

# Modified Method of Adaptive Artificial Viscosity for Solution of Gas Dynamics Problems on Parallel Computer Systems

Igor Popov<sup>1,2,\*</sup> and Sergey Sukov<sup>1,\*\*</sup>

<sup>1</sup> Keldysh Institute of Applied Mathematics of RAS  
4 Miusskaya square, 125047, Moscow, Russia

<sup>2</sup> National Research Nuclear University MEPhI  
31 Kashirskoe shosse, 115409, Moscow, Russia

**Abstract.** A modification of the adaptive artificial viscosity (AAV) method is considered. This modification is based on one stage time approximation and is adopted to calculation of gasdynamics problems on unstructured grids with an arbitrary type of grid elements. The proposed numerical method has simplified logic, better performance and parallel efficiency compared to the implementation of the original AAV method. Computer experiments evidence the robustness and convergence of the method to difference solution.

## Introduction

The paper describes a one-step modification of the adaptive artificial viscosity (AAV) method [1] with generalization for the case of spatial discretization using unstructured grids with an arbitrary type of elements. The implementation of the one-step method has simplified logic, better performance and parallel efficiency compared to the implementation of the original AAV method. Based on the modified AAV method, a parallel algorithm and a program for simulating gas dynamics problems for multiprocessor systems with multi-core general-purpose processors are developed. The paper presents parallel software efficiency and examples of the possibility of using it for large-scale computational experiments.

## 1 Statement of the problem and approximation of equations

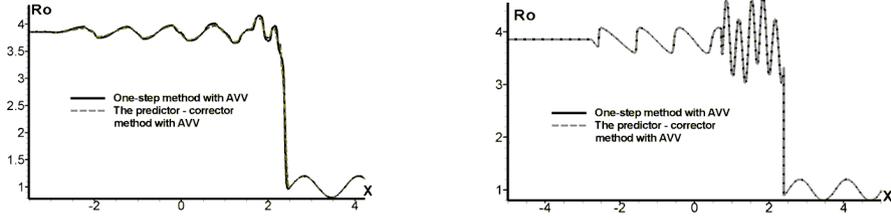
Let us consider a system of Euler equations [2] with time Lax – Vendroff corrections. The system in the difference-differential form is written as

$$\frac{\rho^{n+1} - \rho^n}{\tau} + \operatorname{div}(\rho \mathbf{v}) = \frac{\tau}{2} \left[ \operatorname{div} \left( \sum_{\alpha} \mathbf{e}_{\alpha} \operatorname{div} (I_{\alpha} \mathbf{v}) + \nabla p \right) + \operatorname{div}(\mu \nabla p) \right], \quad (1)$$

$$\begin{cases} \frac{I_{\alpha}^{n+1} - I_{\alpha}^n}{\tau} + \operatorname{div}(I_{\alpha} \mathbf{v}) + \nabla_{\alpha} p = \operatorname{div}(\mu \nabla_{\alpha} p) + \frac{\tau}{2} [\operatorname{div} \mathbf{A}_{\alpha} + \nabla_{\alpha} B], \\ \mathbf{A}_{\alpha} = \sum_{\beta} \mathbf{e}_{\beta} \left( \operatorname{div}(v_{\beta} I_{\alpha} \mathbf{v}) + \frac{I_{\alpha}}{\rho} \nabla_{\beta} p + v_{\beta} \nabla_{\alpha} p \right), \quad B = \operatorname{div}(\rho \mathbf{v}) + (\gamma - 1) p \operatorname{div} \mathbf{v}, \end{cases} \quad (2)$$

\*e-mail: piv2964@mail.ru

\*\*e-mail: ssoukov@gmail.com



**Figure 1.** Profiles of density. Left: discretization continuing 401 points. Right: ibid., 20001 points.

$$\begin{cases} \frac{E^{n+1} - E^n}{\tau} + \operatorname{div}((E + p)\mathbf{v}) = \operatorname{div}(\mu\nabla E) + \frac{\tau}{2} \operatorname{div} \mathbf{C}, \\ \mathbf{C} = \sum_{\alpha} \mathbf{e}_{\alpha} \left( \operatorname{div}(v_{\alpha}(E + p)\mathbf{v}) + \frac{(E + p)}{\rho} \nabla_{\alpha} p + v_{\alpha}((\gamma - 1)p \operatorname{div} \mathbf{v} + \operatorname{div}(p\mathbf{v})) \right). \end{cases} \quad (3)$$

Here  $\mathbf{e}_{\alpha}$ ,  $\mathbf{e}_{\beta}$  are basis vectors in the corresponding coordinate directions  $\alpha, \beta = 1, \dots, D$ ,  $D$  is the space dimension,  $\rho$  is the density,  $\mathbf{v}$  is the velocity vector,  $I$  is the momentum vector,  $E = \rho(0.5|v|^2 + \varepsilon)$  is the energy density,  $\rho\varepsilon$  is the internal energy density related to the pressure  $p$  by the equation of state  $p = (\gamma - 1)\rho\varepsilon$ ,  $\gamma$  is the degree index of the Poisson adiabat.

The Lax–Vendroff corrections ensure the second order of accuracy in time when constructing the difference scheme. The difference approximation of the remaining differential operators can be found in the monograph [1]. In the same paper an algorithm for finding the artificial viscosity is given. For a one-step modified AAV method, only the maximum artificial viscosity is used, both on shock waves and on oscillations of the numerical solution, which has the form:

$$\mu_i^n = \mu_{\max} = \max \frac{\Delta n_i}{2D} \sqrt{v_{n,i}^2 + c_i^2} \left( 1 - \frac{\tau^2 D^2}{\Delta n_i^2} (v_{n,i}^2 + c_i^2) \right), \quad (4)$$

where  $c_i$  is the sound speed,  $\Delta n_i$  is the projection of the vector between the centers of the cells on the normal vector  $\mathbf{n}_i$  to the face in question,  $v_{n,i}$  is the projection of the velocity vector defined at the center of the cell onto the normal vector  $\mathbf{n}_i$  edges.

The proposed numerical algorithm has three stages:

1. Determination of the regions of shock waves (SW), compression waves (CW), rarefaction waves (RW), and oscillations of a numerical solution (OSC) from the  $n$ -th time layer.
2. Setting the artificial viscosity maximum value inside SW and OSC regions and equating it to zero outside these two regions.
3. Computation of the fluxes together with all the unknown functions on the  $(n + 1)$ -th time layer.

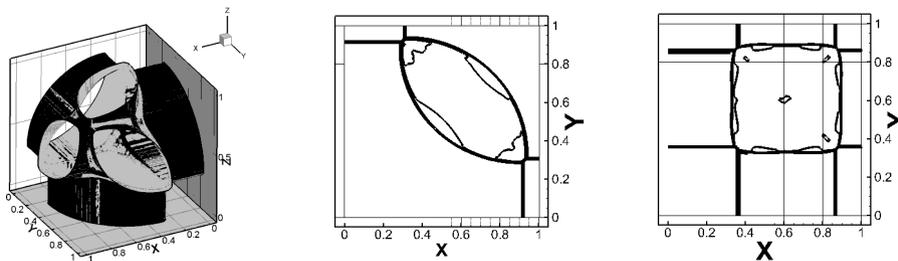
## 2 Numerical experiments

To check the proposed modified method, all the 1D and 2D tests from [3], [4] were solved within proposed approach. In this paper, we discuss the numerical calculation of the Shu–Osher problem [5] with initial data on the interval  $[0, 1]$ :  $\rho_L = 3.857143$ ,  $u_L = 2.629369$ ,  $p_L = 10.333333$ ,  $\rho_R = 1 + 0.2\sin(50x)$ ,  $u_R = 0$ ,  $p_R = 1$ . The initial point of discontinuity was located at the point  $x_0 = 0.1$ . The calculations was carried out up to the time  $T = 0.18$ .

Fig. 1 left shows numerical outputs for a number of 401 calculated points. In Fig. 1 right, the number of calculated points is 20001. With such a number of points, the lines obtained by the modified and original AAV method coincide, from which it can be concluded numerical outputs converge



**Figure 2.** Isolines of density. Left: overall view. Right: enlarged view showing discontinuity interactions.



**Figure 3.** Isolines of density in 3D. Left: 3D view, Center: inside plane  $z = 0$ , Right: inside plane  $z = 0.612$

to the same solution. The solid black line shows the output of the modified AAV method, and the solid gray line the output of the original AAV method. Calculations were carried out assuming the Courant number equal to 0.1.

Figs. 2 show the results of the calculation of the test problem No. 15, respectively, on the decays of the 4 discontinuities given by the initial data, which, after the disintegration of the discontinuity, form two contact discontinuities, a shock wave and a rarefaction wave. Fig. 2 right shows an enlarged image of the interaction of these four discontinuities. The solid black curve shows the results of the calculation using the modified scheme, and the gray dotted curve shows the outputs of the original predictor-corrector AAV method.

The Riemann problem [6] was taken as a three-dimensional test problem. A cube with a side equal to one was divided into eight equal cubes. Each of the cubes has its own velocity and pressure density parameters, see [6]. In this problem, the boundary conditions assumed vanishing normal derivatives of all quantities at cube surfaces. The task was solved up to  $T = 0.25$ . The comparison of the results produced by the modified and original AAV methods on uniform 3D grids  $200^3$ . Fig. 3 left shows the isosurfaces at time  $T = 0.25$  and density  $\rho = 1.5$ . Fig. 3 center shows isolines inside the plane  $z = 0$ , and Fig. 3 right – isolines inside the plane  $z = 0.612$ . The two results are visually indistinguishable.

### 3 Parallel program package

Based on the proposed modification of the AAV method, a parallel program for simulating gas dynamics problems for high-performance cluster architecture systems with multi-core general-purpose processors was developed. The synchronization and interaction of parallel processes occurs within the framework of a hybrid model of a concurrency control for the message transfer (MPI + OpenMP). The distribution of calculations between processes is performed by the decomposition of the calculation domain into subdomains (the method of geometric parallelism). The data exchange area includes

grid elements of two levels of connectivity over a dual graph. At the same time, duplication of calculations concerns only elements of the first level. Preprocessor data processing is implemented using the library functionality of the `hm4PreProcessorLib` [7] routines, designed for distributed processing of meshes with elements like tetrahedron, triangular prism, quadrangular pyramid and hexahedron, containing a total of up to  $5 \cdot 10^8$  elements. The tests of this package on hybrid large volume meshes confirmed high productivity and parallel efficiency of the parallel program.

## 4 Conclusion

The obtained numerical evidence points to the agreement of the outputs of the modified AAV method with those of the original AAV method. Under the refinement of the discretization mesh, the convergence to the same solutions have been found to occur. The proposed modified method of adaptive artificial viscosity on discontinuous solutions can compete with modern numerical methods.

## Acknowledgments

This work was supported by the Russian Foundation for Basic Researches (projects No. 15-01-04620-a, 15-07-04213-a, 16-07-00519-a, 16-29-15095-ofi-m).

## References

- [1] I.V. Popov, I.V. Fryazinov, *Method of Adaptive Artificial Viscosity for Numerical Solution of Gas Dynamics Equations* (KRASAND, Moscow, 2015) 288 pp.
- [2] L. Landau, E. Lifshitz, *Fluid Mechanics: Course of Theoretical Physics*, Vol. 6. (Pergamon Press, Oxford, 1987) 554 p.
- [3] I.V. Popov, I.V. Fryazinov, *Math. Models Comput. Simul.*, **1**(4), 493–502 (2009)
- [4] I.V. Popov, I.V. Fryazinov, *Math. Models Comput. Simul.*, **2**(6), 724–732 (2010)
- [5] C. Shu, S. Osher, *J. Comput. Phys.*, **83**(1), 32–78 (1989)
- [6] M. Kucharik, R. Liska, S. Steinberg, B. Wendroff, *Applied Numerical Mathematics*, **56**, 589–607 (2006)
- [7] S.A. Sukov, *Unstructured mesh preprocessing library hm4PreprocessorLib* (KIAM Preprint, No. 108, 2016)