Keldysh Institute • Publication search

Keldysh Institute preprints • Preprint No. 8, 2019

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Bragin M.D., Rogov B.V.

A conservative limiting method for bicompact schemes

Recommended form of bibliographic references: Bragin M.D., Rogov B.V. A conservative limiting method for bicompact schemes // Keldysh Institute Preprints. 2019. No. 8. 25 p. doi:10.20948/prepr-2019-8-e
URL: http://library.keldysh.ru/preprint.asp?id=2019-8&lg=e

KELDYSH INSTITUTE OF APPLIED MATHEMATICS

Russian Academy of Sciences

M. D. Bragin, B. V. Rogov

A conservative limiting method for bicompact schemes

Moscow — 2019

Michael Dmitrievich Bragin, Boris Vadimovich Rogov

A conservative limiting method for bicompact schemes

In this work, a new limiting method for bicompact schemes is proposed that preserves them conservative. The method is based upon a finite-element treatment of the bicompact approximation. An analogy between Galerkin schemes and bicompact schemes is established. The proposed method is tested on one-dimensional gasdynamics problems that include the Sedov problem, the Riemann "peak" problem, and the Shu-Osher problem. It is shown on these examples that bicompact schemes with conservative limiting are significantly more accurate than hybrid bicompact schemes.

Keywords: bicompact schemes, conservative schemes, monotonicity preserving schemes, hyperbolic equations, discontinuous solutions.

Брагин М. Д., Рогов Б. В.

Консервативная монотонизация бикомпактных схем

В работе предлагается новый метод монотонизации высокоточных бикомпактных схем, не нарушающий их консервативности. Этот метод основан на конечно-элементном представлении бикомпактной аппроксимации. Установлена аналогия между схемами Галеркина и бикомпактными схемами. Разработанный метод проверен на одномерных задачах газодинамики: задаче Седова, задаче Римана с узким пиком плотности, задаче Шу-Ошера. На их примере показано, что бикомпактные схемы с консервативной монотонизацией значительно точнее гибридных бикомпактных схем.

Ключевые слова: бикомпактные схемы, консервативные схемы, сохраняющие монотонность схемы, гиперболические уравнения, разрывные решения.

This research was supported by the Russian Foundation for Basic Research, project no. 18-31-00045.

Contents

Introduction 3
1. Finite-element representation of the bicompact approximation 4
2. Conservative limiting method for bicompact schemes 10
3. Analogy between Galerkin and bicompact schemes. 14
4. Testing of the method on one-dimensional gas dynamics problems. . 16
Conclusions 22
Bibliography list 23

Introduction

An important area in modern computational mathematics is the development of high-order accurate schemes for the numerical solution of nonstationary hyperbolic equations. This class of schemes includes bicompact ones, which combine a number of positive properties. Namely, they have an even (fourth, sixth, and so on) order of accuracy in space on a stencil occupying one grid cell; it is possible to choose an approximation in time; these schemes are efficient, though implicit; and they have good spectral properties [1].

It is well known that hyperbolic equations admit discontinuous solutions. For correct shock-capturing computations of such solutions, the scheme has to be monotone (in the sense of one of the available definitions of this property). However, according to Godunov's theorem [2], a scheme having a linear approximation of derivatives with a minimum order higher than the first cannot be monotone. To overcome this barrier, nonlinear approximations have been created, namely, flux and slope limiters [3–5], numerical filters [6–8], artificial dissipation [9–12], ENO/WENO approaches [13–16], and others.

In [17–21] the monotonicity of high-order accurate bicompact schemes was ensured by applying an original hybrid scheme method that develops the ideas of the classical Fedorenko method [22]. The approach of [17–21] is as follows: in each grid node at an upper time level, the resulting solution is set equal to a nonlinear convex combination of two solutions, one of which is computed using a monotone scheme A, while the other, a nonmonotone high-order accurate scheme B. Schemes A and B use the same initial condition at a lower time level. The weight α of this convex combination depends on the difference between the solutions of schemes A and B at the given grid node.

The hybrid scheme method [17–21] has several advantages. First, it provides the maximum possible degree of locality: to combine solutions, it is sufficient to use data from a single node or cell. Second, the method is versatile: schemes A and B can be arbitrary. Third, it is activated only in zones of transition from areas of steep gradients to areas of smoothness, i. e. hybridization is in these zones where nonmonotonicities are generated. Nevertheless, hybrid schemes [17–21] have a serious disadvantage: they are not conservative. There is no guarantee that the solution obtained by directly weighting the solutions of two even conservative schemes will obey some conservation law.

Our goal in this work is to eliminate this disadvantage in the case when scheme B is bicompact. The solution consists of the following ingredients:

1. The direct weighting of solutions is completely avoided.

2. The weighting factor α in hybrid schemes [17–21] is used as a good indicator of zones generating nonmonotonicity.
3. The numerical solution in a cell is represented as a finite element of the form "integral average + correction terms".
4. The correction terms are limited using α. The continuity of the finite-element approximation on the cell boundaries is maintained without violating conservation laws.

This preprint is organized as follows. A finite-element representation of the bicompact approximation is obtained in Section 1. A conservative limiting method is described in Section 2. An analogy between Galerkin and bicompact schemes is established in Section 3. The new method is tested in Section 4.

1. Finite-element representation of the bicompact approximation

One-dimensional case. Consider a system of one-dimensional homogeneous quasilinear hyperbolic equations:

$$\mathscr{L}_1(\boldsymbol{Q}) \equiv \partial_t \boldsymbol{Q} + \partial_x \boldsymbol{F}(\boldsymbol{Q}) = \boldsymbol{0}, \quad x \in (0, x_{\max}), \quad t \in (0, t_{\max}), \tag{1}$$

where $\boldsymbol{Q} = (Q_1, \ldots, Q_m) = \boldsymbol{Q}(x,t)$ is the sought vector of conservative variables, $\boldsymbol{F}(\boldsymbol{Q})$ is the vector of physical fluxes in the Ox direction, and $\partial_x \equiv \partial/\partial x$. It is assumed that the Jacobian matrix $\boldsymbol{A}(\boldsymbol{Q}) = \partial_{\boldsymbol{Q}} \boldsymbol{F}(\boldsymbol{Q}) > 0$ is positive definite for any $\boldsymbol{Q}$ allowed by system (1). This assumption is not essential for bicompact schemes, but provides a better understanding of their idea. In the case of an indefinite matrix $\boldsymbol{A}$ the global Lax–Friedrichs flux splitting method is used (see, for example, [19]). System (1) is also assumed to be supplemented with an initial condition at $t = 0$ and a boundary condition at $x = 0$ under which it has a unique solution in $[0, x_{\max}] \times [0, t_{\max}]$.

For system (1) a semi-discrete bicompact scheme of fourth-order accuracy in x [23] is written as

$$\begin{cases} \dfrac{d}{dt}(A_0^x \boldsymbol{Q}_{j+1/2}) + \Lambda_1^x \boldsymbol{F}_{j+1/2} = \boldsymbol{0}, \\ \dfrac{d}{dt}(\Lambda_1^x \boldsymbol{Q}_{j+1/2}) + \Lambda_2^x \boldsymbol{F}_{j+1/2} = \boldsymbol{0}, \end{cases} \quad j = \overline{0, N_x - 1}. \tag{2}$$

The following notation is used in (2). On the interval $[0, x_{\max}]$ we introduce a (generally nonuniform) grid

$$\Omega = \Omega_x = \{x_0, x_{1/2}, x_1, x_{3/2}, x_2, \ldots, x_{N_x}\}, \quad x_0 = 0, \quad x_{N_x} = x_{\max},$$
$$h_{x,\,j+1/2} = x_{j+1} - x_j \text{ is the step size in } x, \quad x_{j+1/2} = \frac{x_j + x_{j+1}}{2}.$$

In what follows, the index $(j+1/2)$ on the step in x is omitted for computations performed within a single cell. The vectors $\boldsymbol{Q}_j$ and $\boldsymbol{Q}_{j+1/2}$ approximate the exact solution at the nodes x_j and $x_{j+1/2}$ respectively; $\boldsymbol{F}_j \equiv \boldsymbol{F}(\boldsymbol{Q}_j)$, and $\boldsymbol{F}_{j+1/2} \equiv \boldsymbol{F}(\boldsymbol{Q}_{j+1/2})$. For an arbitrary grid function U, the difference operators $A_0^x, \Lambda_1^x, \Lambda_2^x$ are defined as

$$A_0^x U_{j+1/2} = \frac{U_j + 4U_{j+1/2} + U_{j+1}}{6}, \quad \Lambda_1^x U_{j+1/2} = \frac{U_{j+1} - U_j}{h_x},$$
$$\Lambda_2^x U_{j+1/2} = \frac{4(U_j - 2U_{j+1/2} + U_{j+1})}{h_x^2}.$$

The spatial stencil of scheme (2) is $S = \{x_j,\, x_{j+1/2},\, x_{j+1}\}$, and it is entirely contained in the single cell $K_{j+1} = [x_j, x_{j+1}]$ of Ω. The ODE system (2) is integrated with respect to t by applying A- and L-stable DIRK methods of high order (see, e. g., [24]). Below, n denotes the time level index, τ is the time step (possibly variable), and $\boldsymbol{Q}^n$ is the numerical solution at the level t^n.

The idea of bicompact schemes is as follows. For an order of accuracy in x within a single cell to be higher than the second, the cell has to contain at least three nodes. In computing the solution at the next time level t^{n+1}, the cell K_{j+1} contains two unknown vectors, namely, $\boldsymbol{Q}_{j+1/2}^{n+1}$ and $\boldsymbol{Q}_{j+1}^{n+1}$. Since $\boldsymbol{A} > 0$, the vector $\boldsymbol{Q}_j^{n+1}$ has been found after computing cells with lower indices. Clearly, two equations are necessary to find two unknowns. These equations are derived by discretizing the system $\mathscr{L}_1(\boldsymbol{Q}) = \boldsymbol{0}$ and its differential consequence $\partial_x \mathscr{L}_1(\boldsymbol{Q}) = \boldsymbol{0}$ with respect to x in the cell K_{j+1}. The discretization is based on the method of lines and the finite-volume method (see [23]). As a result, we obtain system (2) consisting of two equations.

Curiously enough, the bicompact spatial approximation in scheme (2) is somehow related to the finite-element method.

Consider a test function $u(x)$ defined on $[0, x_{\max}]$. Suppose that its three values u_j, $u_{j+1/2}$, u_{j+1} at the nodes x_j, $x_{j+1/2}$, x_{j+1}, respectively, are given in any cell K_{j+1}. These data are used to construct a quadratic interpolation polynomial $u_h(x; K_{j+1})$ defined on K_{j+1}:

$$u_h(x; K_{j+1}) = u_{j+1/2} + \xi(\Delta_0^x u_{j+1/2}) + 2\xi^2(\Delta_2^x u_{j+1/2}), \quad x \in K_{j+1}, \qquad (3)$$

where

$$\xi = \frac{x - x_{j+1/2}}{h_x} \in [-1/2, +1/2], \quad \Delta_0^x = h_x \Lambda_1^x, \quad \Delta_2^x = \frac{1}{4} h_x^2 \Lambda_2^x.$$

By definition, polynomial (3) satisfies the equalities

$$u_h(x_i; K_{j+1}) = u_i, \quad i \in \{j,\, j+1/2,\, j+1\}.$$

The set of polynomials (3) forms a continuous piecewise polynomial function, or a spline of smoothness 0 on the interval $[0, x_{\max}]$.

Consider a system of polynomial functions $\{\varphi_l(x)\}_{l=0}^{\infty}$ that is complete in the space $\mathbb{L}_2(K_{j+1})$. Let this system be orthonormal with respect to the scalar product

$$(f, g) = \frac{1}{h_x} \int_{x_j}^{x_{j+1}} f(x) g(x)\, dx \quad \forall f, g \in \mathbb{L}_2(K_{j+1}). \tag{4}$$

Explicit expressions for the first three functions of this system are

$$\varphi_l(x) = p_l(\xi), \quad l = 0, 1, 2,$$
$$p_0(\xi) = 1, \quad p_1(\xi) = 2\sqrt{3}\xi, \quad p_2(\xi) = \frac{\sqrt{5}}{2}(12\xi^2 - 1).$$

In the subsequent sections, we will need the following expansion of polynomial (3) in terms of the functions $\varphi_0(x)$, $\varphi_1(x)$, and $\varphi_2(x)$:

$$u_h(x; K_{j+1}) = \sum_{l=0}^{2} c_l \varphi_l(x), \tag{5}$$

where the expansion coefficients are given by

$$c_0 = A_0^x u_{j+1/2}, \quad c_1 = \frac{\Delta_0^x u_{j+1/2}}{2\sqrt{3}}, \quad c_2 = \frac{\Delta_2^x u_{j+1/2}}{3\sqrt{5}}. \tag{6}$$

Let us derive scheme (2) by an alternative method. For each $t = \text{const}$, the exact solution $\boldsymbol{Q}(x, t)$ of system (1) and the function $\mathscr{F}(x, t) = \boldsymbol{F}[\boldsymbol{Q}(x, t)]$ on the interval $[0, x_{\max}]$ are approximated by splines whose fragments in cells are determined by formulas of type (3) (with u replaced by $\boldsymbol{Q}$ or $\mathscr{F}$). The constructed splines are continuous finite-element approximations of the functions $\boldsymbol{Q}(x, t)$ and $\mathscr{F}(x, t)$ on the interval $[0, x_{\max}]$. To derive equations for approximate grid values of $\boldsymbol{Q}(x, t)$, we substitute these splines into the left-hand sides of the equations $\mathscr{L}_1(\boldsymbol{Q}) = \boldsymbol{0}$ and $\partial_x \mathscr{L}_1(\boldsymbol{Q}) = \boldsymbol{0}$, average the left-hand sides over each cell K_{j+1}, and set the resulting expressions to zero:

$$\left.\begin{aligned} \boldsymbol{0} &= \frac{1}{h_x} \int_{x_j}^{x_{j+1}} (\partial_t \boldsymbol{Q}_h + \partial_x \mathscr{F}_h)\, dx = \frac{d}{dt}(A_0^x \boldsymbol{Q}_{j+1/2}) + \Lambda_1^x \boldsymbol{F}_{j+1/2}, \\ \boldsymbol{0} &= \frac{1}{h_x} \int_{x_j}^{x_{j+1}} (\partial_t \partial_x \boldsymbol{Q}_h + \partial_x^2 \mathscr{F}_h)\, dx = \frac{d}{dt}(\Lambda_1^x \boldsymbol{Q}_{j+1/2}) + \Lambda_2^x \boldsymbol{F}_{j+1/2}. \end{aligned}\right\}$$

Obviously, the result coincides with scheme (2).

Two-dimensional case. Now consider the two-dimensional version of system (1) in the simplest computational domain D:

$$\begin{gathered}\mathscr{L}_2(\boldsymbol{Q}) \equiv \partial_t \boldsymbol{Q} + \partial_x \boldsymbol{F}(\boldsymbol{Q}) + \partial_y \boldsymbol{G}(\boldsymbol{Q}) = \boldsymbol{0}, \\ (x, y) \in D = (0, x_{\max}) \times (0, y_{\max}), \quad t \in (0, t_{\max}),\end{gathered} \tag{7}$$

where $\boldsymbol{G}(\boldsymbol{Q})$ is the vector of physical fluxes in the Oy direction. By analogy with the one-dimensional case, we assume that the Jacobian matrices $\boldsymbol{A}(\boldsymbol{Q}) > 0$ and $\boldsymbol{B}(\boldsymbol{Q}) = \partial_{\boldsymbol{Q}} \boldsymbol{G}(\boldsymbol{Q}) > 0$ for any $\boldsymbol{Q}$ allowed by (7). System (7) is supplemented with an initial condition at $t = 0$ and boundary conditions at $x = 0$ and $y = 0$ under which a unique solution is assumed to exist in $\overline{D} \times [0, t_{\max}]$, where $\overline{D} = D \cup \partial D$ and ∂D is the boundary of the domain D.

The semi-discrete bicompact scheme of fourth-order accuracy in x, y for system (7) has the form (see [23])

$$\begin{cases} \dfrac{d}{dt}(A_0^y A_0^x \boldsymbol{Q}_C) + A_0^y \Lambda_1^x \boldsymbol{F}_C + \Lambda_1^y A_0^x \boldsymbol{G}_C = \boldsymbol{0}, \\ \dfrac{d}{dt}(A_0^y \Lambda_1^x \boldsymbol{Q}_C) + A_0^y \Lambda_2^x \boldsymbol{F}_C + \Lambda_1^y \Lambda_1^x \boldsymbol{G}_C = \boldsymbol{0}, \\ \dfrac{d}{dt}(\Lambda_1^y A_0^x \boldsymbol{Q}_C) + \Lambda_1^y \Lambda_1^x \boldsymbol{F}_C + \Lambda_2^y A_0^x \boldsymbol{G}_C = \boldsymbol{0}, \\ \dfrac{d}{dt}(\Lambda_1^y \Lambda_1^x \boldsymbol{Q}_C) + \Lambda_1^y \Lambda_2^x \boldsymbol{F}_C + \Lambda_2^y \Lambda_1^x \boldsymbol{G}_C = \boldsymbol{0}, \end{cases} \tag{8}$$

where $C = (j + 1/2,\ k + 1/2)$ is a multi-index, $j = \overline{0, N_x - 1}$, and $k = \overline{0, N_y - 1}$. Similar notation is used for schemes (2) and (8): on the interval $[0, y_{\max}]$, we introduce a grid (possibly nonuniform)

$$\begin{gathered}\Omega_y = \{y_0, y_{1/2}, y_1, y_{3/2}, y_2, \ldots, y_{N_y}\}, \quad y_0 = 0, \quad y_{N_y} = y_{\max}, \\ h_{y,\,k+1/2} = y_{k+1} - y_k \text{ is the step size in } y, \quad y_{k+1/2} = \frac{y_k + y_{k+1}}{2}.\end{gathered}$$

The grid in $\overline{D}$ is $\Omega = \Omega_x \times \Omega_y$. Its cells are the rectangles

$$K_i = [x_j, x_{j+1}] \times [y_k, y_{k+1}], \quad i = jN_y + k + 1.$$

The difference operators A_0^y, Λ_1^y, and Λ_2^y are defined in a similar manner to the operators A_0^x, Λ_1^x, Λ_2^x. The operators indexed by "x" and "y" act only on the first and second indices of a grid function, respectively. It is easy to show that the operators acting in the Ox direction commute with those acting in the Oy

direction. The spatial stencil S of scheme (8) consists of nine points:

$$S = \{(x_j, y_k),\ (x_j, y_{k+1/2}),\ (x_j, y_{k+1}),\ (x_{j+1/2}, y_k),\ (x_{j+1/2}, y_{k+1/2}),\\ (x_{j+1/2}, y_{k+1}),\ (x_{j+1}, y_k),\ (x_{j+1}, y_{k+1/2}),\ (x_{j+1}, y_{k+1})\}.$$

Note that, as in the one-dimensional case, the stencil lies entirely in the single cell K_i.

Let us generalize polynomial (3) to the two-dimensional case. Suppose that $u(x, y)$ is a test function defined on $\overline{D}$. Clearly, the desired interpolating polynomial $u_h(x, y; K_i)$ can be neither quadratic nor cubic in x, y: it is defined by six coefficients in the former case and by ten coefficients in the latter case, while there are nine values of u at nine nodes in the cell K_i. An attempt to express the coefficients u_h in terms of these values would lead to an overdetermined or underdetermined SLAE.

Note that any two-dimensional difference operator in scheme (8) represents the product of two one-dimensional operators, one acting along the Oy axis and the other, along the Ox axis. Moreover, polynomial (3) can be written as the result produced by a one-parameter difference operator acting on $u_{j+1/2}$:

$$u_h(x; K_{j+1}) = \mathscr{P}^x(\xi) u_{j+1/2}, \quad \mathscr{P}^x(\xi) = 1 + \xi\Delta_0^x + 2\xi^2\Delta_2^x.$$

In a similar manner, the one-parameter operator $\mathscr{P}^y(\eta)$ is defined as

$$\mathscr{P}^y(\eta) = 1 + \eta\Delta_0^y + 2\eta^2\Delta_2^y,$$

where

$$\eta = \frac{y - y_{k+1/2}}{h_y} \in [-1/2, +1/2], \quad \Delta_0^y = h_y\Lambda_1^y, \quad \Delta_2^y = \frac{1}{4}h_y^2\Lambda_2^y.$$

Let us try $\mathscr{P}^y(\eta)\mathscr{P}^x(\xi)u_C$ as the desired two-dimensional polynomial. Expanding all brackets, we obtain

$$u_h(x, y; K_i) = u_C + \xi(\Delta_0^x u_C) + \eta(\Delta_0^y u_C) + 2\xi^2(\Delta_2^x u_C) + \xi\eta(\Delta_0^y\Delta_0^x u_C) + \\ + 2\eta^2(\Delta_2^y u_C) + 2\xi^2\eta(\Delta_0^y\Delta_2^x u_C) + 2\xi\eta^2(\Delta_2^y\Delta_0^x u_C) + 4\xi^2\eta^2(\Delta_2^y\Delta_2^x u_C). \quad (9)$$

Note that (9) is a well-known biquadratic interpolation of the function $u(x, y)$ in the cell K_i. The set of polynomials (9) forms a two-dimensional spline of smoothness 0 in $\overline{D}$.

Following the line of reasoning used in the one-dimensional case, we check whether scheme (8) can be derived using polynomial (9). The exact solution $\boldsymbol{Q}(x, y, t)$ of system (7) and the functions $\mathscr{F}(x, y, t) = \boldsymbol{F}[\boldsymbol{Q}(x, y, t)]$,

$\mathscr{G}(x,y,t) = \boldsymbol{G}[\boldsymbol{Q}(x,y,t)]$ in $\overline{D}$ are approximated by splines whose fragments are given by formulas of type (9) (with u replaced by $\boldsymbol{Q}$, $\mathscr{F}$, or $\mathscr{G}$). These splines are continuous finite-element approximations of $\boldsymbol{Q}(x,y,t)$, $\mathscr{F}(x,y,t)$, and $\mathscr{G}(x,y,t)$ in $\overline{D}$. Substituting them into the left-hand sides of the equations

$$\mathscr{L}_2(\boldsymbol{Q}) = \boldsymbol{0}, \quad \partial_x\mathscr{L}_2(\boldsymbol{Q}) = \boldsymbol{0}, \quad \partial_y\mathscr{L}_2(\boldsymbol{Q}) = \boldsymbol{0}, \quad \partial_x\partial_y\mathscr{L}_2(\boldsymbol{Q}) = \boldsymbol{0},$$

averaging these left-hand sides over the cell K_i, and setting the resulting expressions to zero, we obtain

$$\left.\begin{aligned}
\boldsymbol{0} &= \frac{1}{h_x h_y}\int\limits_{K_i}(\partial_t\boldsymbol{Q}_h + \partial_x\mathscr{F}_h + \partial_y\mathscr{G}_h)\,dxdy = \\
&= \frac{d}{dt}(A_0^y A_0^x\boldsymbol{Q}_C) + A_0^y\Lambda_1^x\boldsymbol{F}_C + \Lambda_1^y A_0^x\boldsymbol{G}_C, \\
\boldsymbol{0} &= \frac{1}{h_x h_y}\int\limits_{K_i}(\partial_t\partial_x\boldsymbol{Q}_h + \partial_x^2\mathscr{F}_h + \partial_x\partial_y\mathscr{G}_h)\,dxdy = \\
&= \frac{d}{dt}(A_0^y\Lambda_1^x\boldsymbol{Q}_C) + A_0^y\Lambda_2^x\boldsymbol{F}_C + \Lambda_1^y\Lambda_1^x\boldsymbol{G}_C, \\
\boldsymbol{0} &= \frac{1}{h_x h_y}\int\limits_{K_i}(\partial_t\partial_y\boldsymbol{Q}_h + \partial_x\partial_y\mathscr{F}_h + \partial_y^2\mathscr{G}_h)\,dxdy = \\
&= \frac{d}{dt}(\Lambda_1^y A_0^x\boldsymbol{Q}_C) + \Lambda_1^y\Lambda_1^x\boldsymbol{F}_C + \Lambda_2^y A_0^x\boldsymbol{G}_C, \\
\boldsymbol{0} &= \frac{1}{h_x h_y}\int\limits_{K_i}(\partial_t\partial_x\partial_y\boldsymbol{Q}_h + \partial_x^2\partial_y\mathscr{F}_h + \partial_x\partial_y^2\mathscr{G}_h)\,dxdy = \\
&= \frac{d}{dt}(\Lambda_1^y\Lambda_1^x\boldsymbol{Q}_C) + \Lambda_1^y\Lambda_2^x\boldsymbol{F}_C + \Lambda_2^y\Lambda_1^x\boldsymbol{G}_C.
\end{aligned}\right\}$$

Therefore, scheme (8) can be derived from approximation (9) in the spirit of the finite-element method (and the finite-volume method).

In Section 2, we will need polynomial (9) expressed in terms of the system of polynomial functions $\{\psi_l(x,y)\}_{l=0}^{\infty}$, which is complete in $\mathbb{L}_2(K_i)$ and orthonormal with respect to the scalar product

$$(f,g) = \frac{1}{h_x h_y}\int\limits_{K_i} f(x,y)g(x,y)\,dxdy \quad \forall f,g \in \mathbb{L}_2(K_i).$$

The functions ψ_l can easily be expressed in terms of φ_l and p_l:

$$\psi_l(x,y) = \varphi_r(x)\varphi_s(y) = p_r(\xi)p_s(\eta), \quad r = r(l), \quad s = s(l), \quad \forall l \geqslant 0. \qquad (10)$$

The required first nine functions of the system are

$$\psi_0(x,y)=1,\quad \psi_1(x,y)=2\sqrt{3}\xi,\quad \psi_2(x,y)=2\sqrt{3}\eta,$$
$$\psi_3(x,y)=\frac{\sqrt{5}}{2}(12\xi^2-1),\quad \psi_4(x,y)=12\xi\eta,\quad \psi_5(x,y)=\frac{\sqrt{5}}{2}(12\eta^2-1),$$
$$\psi_6(x,y)=\sqrt{15}(12\xi^2-1)\eta,\quad \psi_7(x,y)=\sqrt{15}\xi(12\eta^2-1),$$
$$\psi_8(x,y)=\frac{5}{4}(12\xi^2-1)(12\eta^2-1).$$

The expression of polynomial (9) in terms of this system has the form

$$u_h(x,y;K_i)=\sum_{l=0}^{8}c_l\psi_l(x,y), \tag{11}$$

where the expansion coefficients are given by

$$\begin{aligned} c_0&=A_0^yA_0^xu_C,\quad & c_1&=\frac{A_0^y\Delta_0^xu_C}{2\sqrt{3}},\quad & c_2&=\frac{\Delta_0^yA_0^xu_C}{2\sqrt{3}},\\ c_3&=\frac{A_0^y\Delta_2^xu_C}{3\sqrt{5}},\quad & c_4&=\frac{\Delta_0^y\Delta_0^xu_C}{12},\quad & c_5&=\frac{\Delta_2^yA_0^xu_C}{3\sqrt{5}},\\ c_6&=\frac{\Delta_0^y\Delta_2^xu_C}{6\sqrt{15}},\quad & c_7&=\frac{\Delta_2^y\Delta_0^xu_C}{6\sqrt{15}},\quad & c_8&=\frac{\Delta_2^y\Delta_2^xu_C}{180}. \end{aligned} \tag{12}$$

Three-dimensional case. The transition from the two- to three-dimensional case is similar to the transition from the one- to two-dimensional case, so we will not consider the three-dimensional case in detail. Expressions for the three-dimensional basis functions $\psi_l(x,y,z)$ and the coefficients c_l ($l=\overline{0,26}$) can be easily written by analogy with formulas (10) and (12) and the difference operators of the three-dimensional semi-discrete bicompact scheme (see [23]).

Conclusion. Bicompact schemes are based on the continuous finite-element approximations (5), (11). Relying on these approximations, a conservative limiting method is constructed for bicompact schemes in Section 2. In Section 3, the analogy between bicompact and finite-element schemes is deepened and we discuss in what sense the approximation of differential consequences of the original system of equations can be understood.

2. Conservative limiting method for bicompact schemes

Let us describe a limiting method for bicompact schemes that does not violate their conservativeness. Like in the case of a hybrid scheme [21], we consider two schemes: a monotone scheme A and a high-order accurate

bicompact scheme B. Note that the numerical conservation laws for schemes A and B do not need to be identical.

Suppose that the numerical solution $\boldsymbol{Q}^n$ at the level t^n is known. By using $\boldsymbol{Q}^n$, we compute two solutions at the next time level t^{n+1}, namely, $\boldsymbol{Q}_A^{n+1}$ is computed by applying scheme A, and $\boldsymbol{Q}_B^{n+1}$, by applying scheme B. In what follows, all operations are executed at the level t^{n+1}, so the superscript $n+1$ is omitted for brevity. In each cell K_i, the coefficients $\boldsymbol{c}_l$ of the finite-element approximation (5) or (11) for the solution $\boldsymbol{Q}_B$ are determined using formulas (6) or (12) (with u replaced by $\boldsymbol{Q}_B$). In addition to $\boldsymbol{c}_l$, the weighting factors α_s of the hybrid scheme are computed in the cell K_i [21]:

$$\alpha_s = f(w_s), \quad w_s = \frac{C_1 |Q_{As}(\boldsymbol{r}_i) - Q_{Bs}(\boldsymbol{r}_i)|}{\max_{K_i} Q_{As} - \min_{K_i} Q_{As}}, \quad f(w) = \frac{w^2}{1+w^2}, \quad s = \overline{1,m}, \tag{13}$$

where Q_{As} and Q_{Bs} are sth components of the solutions $\boldsymbol{Q}_A$ and $\boldsymbol{Q}_B$, $\boldsymbol{r}_i \in \Omega$ is the center of K_i. In general, $\boldsymbol{r}$ denotes the radius vector of an arbitrary point in space. The number $C_1 \geqslant 0$ is a tuned parameter of the method. An implementation of (13) in a program code requires an additional small term of the order of machine precision in the denominator of the fraction in the formula for w_s in order to prevent division by zero when $Q_{As} = \text{const}$ on K_i.

Note that, in contrast to [21], the amplitude of variations in Q_{As} in the denominator of the expression for w_s in (13) is computed locally over the cell K_i, rather than globally over the entire computational domain.

Unlike [21], the factors α_s are used for correcting the higher-order coefficients of the finite-element approximation in scheme B in a cell, rather than for weighting the solutions $\boldsymbol{Q}_A$ and $\boldsymbol{Q}_B$ directly. Specifically, the components c_{ls} of the vectors $\boldsymbol{c}_l$ for $l \geqslant 1$ are replaced by the quantities

$$\widetilde{c}_{ls} = (1 - \alpha_s) c_{ls}, \quad l \geqslant 1. \tag{14}$$

The component c_{0s} remains unchanged. In other words, the correction to the integral average c_{0s} of the approximation $Q_{Bs,h}(\boldsymbol{r}; K_i)$ is multiplied by the correction factor $(1 - \alpha_s)$:

$$Q_{Bs,h}(\boldsymbol{r}; K_i) \to \widetilde{Q}_{Bs,h}(\boldsymbol{r}; K_i) = c_{0s} + (1 - \alpha_s) \left[Q_{Bs,h}(\boldsymbol{r}; K_i) - c_{0s} \right].$$

Obviously, the replacement of $\boldsymbol{c}_l$ by $\widetilde{\boldsymbol{c}}_l$ for $l \geqslant 1$ leaves the integral averages in all cells unchanged; therefore, the total integral of the numerical solution over D remains unchanged as well. However, since the coefficients $\boldsymbol{c}_l$ in each cell are corrected irrespective of those in neighboring cells, the continuity of the approximation in $\overline{D}$ is violated. At each node of Ω lying on the boundary

between cells, there are several values of the numerical solution, which is not acceptable for schemes A and B.

Accordingly, our goal is to merge the approximations in cells continuously, so that the total integral of the numerical solution over D remains unchanged.

The integral under discussion is

$$\boldsymbol{I} = \sum_i \int\limits_{K_i} \boldsymbol{Q}_{B,h}(\boldsymbol{r}; K_i)\, dV = \sum_i \int\limits_{K_i} \widetilde{\boldsymbol{Q}}_{B,h}(\boldsymbol{r}; K_i)\, dV = \sum_i \boldsymbol{c}_0(K_i)\Delta V_i, \qquad (15)$$

where ΔV_i is the volume of the cell K_i. The expression for $\boldsymbol{I}$ in (15) can be rewritten as

$$\boldsymbol{I} = \sum_{\boldsymbol{r}\in\Omega} \sum_{i|\boldsymbol{r}\in S(K_i)} \omega_i(\boldsymbol{r})\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})\Delta V_i. \qquad (16)$$

Let us explain formula (16). The outer sum extends over all nodes of the grid Ω, while the inner sum extends over those cells in which the spatial stencil contains a node $\boldsymbol{r} \in \Omega$; $\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})$ is the limited value of the solution at the node $\boldsymbol{r}$ on the side of the cell K_i computed using the coefficients $\boldsymbol{c}_0, \widetilde{\boldsymbol{c}}_l$, $l \geqslant 1$; and $\omega_i(\boldsymbol{r})$ is the weight of $\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})$ in the quadrature formula for the cell K_i. The resulting value $\boldsymbol{Q}(\boldsymbol{r}) = \boldsymbol{Q}^{n+1}(\boldsymbol{r})$ is made up of $\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})$:

$$\boldsymbol{Q}(\boldsymbol{r}) = \frac{\sum_{i|\boldsymbol{r}\in S(K_i)} \omega_i(\boldsymbol{r})\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})\Delta V_i}{\sum_{i|\boldsymbol{r}\in S(K_i)} \omega_i(\boldsymbol{r})\Delta V_i}. \qquad (17)$$

In view of (17), formula (16) becomes

$$\boldsymbol{I} = \sum_{\boldsymbol{r}\in\Omega} \boldsymbol{Q}(\boldsymbol{r}) \sum_{i|\boldsymbol{r}\in S(K_i)} \omega_i(\boldsymbol{r})\Delta V_i. \qquad (18)$$

The form of integral (18) exactly corresponds to the case of a continuous approximation. The summation in (18) is carried out over all nodes of Ω. Each term of this sum is the value of the solution at a node multiplied by the sum of products of quadrature weights and volumes in the cells containing this node. Note that, after applying the merging procedure, generally speaking, the integral average in each cell changes, but the total integral remains unchanged by construction.

After completing the merging procedure, the resulting solution at the level t^{n+1} has been constructed. Summarizing what was said above, the proposed method can be formulated as a sequence of steps:

1. Compute the solutions $\boldsymbol{Q}_A^{n+1}$ and $\boldsymbol{Q}_B^{n+1}$ by applying schemes A and B, respectively, with $\boldsymbol{Q}^n$ being their common initial condition.

2. In each cell K_i of the grid Ω, find the coefficients $\boldsymbol{c}_l$ of the finite-element approximation for the solution $\boldsymbol{Q}_B^{n+1}$. Compute the weighting factors α_s using formula (13).
3. Replace $\boldsymbol{c}_l$ by $\widetilde{\boldsymbol{c}}_l$ for $l \geqslant 1$ in all grid cells by using formula (14).
4. Compute the resulting solution $\boldsymbol{Q}^{n+1}$ in each node of Ω by applying formula (17).

Remarks:

1. In domains where the solution is smooth, we have $\alpha_s = O(\tau^2)$ as $\tau \to 0$. In such cells, the finite-element approximations $\boldsymbol{Q}_{B,h}(\boldsymbol{r}; K_i)$ are nearly not corrected, the deviations from continuity on the cell boundaries are small, and merging (17) does almost nothing, since all the values $\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})$ are close to each other.
2. Sums of the form $\sum_{i|\boldsymbol{r}\in S(K_i)}$ consist of only one term at (a) cell centers; (b) centers of cell faces lying on faces of $\overline{D}$; (c) centers of edges of cells lying on edges of $\overline{D}$; and (d) vertices of $\overline{D}$.
3. Expression (17) is a linear combination of the vectors $\widetilde{\boldsymbol{L}}_i(\boldsymbol{r})$ with positive coefficients.
4. In regions of steep changes of gradients, where nonmonotonicity is generated and the solutions produced by schemes A and B differ significantly, we have $\alpha_s \to 1$. In this limit, the finite-element approximations $\boldsymbol{Q}_{B,h}(\boldsymbol{r}; K_i)$ yield constants (leading terms), from which the solution on the cell boundaries is reconstructed monotonically and linearly by applying formula (17).

Example. Let us analyze formula (17) in the one-dimensional case. At half-integer nodes and at integer nodes at the endpoints of the interval $[0, x_{\max}]$, we have

$$\boldsymbol{Q}_{j+1/2} = \widetilde{\boldsymbol{L}}_{j+1}(x_{j+1/2}), \quad \boldsymbol{Q}_0 = \widetilde{\boldsymbol{L}}_1(0), \quad \boldsymbol{Q}_{N_x} = \widetilde{\boldsymbol{L}}_{N_x}(x_{\max}).$$

At internal integer nodes,

$$\boldsymbol{Q}_j = \frac{\frac{1}{6}\widetilde{\boldsymbol{L}}_j(x_j)h_{x,j-1/2} + \frac{1}{6}\widetilde{\boldsymbol{L}}_{j+1}(x_j)h_{x,j+1/2}}{\frac{1}{6}h_{x,j-1/2} + \frac{1}{6}h_{x,j+1/2}} = \frac{\widetilde{\boldsymbol{L}}_j(x_j)h_{x,j-1/2} + \widetilde{\boldsymbol{L}}_{j+1}(x_j)h_{x,j+1/2}}{h_{x,j-1/2} + h_{x,j+1/2}},$$

$$j = \overline{1, N_x - 1}. \quad (19)$$

In the special case of a uniform grid, formula (19) takes an especially simple form:

$$\boldsymbol{Q}_j = \frac{\widetilde{\boldsymbol{L}}_j(x_j) + \widetilde{\boldsymbol{L}}_{j+1}(x_j)}{2}, \quad j = \overline{1, N_x - 1},$$

i. e., the resulting solution at each internal integer node is equal to the half-sum of the limits, at this node, of corrected finite-element approximations in cells whose common boundary contains this node.

3. Analogy between Galerkin and bicompact schemes

Interestingly, there is an analogy between Galerkin and bicompact schemes, which is demonstrated as applied to system (1). Its exact solution in the cell K_{j+1} is approximated by a linear combination of the basis polynomials $\varphi_l(x)$ (see Section 1) of maximum degree $l_{\max}$:

$$\boldsymbol{Q}(x,t) \approx \boldsymbol{Q}_h(x,t;K_{j+1}) = \sum_{l=0}^{l_{\max}} \boldsymbol{c}_l(t)\varphi_l(x). \tag{20}$$

Equations of a semi-discrete Galerkin scheme for system (1) are derived from the condition of orthogonality of the residual to the first $(l_{\max}+1)$ basis functions of the system $\{\varphi_l(x)\}_{l=0}^{\infty}$:

$$(\mathscr{L}_1(\boldsymbol{Q}_h), \varphi_l) = \boldsymbol{0}, \quad l = \overline{0, l_{\max}}, \tag{21}$$

where the scalar product is given by formula (4). In their final form, Eqs. (21) are written as

$$\frac{d\boldsymbol{c}_l}{dt} + \frac{\varphi_l(x_{j+1})\widehat{\boldsymbol{F}}_{j+1} - \varphi_l(x_j)\widehat{\boldsymbol{F}}_j}{h_x} - \frac{1}{h_x}\int\limits_{x_j}^{x_{j+1}} \boldsymbol{F}[\boldsymbol{Q}_h(x,t;K_{j+1})]\frac{d\varphi_l(x)}{dx}\,dx = \boldsymbol{0},$$

$$l = \overline{0, l_{\max}}. \tag{22}$$

There are two versions of scheme (22). If approximation (20) can undergo strong discontinuities on cell boundaries, then we have a discontinuous Galerkin scheme: Eqs. (22) are solved for all $\boldsymbol{c}_l$, $l = \overline{0, l_{\max}}$, and the numerical fluxes $\widehat{\boldsymbol{F}}_j$ on cell boundaries are found using the exact or some approximate solution of Riemann problems in small neighborhoods of the points (x_j, t). If approximation (20) has to be continuous on the entire interval $[0, x_{\max}]$, then we have a continuous Galerkin scheme. Due to the additional continuity constraint, the number of Eqs. (22) is reduced by one (in each cell) and the numerical fluxes are $\widehat{\boldsymbol{F}}_j = \boldsymbol{F}[\boldsymbol{Q}_h(x_j,t;K_{j+1})] = \boldsymbol{F}(\boldsymbol{Q}_j) = \boldsymbol{F}_j$.

Consider the continuous version of Galerkin scheme (22) in the special case of $l_{\max} = 2$. Out of Eqs. (22), we retain the first two ($l = 0, 1$) and the coefficients $\boldsymbol{c}_2$ are determined assuming that the numerical solution is continuous on cell boundaries. The equation for $\boldsymbol{c}_0$ is

$$\frac{d\boldsymbol{c}_0}{dt} + \frac{\boldsymbol{F}_{j+1} - \boldsymbol{F}_j}{h_x} = \boldsymbol{0}. \tag{23}$$

The equation on $\boldsymbol{c}_1$ has the form

$$\frac{d\boldsymbol{c}_1}{dt} + \frac{\sqrt{3}}{h_x}\left[\boldsymbol{F}_j - \frac{2}{h_x}\int\limits_{x_j}^{x_{j+1}} \boldsymbol{F}[\boldsymbol{Q}_h(x,t;K_{j+1})]\,dx + \boldsymbol{F}_{j+1}\right] = \boldsymbol{0}. \tag{24}$$

Let us approximate the integral of $\boldsymbol{F}$ in Eq. (24) using Simpson's rule with an $\boldsymbol{O}(h_x^4)$ error:

$$\frac{1}{h_x}\int\limits_{x_j}^{x_{j+1}} \boldsymbol{F}[\boldsymbol{Q}_h(x,t;K_{j+1})]\,dx \approx \frac{\boldsymbol{F}_j + 4\boldsymbol{F}_{j+1/2} + \boldsymbol{F}_{j+1}}{6}. \tag{25}$$

Substituting (25) into (24) yields

$$\frac{d\boldsymbol{c}_1}{dt} + \frac{2(\boldsymbol{F}_j - 2\boldsymbol{F}_{j+1/2} + \boldsymbol{F}_{j+1})}{\sqrt{3}\,h_x} = \boldsymbol{0}. \tag{26}$$

Since the finite-element approximation (20) is continuous, its coefficients are expressed in terms of the nodal values of $\boldsymbol{Q}_h(x,t;K_{j+1})$ using formulas (6) with u replaced by $\boldsymbol{Q}$. The expression for $\boldsymbol{c}_0$ in terms of solution values at nodes of Ω is substituted into Eq. (23). Then, using the difference operators A_0^x and Λ_1^x for notational brevity, we obtain

$$\frac{d}{dt}(A_0^x \boldsymbol{Q}_{j+1/2}) + \Lambda_1^x \boldsymbol{F}_{j+1/2} = \boldsymbol{0},$$

i. e., the first equation of the semi-discrete bicompact scheme (2). Similar transformations with Eq. (26) yield

$$\frac{d}{dt}(\Lambda_1^x \boldsymbol{Q}_{j+1/2}) + \Lambda_2^x \boldsymbol{F}_{j+1/2} = \boldsymbol{0},$$

i. e., the second equation of scheme (2).

It follows from what was said above that, for small mesh steps, the equations of the semi-discrete bicompact scheme are close to those of the semi-discrete continuous Galerkin scheme, which is what we mean by the analogy between these classes of schemes. This analogy suggests two conclusions. First, the discretization of differential consequences of the system of differential equations can be treated as equations for determining the higher-order coefficients of the finite-element approximation. Second, bicompact schemes have not only a classical (strong) approximation but also a weak one; accordingly, they can be expected to converge to discontinuous solutions under mesh refinement.

4. Testing of the method on one-dimensional gas dynamics problems

Let us test the conservative limiting method developed for bicompact schemes in Section 2 on one-dimensional gas dynamics problems.

System of equations. The system of one-dimensional gasdynamic Euler equations has the form of (1), where

$$\boldsymbol{Q} = \begin{bmatrix} \rho \\ \rho v \\ E \end{bmatrix}, \quad \boldsymbol{F}(\boldsymbol{Q}) = \begin{bmatrix} \rho v \\ \rho v^2 + p \\ v(E+p) \end{bmatrix}, \quad E = \frac{p}{\gamma - 1} + \frac{\rho v^2}{2}.$$

Here ρ, v, p, and E denote the density, velocity, pressure, and specific total energy (per unit volume), respectively, and $\gamma = \text{const}$ is the ratio of specific heats. Below, in all problems, the gas is diatomic and $\gamma = 1.4$.

Scheme and its parameters. As schemes A and B, we use bicompact schemes of fourth-order accuracy in x obtained from the semi-discrete scheme (2). Integration with respect to t in scheme A is performed by the implicit Euler method (baseline scheme), while scheme B is based on an L-stable stiffly accurate three-stage SDIRK method of third order [24, Eq. (17)]. For smooth solutions, the truncation errors of schemes A and B are $\boldsymbol{O}(h_x^4, \tau)$ and $\boldsymbol{O}(h_x^4, \tau^3)$ respectively. Scheme A is monotonic for Courant numbers not smaller than 0.25.

The parameter $C_1 \geqslant 0$ is chosen depending on the problem. The flux splitting parameter C_2^x (see [19]) is automatically updated before each transition from the level t^n to t^{n+1} by applying the formula

$$C_2^x = \frac{1+2\delta}{2} V_{\max}^x, \quad V_{\max}^x = \max_{\substack{s=\overline{1,m} \\ x \in \Omega}} |\lambda_s(\boldsymbol{Q}^n(x); \boldsymbol{A})|,$$

where $\lambda_s(\boldsymbol{Q}; \boldsymbol{X})$ is the sth eigenvalue of the matrix $\boldsymbol{X}(\boldsymbol{Q})$ and $\delta > 0$ is a "reserve factor of positive/negative definiteness" of the Jacobian matrices of split fluxes. In what follows, we use $\delta = 0.2$ everywhere. The time step is variable and is also computed automatically:

$$\tau = \tau^{n+1} = t^{n+1} - t^n = \frac{2\kappa h_x}{V_{\max}^x + 2C_2^x},$$

where $\kappa = \text{const}$ is the maximum Courant number, which is a given parameter. Uniform grids in x are used in all problems.

Since negative values of ρ and E are unacceptable, the weighting factors α_s have to quickly become equal to 1 as zero densities or specific energies

are approached. Accordingly, the expression for w_s in formula (13) is replaced by

$$w_s = \frac{C_1|Q_{As}(\boldsymbol{r}_i) - Q_{Bs}(\boldsymbol{r}_i)|}{\min\{\sigma \max_{K_i}|Q_{As}|,\ \max_{K_i} Q_{As} - \min_{K_i} Q_{As}\}}, \quad s = \overline{1,m},$$

where $\sigma > 0$ is a small factor. In what follows, we everywhere use $\sigma = 0.1$.

The nonlinear equations of schemes A and B are solved by Newton's method up to the relative error $\mathrm{rtol} = 10^{-9}$.

Sedov blast wave problem. We begin with the well-known Sedov problem of a strong blast in an ideal gas [25]. Consider a blast with plane symmetry. This problem is remarkable in that the nonconservative behavior of hybrid schemes [18, 20, 21] is clearly manifested in it.

The initial and boundary conditions are set as follows. The blast occurs at the time $t = 0$ at the point $x = x_0 = 0.5$; $x_{\max} = 1$. The blast energy $\mathscr{E}_0$ is specified so that the shock waves have traveled a distance of 0.4 from the blast point by the time $t = t_{\max} = 0.01$; namely, $\mathscr{E}_0 = 689.593$. The initial conditions are set as follows: for $x \in \Omega$,

$$\rho(x,0) = 1,\ v(x,0) = 0,\ E(x,0) = \begin{cases} \mathscr{E}_0/h_x & \text{at } x = x_0, \\ 0.5\mathscr{E}_0/h_x & \text{at } x = x_0 \pm 0.5h_x, \\ 10^{-2}/(\gamma - 1) & \text{otherwise.} \end{cases}$$

The boundary conditions are constant.

The computations were performed on grids with $N_x = 100, 200, 400$ cells at the Courant number $\kappa = 0.8$ and $C_1 = 0.5$.

The computed density, pressure, and velocity profiles at a final time are presented on Figs. 1–3, respectively. The values of numerical solutions at integer nodes are shown by color markers, and the exact solution is depicted by the black solid curve. It can be seen that the bicompact scheme with the new limiting method provides a very good resolution of the shock waves (on 3–4 cells). Additionally, this scheme reproduces sharp peaks of density and pressure quite well. Importantly, the coordinates of the shock waves in the exact and numerical solutions are nearly indistinguishable from each other. Note that the integral of the numerical $\boldsymbol{Q}$ preserves its value with a relative error of order rtol in each component.

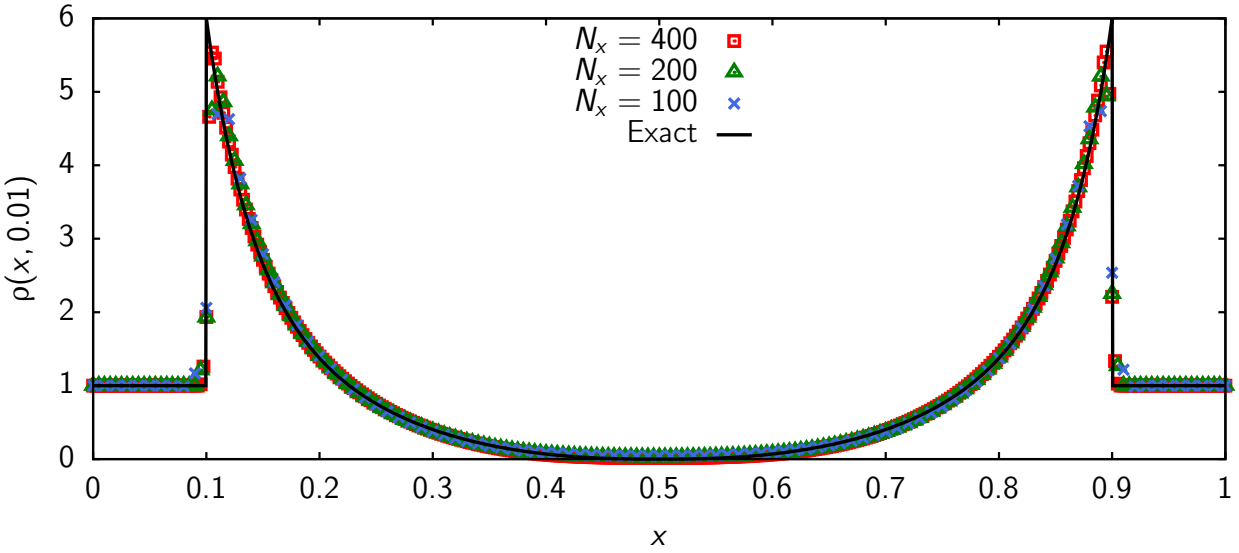

Fig. 1. Density profiles in the Sedov blast wave problem at $t = t_{\max} = 0.01$

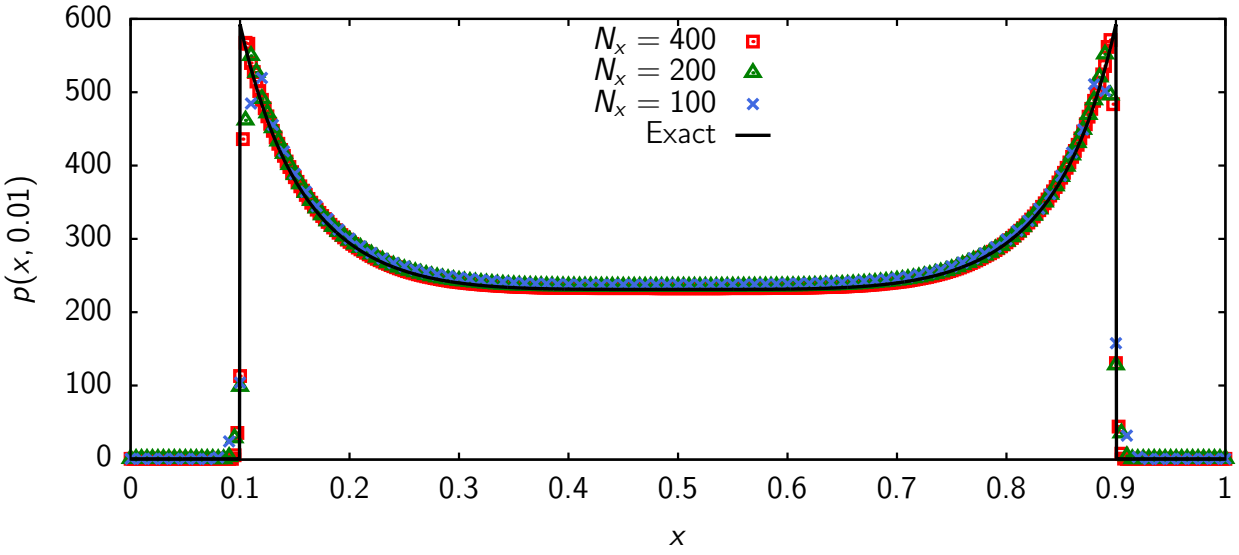

Fig. 2. Pressure profiles in the Sedov blast wave problem at $t = t_{\max} = 0.01$

Hybrid schemes [18, 20, 21] are not conservative: generally speaking, a convex weighting of solutions of two conservative schemes does not necessarily yield a solution obeying some conservation law. In all previously computed problems, the hybrid schemes [18, 20, 21] did not exhibit a perceptible nonconservative behavior. However, in the Sedov blast wave problem, the hybrid scheme [21] noticeably underestimates the velocities of shock waves (by several tens of percent); moreover, mesh refinement does not lead to convergence to the exact solution. In contrast to [21], the hybrid schemes from [18, 20] make use of C_1/τ rather than C_1 in the formula for weighting factors. As a result, the schemes [18, 20] produce nearly correct shock wave velocities under

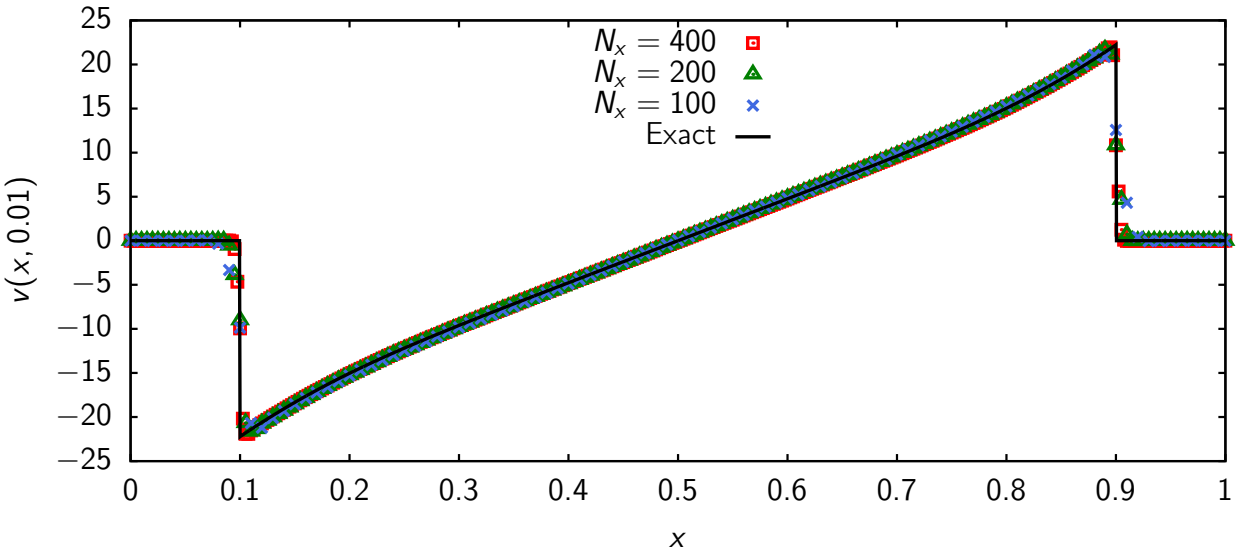

Fig. 3. Velocity profiles in the Sedov blast wave problem at $t = t_{\max} = 0.01$

mesh refinement, but this requires large values of C_1, which lead to unsatisfactorily high numerical dissipation (for example, a density peak will be at the level $\rho = 3$ even on a grid with $N_x = 800$). Presumably, the above-described effects and features of the Sedov problem are explained by the fact that its initial condition depends substantially on the step h_x, i. e., $E(x_0, 0) \sim h_x^{-1}$.

"Peak test" Riemann problem. Now we consider the one-dimensional "peak test" Riemann problem [26]. This problem was chosen for several reasons. First, it involves both a strong shock wave and an intense contact discontinuity with a density difference of ≈ 300 times; moreover, the density value on one side of the contact discontinuity is close to zero. Even a small nonmonotonicity imposed on this small density field can make the scheme fail. Second, the density profile at $t = t_{\max}$ contains a narrow peak, which cannot be resolved by some schemes. Third, this narrow peak is made up of the above-mentioned contact discontinuity and shock wave: a good resolution of such a structure requires low dissipation, which may lead to dangerous nonmonotonicity near the "difficult" contact discontinuity; high dissipation protects from nonmonotonicities, but worsens the resolution of the peak.

In contrast to [26], the computations were performed not on the interval $[0.1, 0.6]$, but rather on $[0, 1]$ ($x_{\max} = 1$) with twice as large N_x. More specifically, we used grids with $N_x = 1600, 3200$. The number of cells $N_x = 1600$ corresponded to $h_x = h_x^* = 6.25 \cdot 10^{-4}$, which is a standard (for comparison) stepsize in x for this problem. The Courant number was $\kappa = 1$, and $C_1 = 1$. It should be noted that bicompact schemes do not yield an exact solution for $\kappa = 1$, in contrast to some explicit schemes in cer-

tain problems. Moreover, bicompact schemes do not yield an exact solution at any Courant number and the situation with these schemes worsens for larger values of κ, since the numerical dissipation in bicompact schemes grows with increasing κ.

The computed density profiles at a final time are displayed on Fig. 4. It can be seen that the bicompact scheme with the conservative limiting method predicts the shock wave location without errors. The difference of the integrals of the numerical $\boldsymbol{Q}$ at the times $t = 0$ and $t = t_{\max}$ deviates from the value $[t_{\max}(\mathscr{F}(0,0) - \mathscr{F}(x_{\max},0))]$ with a relative error of order rtol in all components.

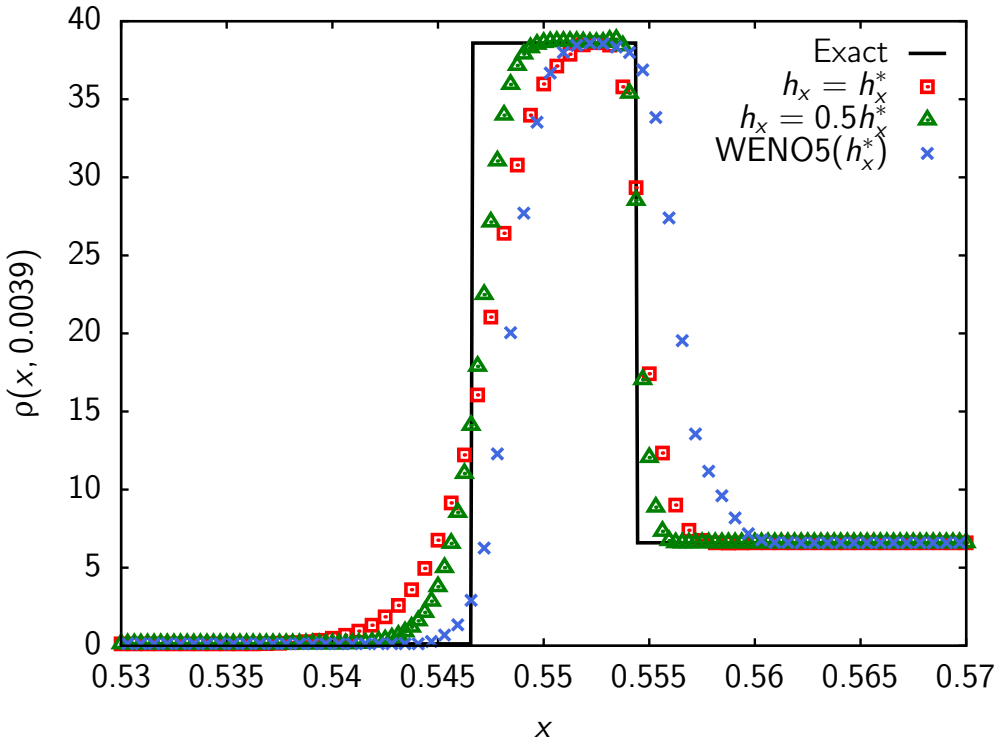

Fig. 4. Density profiles near the narrow peak in the "peak test" Riemann problem at $t = t_{\max} = 0.0039$

As compared with the classical WENO5 [15], the bicompact scheme provides a somewhat better resolution of the shock wave (by 4 cells). Note that WENO5 propagates the peak with a slightly overestimated velocity, though without errors in the integrals of $\boldsymbol{Q}$. Both schemes give three points on the "top" of the peak. Note that the conservative limiting method is much better than the hybrid scheme of [18]: even on a four-time denser grid with step $0.25h_x^*$, the solution of the scheme [18] deviates from the top of the peak.

Shu-Osher problem. The well-known Shu-Osher test problem [14] concerning the interaction of a shock wave and a sinusoidal density background

has not yet been computed using bicompact schemes. It is interesting to determine the accuracy at which the designed limiting method transmits smooth perturbations through the shock wave surface.

In contrast to [14], we solved this problem on the interval $[0, 10]$ ($x_{\max} = 10$), rather than on $[-5, 5]$. The computations were performed on grids with $N_x = 200, 400, 800$. As an "exact solution", we used the converged numerical solution produced by the tested bicompact scheme on a grid with $N_x = 2000$. We also used the Courant number $\kappa = 0.5$ and $C_1 = 0.5$.

The computed density profiles at a final time are presented on Fig. 5. Differences between the numerical solutions can be seen only about the points $x = 5.75$ and $x = 7.35$ (near a small sharp peak adjoining the shock wave on its left side). The solution on the grid with $N_x = 200$ has smaller amplitudes of ρ behind the shock wave, which moves from left to right. On all grids, the bicompact scheme resolves the shock wave over two cells. The shock wave location is correctly reproduced (judging from other known solutions) and the global conservation laws are satisfied with a relative error of order rtol, as in the previous problems.

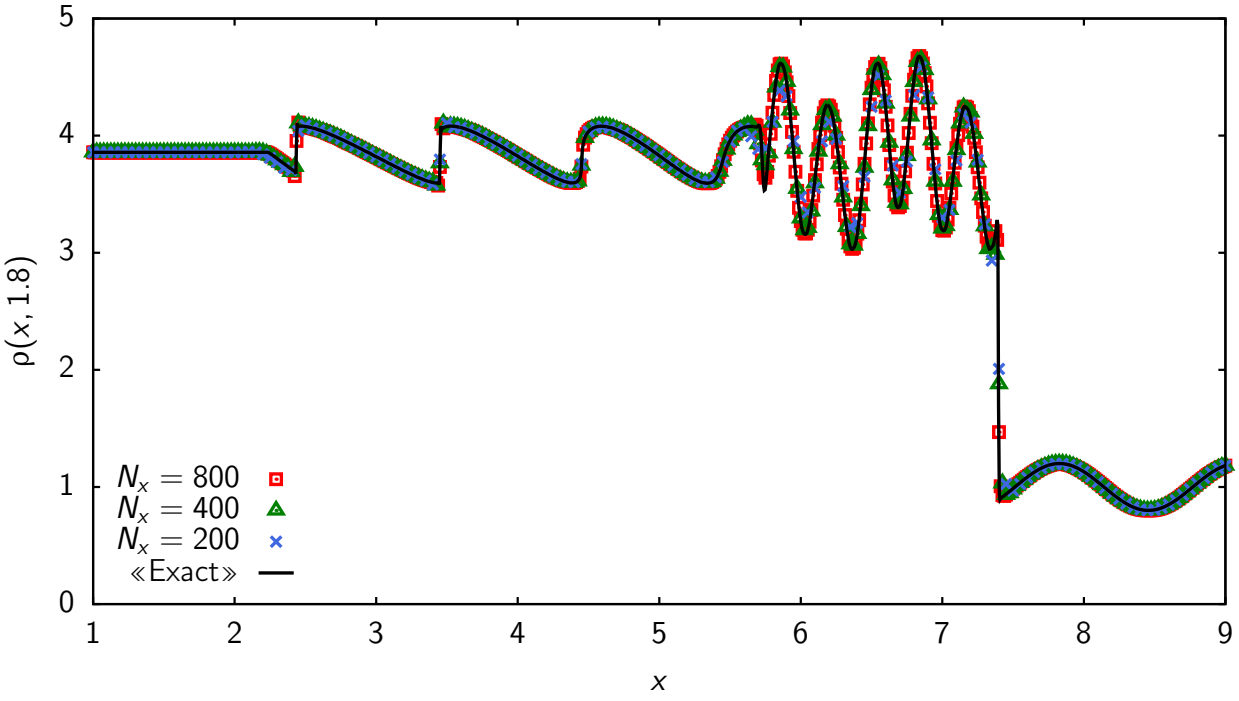

Fig. 5. Density profiles in the Shu-Osher problem at $t = t_{\max} = 1.8$

Fig. 6 shows a zoomed fragment of Fig. 5 for $x \in [5.5, 7.5]$, $\rho \in [2.75, 4.75]$. Relying on this plot, we can better estimate the order of accuracy of the scheme on the "sine curve" localized within the interval $[5.8, 7.3]$. The solution on the grid with $N_x = 200$ goes fairly close to the "exact" one, but

the differences between them are noticeable. With a halved mesh size, the solution becomes nearly indistinguishable from the "exact" one for $x \in [5.8, 7.3]$, which suggests that the scheme has a high order of accuracy in this domain. A quantitative comparison of numerical solutions produced by the Runge method at the extrema of the sine-like curve gives an order of accuracy ≈ 3.15. This result agrees well with the theoretical order of accuracy of the scheme.

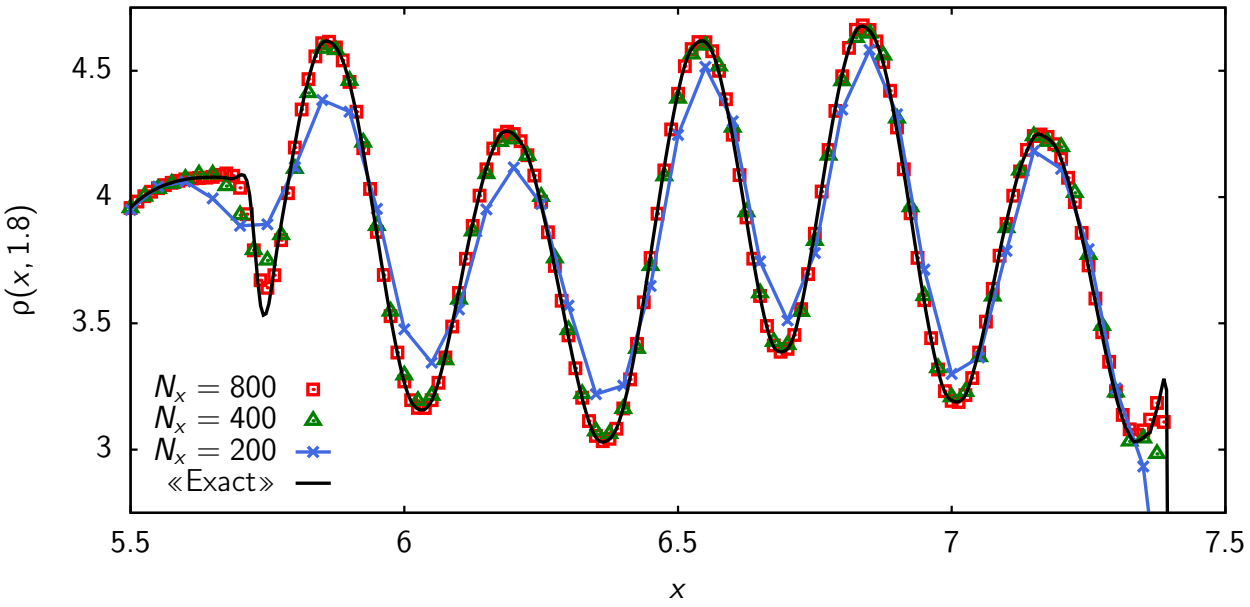

Fig. 6. Zoomed fragment of Fig. 5 for $x \in [5.5, 7.5]$, $\rho \in [2.75, 4.75]$

Conclusions

A finite-element representation of the bicompact approximation was obtained. Based on this representation, a conservative limiting method for bicompact schemes was constructed for the first time. The principal idea underlying the method is that the higher-order coefficients of a continuous finite-element approximation are corrected using weighting factors of a hybrid scheme. The correction in each cell is local and independent of the neighboring cells. Since this correction violates the continuity of the finite-element approximation, the application of the method is completed with a conservative merging of the approximation polynomials on the cell boundaries.

An analogy between Galerkin and bicompact schemes was established. This result has two consequences. First, discretizations of differential consequences of systems of differential equations can be treated as equations for the higher-order coefficients of the finite-element approximation. Second, bicompact schemes have the property of a weak approximation.

The limiting method developed for bicompact schemes was tested as applied to one-dimensional gas dynamics problems, namely, the Sedov blast wave problem, the “peak test” Riemann problem, and the Shu-Osher problem. The numerical results suggest that bicompact schemes with the conservative limiting method guarantee the fulfillment of a numerical conservation law (which is important, for example, in the Sedov problem). Moreover, such bicompact schemes are much more accurate than previously applied hybrid bicompact schemes.

Bibliography list

1. Rogov B. V. Dispersive and dissipative properties of the fully discrete bicompact schemes of the fourth order of spatial approximation for hyperbolic equations // Appl. Numer. Math. — 2019. — Vol. 139. — P. 136–155.

2. Godunov S. K. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics // Mat. Sb. (NS). — 1959. — Vol. 47 (89), no. 3. — P. 271–306.

3. Cockburn B., Shu C.-W. Nonlinearly stable compact schemes for shock calculations // SIAM J. Numer. Anal. — 1994. — Vol. 31, no. 3. — P. 607–627.

4. Yee H. C. Explicit and implicit multidimensional compact high-resolution shock-capturing methods: Formulation // J. Comput. Phys. — 1997. — Vol. 131, no. 1. — P. 216–232.

5. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws / L. Krivodonova, J. Xin, J.-F. Remacle et al. // Appl. Numer. Math. — 2004. — Vol. 48. — P. 323–338.

6. Yee H. C., Sandham N. D., Djomehri M. J. Low-dissipative high-order shock-capturing methods using characteristic-based filters // J. Comput. Phys. — 1999. — Vol. 150, no. 1. — P. 199–238.

7. Ekaterinaris J. A. High-order accurate, low numerical diffusion methods for aerodynamics // Prog. Aerosp. Sci. — 2005. — Vol. 41. — P. 192–300.

8. Yee H. C., Sjögreen B. Adaptive filtering and limiting in compact high order methods for multiscale gas dynamics and MHD systems // Comput. Fluids. — 2008. — Vol. 37, no. 5. — P. 593–619.

9. Von Neumann J., Richtmyer R. D. A method for the numerical calculation of hydrodynamic shocks // J. Appl. Phys. — 1950. — Vol. 21, no. 3. — P. 232–237.

10. Ostapenko V. V. Symmetric compact schemes with higher order conservative artificial viscosities // Comput. Math. Math. Phys. — 2002. — Vol. 42, no. 7. — P. 980–999.

11. Guermond J.-L., Pasquetti R., Popov B. Entropy viscosity method for nonlinear conservation laws // J. Comput. Phys. — 2011. — Vol. 230. — P. 4248–4267.

12. Kurganov A., Liu Y. New adaptive artificial viscosity method for hyperbolic systems of conservation laws // J. Comput. Phys. — 2012. — Vol. 231, no. 24. — P. 8114–8132.

13. Uniformly high order accurate essentially non-oscillatory schemes, III / A. Harten, B. Engquist, S. Osher, S. R. Chakravarthy // J. Comput. Phys. — 1987. — Vol. 71, no. 2. — P. 231–303.

14. Shu C.-W., Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II // J. Comput. Phys. — 1989. — Vol. 83. — P. 32–78.

15. Jiang G.-S., Shu C.-W. Efficient implementation of weighted ENO schemes // J. Comput. Phys. — 1996. — Vol. 126. — P. 202–228.

16. Qiu J., Shu C.-W. Runge-Kutta discontinuous Galerkin method using WENO limiters // SIAM J. Sci. Comput. — 2005. — Vol. 26, no. 3. — P. 907–929.

17. Rogov B. V., Mikhailovskaya M. N. Monotone bicompact schemes for a linear advection equation // Dokl. Math. — 2011. — Vol. 83, no. 1. — P. 121–125.

18. Mikhailovskaya M. N., Rogov B. V. Monotone compact running schemes for systems of hyperbolic equations // Comput. Math. Math. Phys. — 2012. — Vol. 52, no. 4. — P. 578–600.

19. Chikitkin A. V., Rogov B. V., Utyuzhnikov S. V. High-order accurate monotone compact running scheme for multidimensional hyperbolic equations // Appl. Numer. Math. — 2015. — Vol. 93. — P. 150–163.

20. Bragin M. D., Rogov B. V. Minimal dissipation hybrid bicompact schemes for hyperbolic equations // Comput. Math. Math. Phys. — 2016. — Vol. 56, no. 6. — P. 947–961.

21. Bragin M. D., Rogov B. V. A new hybrid scheme for computing discontinuous solutions of hyperbolic equations // Keldysh Institute Preprints. — 2016. — no. 22. — 20 p.

22. Fedorenko R. P. The application of difference schemes of high accuracy to the numerical solution of hyperbolic equations // Comput. Math. Math. Phys. — 1962. — Vol. 2, no. 6. — P. 1355–1365.

23. Rogov B. V. High-order accurate monotone compact running scheme for multidimensional hyperbolic equations // Comput. Math. Math. Phys. — 2013. — Vol. 53, no. 2. — P. 205–214.

24. Skvortsov L. M. Diagonally implicit Runge–Kutta FSAL methods for stiff and differential-algebraic systems // Matem. Mod. — 2002. — Vol. 14, no. 2. — P. 3–17.

25. Sedov L. I. Similarity and dimensional methods in mechanics. — 10th edition. — Boca Raton, Florida : CRC Press, 1993.

26. Liska R., Wendroff B. Comparison of several difference schemes on 1D and 2D test problems for the Euler equations // SIAM J. Sci. Comput. — 2003. — Vol. 25, no. 3. — P. 995–1017.