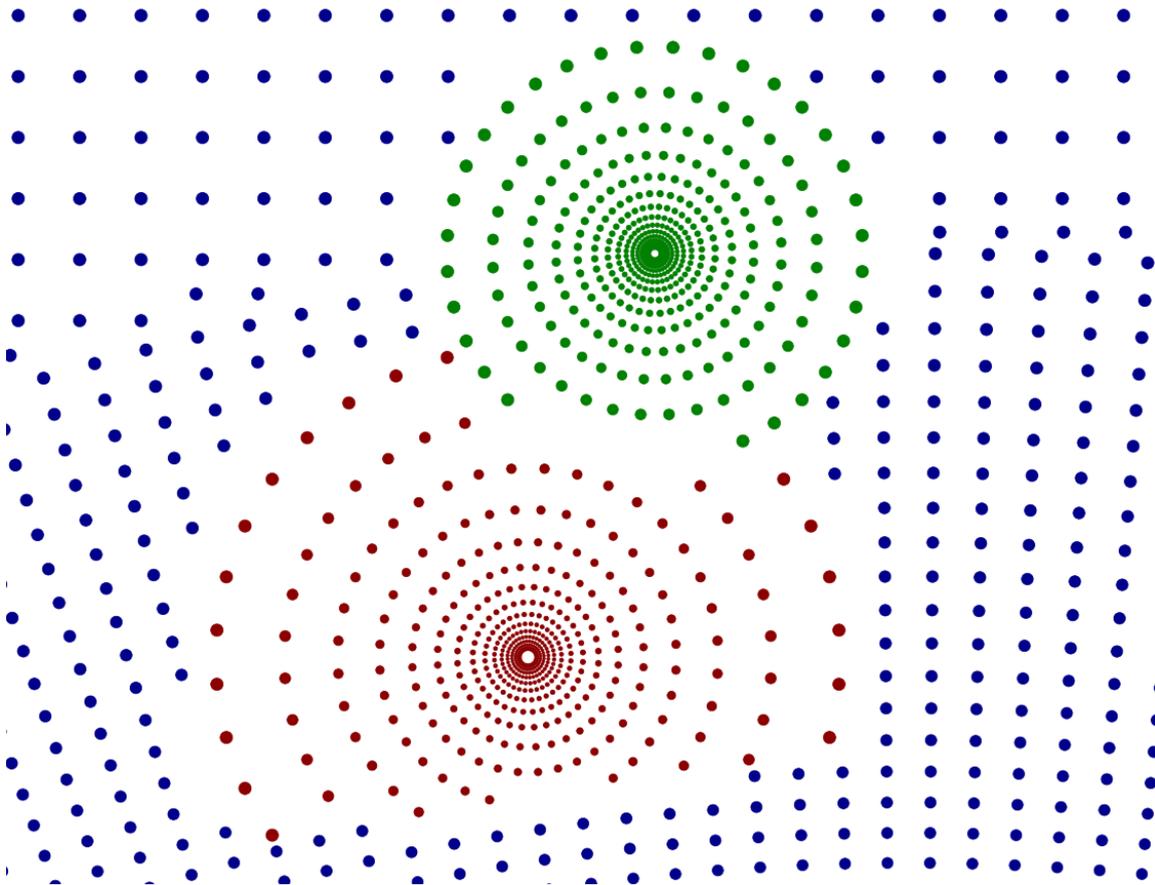


Figure 4. Formation of a single cloud from nodes belonging to different domains.

An approach based on the use of a single cloud of computational nodes makes it possible to simulate the gas-dynamic interaction of a particle with the body as well as the interaction between the particles.

The software implementation of the described algorithms is made in the C++ programming language in combination with the OpenCL parallel computing technology, which allows the use of heterogeneous parallel computing devices, including graphics processors from different manufacturers.

A series of computational experiments have been carried out to test the operation of the presented method for simulating gas flow around moving objects. In one of the series of tests, the flow around a sphere was calculated in the following modes:

- the sphere is at rest $M_p = 0$, the Mach number of the oncoming flow $M_\infty = 2$;
- the sphere moves against the oncoming flow ($M_\infty = 1$), the sphere velocity corresponds to Mach number $M_p = 1$
- the gas is stationary $M_\infty = 0$, the sphere moves with the velocity corresponding to Mach number $M_p = 2$

Reynolds number is $Re = \frac{\rho UL}{\mu} = 10^5$, where $L = D$ is the sphere diameter, and U is the gas velocity relative to the sphere. The shadow patterns of the gas flow (see Figure 5) as well as the pressure on the surface (see Figure 6) are identical in all three cases, which testifies the correctness of the computational model.

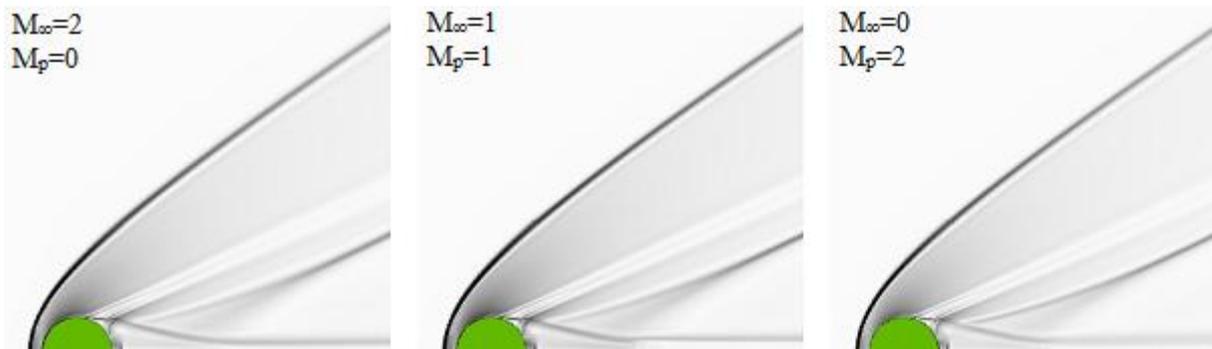


Figure 5. Shadow pattern of the flow around the sphere (single cloud of nodes).

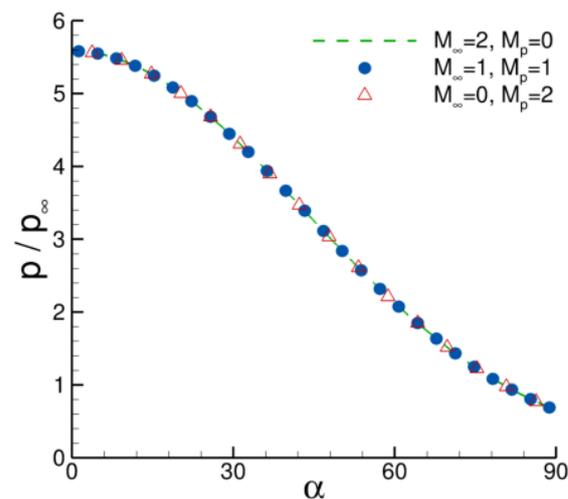


Figure 6. Distribution of the gas pressure along the sphere surface.

3. Results and Discussion

A computational experiment was carried out to simulate the motion of a large spherical particle with a diameter $d_p = 2 \cdot 10^{-4}$ m of silicon dioxide and a density $\rho_p = 2170 \frac{kg}{m^3}$ along the symmetry axis of a sphere with a diameter $D = 0.075$ m toward the oncoming flow. The Mach and Reynolds numbers of the oncoming flow are $M_\infty = 6$ and $Re|_{L=D} = 1.09 \cdot 10^6$. The initial velocity of the particle starting from the sphere surface is $100 \frac{m}{s}$. Figure 7 shows the shadow patterns of the flow at successive times $t_0 = 0$ s, $t_1 = 1 \cdot 10^{-4}$ s, $t_2 = 1.3 \cdot 10^{-4}$ s, $t_3 = 1.46 \cdot 10^{-4}$ s, $t_4 = 2.9 \cdot 10^{-4}$ s, $t_5 = 3.77 \cdot 10^{-4}$ s, and in Figure 8, the distributions of the pressure and convective heat flux along the surface are presented that correspond to those time moments. The angular coordinate here is counted from the critical point. The value of the heat flux is normalized by the value of the heat flux at the critical point, calculated with the well-known Fay–Riddell formula [33].

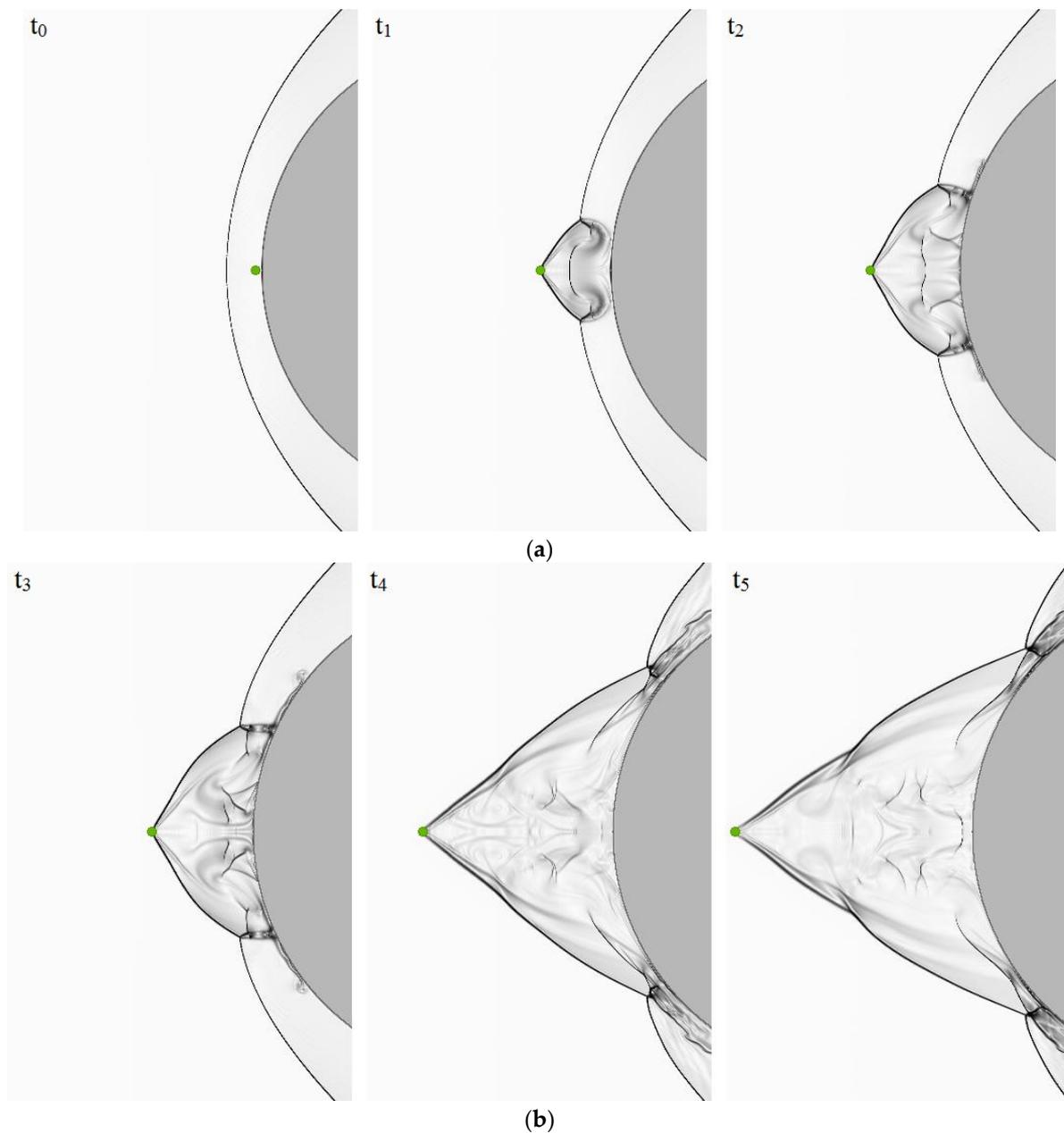


Figure 7. (a). Shadow pictures of the flow in the process of particle movement along the symmetry axis. (b). Shadow pictures of the flow in the process of particle movement along the symmetry axis (continued).

When the particle passes through the head shock wave, the stationary shock-wave structure is destroyed, resulting in the formation of a cone-shaped perturbation region, the top of which is moving along with the particle. In the zone of contact between the cone-shaped and bow shock waves, a λ -configuration of three waves arises with the formation of an impact jet directed towards the surface, which causes a significant increase in pressure and in the heat flux in the local area affected by the jet (see Figure 8). As the particle moves away from the surface, the wave configuration evolves, and the zone of their contact, together with the region of the intense gas impact on the surface, shifts to the periphery. A detailed analysis of the observed phenomena and mechanisms of their occurrence is given in [12], where a similar problem is solved by the finite volume method on adaptive Cartesian grids.

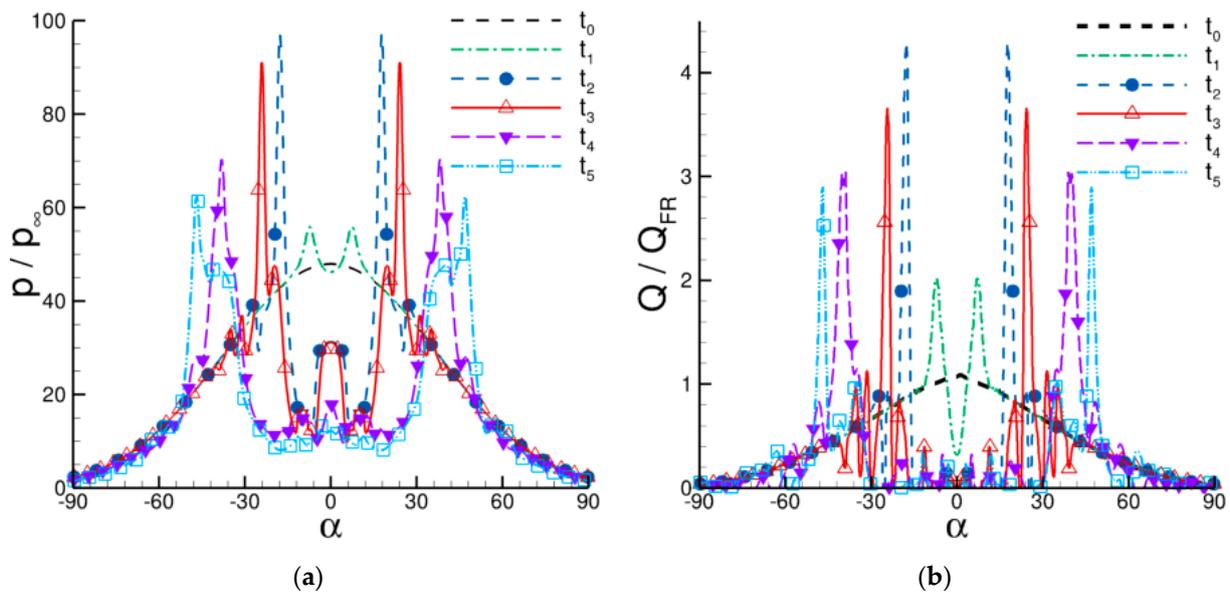


Figure 8. Oscillations of the pressure (a) and the convective heat flux (b) on the surface, induced by the gas-dynamic interaction of a particle with a shock layer.

Figure 9 shows the shadow patterns of the flow when two identical particles are sequentially launched at an angle to the symmetry axis. In the considered configuration, the second particle finds itself under the influence of the wake of the first particle, loses the velocity component faster, and, as a result, moves away from the sphere to a noticeably smaller distance before turning around and moving to the periphery.

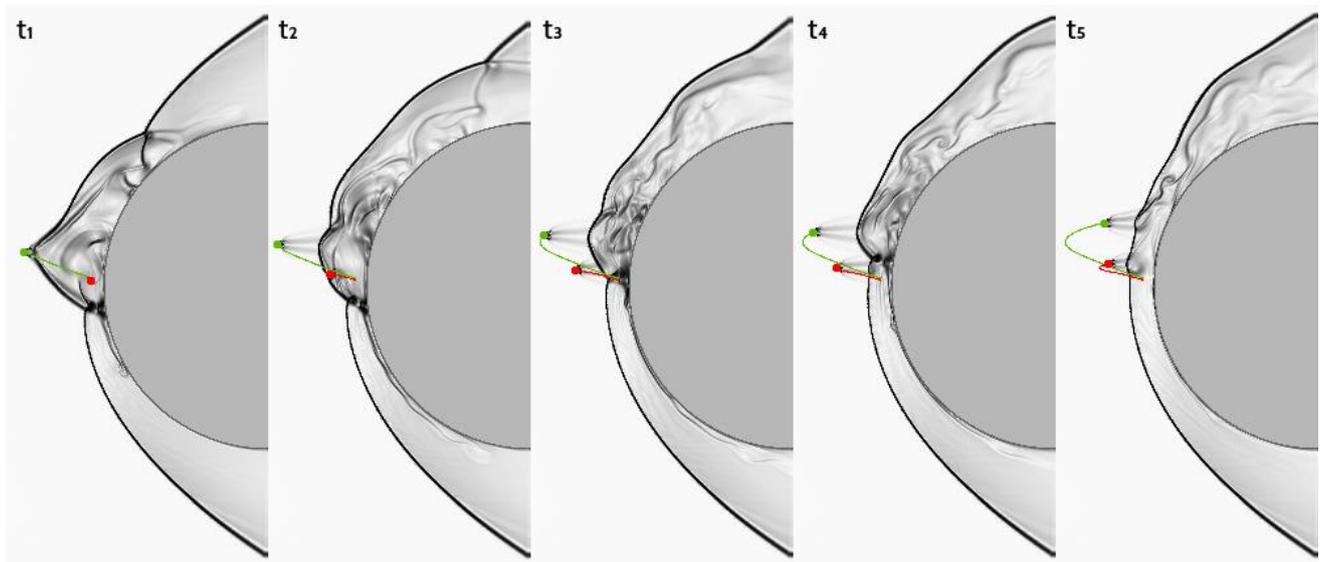


Figure 9. Shadow pictures of the flow under the influence of a pair of particles.

Figure 10 shows graphs of the pressure and the convective heat flux along the surface at the time instants corresponding to the flow patterns in Figure 9. There is an approximately two times increase in the pressure and an almost four times increase in the heat flux in a local area near the critical point. Significant perturbations of the gas parameters near the surface, leading to the appearance of regions of multiple enhancement of the convective heat flux, accompany the particles during their movement.

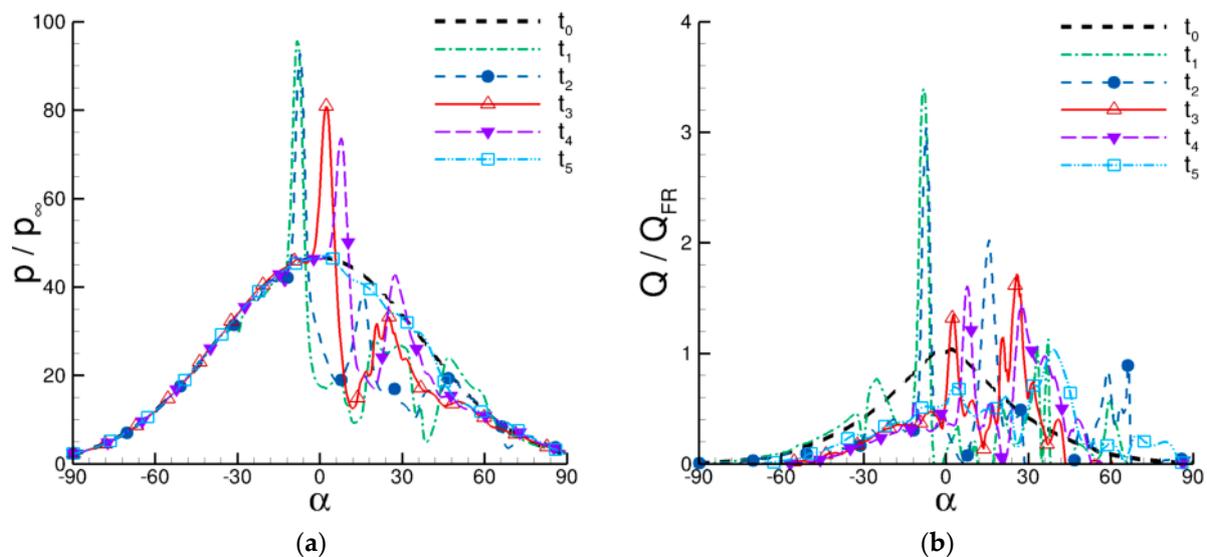


Figure 10. Pressure (a) and convective heat flux (b) oscillations on the sphere surface, induced by the gas-dynamic interaction of a pair of particles with a shock layer.

4. Conclusions

A meshless algorithm has been proposed for modeling the gas-dynamic interaction of moving particles with the shock layer. The algorithm allows for overcoming the problems connected with the large difference in scales of the flow around the particles and around the body. The algorithm is based on the approximation of partial derivatives by the least squares method on a set of nodes distributed in the calculation area. Along with the stationary main set of nodes, each moving particle is surrounded by its own cloud of points belonging to its domain. Clouds of computational nodes moving in space together with particles interact with each other, forming a single connected cloud of nodes. When calculating the fluxes, the nodes' belonging to different domains and their relative velocities are taken into account. Numerical experiments have been carried out to simulate the motion of a single particle and a pair of particles in a supersonic flow around a sphere. The developed algorithm opens up wide opportunities for the detailed study of the collective effects of a coarsely dispersed admixture on the flow and heat transfer in the shock layer for a blunt body in a supersonic flow.

Author Contributions: Methodology, software, and investigation, A.S.; methodology, investigation, and validation, D.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was carried out within the framework of the state assignment issued by the Ministry of Education and Science of Russia under project number FSFF-2020-0013.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Ershova, T.V.; Mikhatulin, D.S.; Reviznikov, D.L.; Sposobin, A.V.; Vinnikov, V.V. Numerical Simulation of Heat and Mass Transfer between Heterogeneous Flow and an Obstacle. *Comput. Therm. Sci.* **2011**, *3*, 15–30. [[CrossRef](#)]
- Tsirkunov, Y.M. Gas-particle flows around bodies—Key problems, modeling and numerical analysis. In Proceedings of the Fourth International Conference on Multiphase Flow—CD ROM Proc. ICMF'2001, New Orleans, LA, USA, 27 May–1 June 2001; Michaelides, E., Ed.; Paper No. 609.
- Molleson, G.V.; Stasenkov, A.L. Acceleration of Microparticles and their Interaction with a Solid Body. *High Temp.* **2017**, *55*, 906–913. [[CrossRef](#)]

4. Oesterlé, B.; Volkov, A.N.; Tsirkunov, Y.M. Numerical investigation of two-phase flow structure and heat transfer in a supersonic dusty gas flow over a blunt body. *Prog. Flight Phys.* **2013**, *5*, 441–456.
5. Osipov, A.N.; Egorova, L.A.; Sakharov, V.I.; Wang, B. Heat transfer in supersonic dusty-gas flow past a blunt body with inertial particle deposition effect. *Prog. Nat. Sci.* **2002**, *12*, 887–892.
6. Reviznikov, D.L.; Sposobin, A.V.; Dombrovsky, L.A. Radiative Heat Transfer from Supersonic Flow with Suspended Polydisperse Particles to a Blunt Body: Effect of Collisions between Particles. *Comput. Therm. Sci.* **2015**, *5*, 313–325. [[CrossRef](#)]
7. Varaksin, A.Y. Fluid dynamics and thermal physics of two-phase flows: Problems and achievements. *High Temp.* **2013**, *51*, 377–407. [[CrossRef](#)]
8. Varaksin, A.Y. Gas-Solid Flows Past Bodies. *High Temp.* **2018**, *56*, 275–295. [[CrossRef](#)]
9. Fleener, W.A.; Watson, R.H. Convective heating in dust-laden hypersonic flows. In Proceedings of the 8th Thermophysics Conference, Palm Springs, CA, USA, 16–18 July 1973; AIAA Paper 1973, No. 73-761.
10. Holden, M.S.; Duryea, G.; Gustafson, G.; Hudack, L. An Experimental Study of Particle-Induced Convective Heating Augmentation. In Proceedings of the 9th Fluid and Plasma Dynamics Conference, San Diego, CA, USA, 14–16 July 1976; AIAA Paper 1976, No. 76-320.
11. Vladimirov, A.S.; Ershov, I.V.; Makarevich, G.A.; Khodtsev, A.V. Experimental investigation of the process of interaction between heterogeneous flows and flying bodies. *High Temp.* **2008**, *46*, 512–517. [[CrossRef](#)]
12. Reviznikov, D.L.; Sposobin, A.V.; Ivanov, I.E. Change in the Structure of a Flow under the Action of Highly Inertial Particle when a Hypersonic Heterogeneous Flow Passes over a Body. *High Temp.* **2018**, *56*, 884–889. [[CrossRef](#)]
13. Reviznikov, D.L.; Sposobin, A.V.; Ivanov, I.E. Comparative Analysis of Calculated and Experimental Data on an Oscillating Flow Induced by the Gasdynamic Interaction of a Particle with a Shock Layer. *High Temp.* **2020**, *58*, 839–845. [[CrossRef](#)]
14. Sposobin, A.V.; Reviznikov, D.L. Impact of High Inertia Particles on the Shock Layer and Heat Transfer in a Heterogeneous Supersonic Flow around a Blunt Body. *Fluids* **2021**, *6*, 406. [[CrossRef](#)]
15. Benek, J.A.; Buning, P.G.; Steger, J.L. A 3-D Chimera Grid Embedding Technique. In Proceedings of the 7th Computational Physics Conference, Cincinnati, OH, USA, 15–17 July 1985; AIAA Paper 1985, No. 85-1523.
16. Wang, Z.J.; Parthasarathy, V. A Fully Automated Chimera Methodology for Multiple Moving Body Problems. *Int. J. Numer. Methods Fluids* **2000**, *33*, 919–938. [[CrossRef](#)]
17. Lee, K.R.; Park, J.H.; Kim, K.H. High-Order Interpolation Method for Overset Grid Based on Finite Volume Method. *AIAA J.* **2011**, *49*, 1387–1398. [[CrossRef](#)]
18. Deryugin, Y.N.; Sarazov, A.S.; Zhuchkov, R.N. Features of overset meshes methodology on unstructured grids. *Math. Model. Comput. Simul.* **2017**, *9*, 587–597. [[CrossRef](#)]
19. Bakhvalov, P.A.; Bobkov, V.G.; Kozubskaya, T.K. Application of schemes with a quasi-one-dimensional reconstruction of variables for calculations on nonstructured sliding grids. *Math Model. Comput Simul.* **2017**, *9*, 155–168. [[CrossRef](#)]
20. Yamakawa, M.; Chikaguchi, S.; Asao, S.; Hamato, S. Multi Axes Sliding Mesh Approach for Compressible Viscous Flows. In *Computational Science—ICCS 2020*; Krzhizhanovskaya, V., Závodszy, G., Lees, M.H., Dongarra, J.J., Sloot, P.M.A., Brissos, S., Teixeira, J., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12143, pp. 46–59.
21. Dürrwächter, J.; Kurz, M.; Kopper, P.; Kempf, D.; Munz, C.; Beck, A. An efficient sliding mesh interface method for high-order discontinuous Galerkin schemes. *Comput. Fluids* **2021**, *217*, 104825. [[CrossRef](#)]
22. Steijl, R.; Barakos, G. Computational Investigation of Rotor–Fuselage Interactional Aerodynamics using Sliding–Plane CFD Method. *AIAA J.* **2009**, *47*, 2143–2157. [[CrossRef](#)]
23. Nam, H.J.; Park, Y.; Kwon, O.J. Simulation of Unsteady Rotor–Fuselage Aerodynamic Interaction Using Unstructured Adaptive Meshes. *J. Am. Helicopter Soc.* **2006**, *51*, 141–149. [[CrossRef](#)]
24. Li, H.; Mulay, S.S. *Meshless Methods and Their Numerical Properties*; CRC Press: New York, NY, USA, 2013.
25. Vasilyev, A.N.; Kolbin, I.S.; Reviznikov, D.L. Meshfree computational algorithms based on normalized radial basis functions. In *Advances in Neural Networks*; Cheng, L., Liu, Q., Ronzhin, A., Eds.; Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 9719, pp. 583–591.
26. Li, J.; Hon, Y. Domain Decomposition for Radial Basis Meshless Methods. *Numer. Methods Part. Differ. Equ.* **2004**, *20*, 450–462. [[CrossRef](#)]
27. Tolstykh, A.I.; Shirobokov, D.A. Meshless method based on radial basis functions. *Comput. Math. Math. Phys.* **2005**, *45*, 1447–1454.
28. Hashemi, M.Y.; Jahangirian, A. Implicit fully mesh-less method for compressible viscous flow calculations. *J. Comput. Appl. Math.* **2011**, *235*, 4687–4700. [[CrossRef](#)]
29. Sattarzadeh, S.; Jahangirian, A. 3D implicit mesh-less method for compressible flow calculations. *Sci. Iran.* **2012**, *19*, 503–512. [[CrossRef](#)]
30. Sattarzadeh, S.; Jahangirian, A.; Hashemi, M.Y. Unsteady Compressible Flow Calculations with Least-Square Mesh-less Method. *J. Appl. Fluid Mech.* **2016**, *9*, 233–241. [[CrossRef](#)]
31. Kim, K.H.; Kim, C.; Rho, O.H. Methods for the accurate computations of hypersonic flows I. AUSMPW+ Scheme. *J. Comput. Phys.* **2001**, *174*, 38–80. [[CrossRef](#)]

32. Wang, Y.; Cai, X.; Zhang, M.; Ma, X.; Ren, D.; Tan, J. The study of the three-Dimensional meshless solver based on AUSM+-up and MUSCL scheme. In Proceedings of the 2015 International Conference on Electromechanical Control Technology and Transportation, Zhuhai, China, 31 October–1 November 2015. [[CrossRef](#)]
33. Fay, J.A.; Riddell, F.R. Theory of stagnation point heat transfer in dissociated air. *J. Aeronaut. Sci.* **1958**, *25*, 73–85. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.