

# Numerical simulation of compressible gas flows using regularized gas dynamic equations solver QGDFoam

Cite as: AIP Conference Proceedings **2027**, 030105 (2018); <https://doi.org/10.1063/1.5065199>  
Published Online: 02 November 2018

M. V. Kraposhin, T. G. Elizarova, M. A. Istomina, D. A. Ryazanov, and K. A. Vatutin



View Online



Export Citation

## ARTICLES YOU MAY BE INTERESTED IN

[High speed flow verification using open source CFD software](#)

AIP Conference Proceedings **1768**, 020038 (2016); <https://doi.org/10.1063/1.4963060>

[A theoretical approximation of the shock standoff distance for supersonic flows around a circular cylinder](#)

Physics of Fluids **29**, 026102 (2017); <https://doi.org/10.1063/1.4975983>

[A multicomponent real-fluid fully compressible four-equation model for two-phase flow with phase change](#)

Physics of Fluids **31**, 026102 (2019); <https://doi.org/10.1063/1.5065781>

## Lock-in Amplifiers up to 600 MHz



Zurich  
Instruments



# Numerical Simulation of Compressible Gas Flows Using Regularized Gas Dynamic Equations Solver QGDFoam

M.V. Kraposhin<sup>1,a)</sup>, T.G. Elizarova<sup>2,b)</sup>, M.A. Istomina<sup>2</sup>, D.A. Ryazanov<sup>1,3</sup> and K.A. Vatutin<sup>1,3</sup>

<sup>1</sup>*Ivannikov Institute for System Programming of RAS, Moscow, Russia*

<sup>2</sup>*Keldysh Institute of Applied Mathematics of RAS, Moscow, Russia*

<sup>3</sup>*Lomonosov Moscow State University, Moscow, Russia*

<sup>a)</sup>Corresponding author: m.kraposhin@ispras.ru

<sup>b)</sup>telizar@mail.ru

**Abstract.** The work presents results of the application of a new OpenFOAM<sup>®</sup> solver *QGDFoam* for the numerical simulation of viscous compressible flows in a wide range of Mach numbers. The new solver is based on the explicit approximation of regularized, or quasi-gas dynamic (QGD) equations. The mixed finite-volume and finite-difference approximation is constructed on unstructured spatial grids with co-located variables storage. The solver has been tested for a number of 1D Riemann problems (Sod's problem, Noh test and others) and 2D cases (Mach 3 forward step, Ladenburg supersonic jet flow with Mach reflection, NASA Langley supersonic overexpanded jet flow and subsonic laminar flow over a backward-facing step). Results of numerical simulations were compared with analytic solutions and OpenFOAM<sup>®</sup> implementation of the Kurganov-Tadmor scheme, known as rhoCentralFoam. The testing procedure has shown that whereas QGD algorithm is more diffusive than Godunov-type methods with 2nd order TVD schemes with limiters, it is far less diffusive compared with pure upwind schemes as HLL. It was shown that OpenFOAM implementation of the QGD algorithm allows to compute successfully subsonic, sonic and supersonic flows, while other OpenFOAM<sup>®</sup> solvers have a very limited operational Mach number range. Preliminary results of QGDFoam application for large-scale 3D problems are presented. Scaling tests for up to 768 cores showed good scalability of *QGDFoam* solver.

## INTRODUCTION

The solver *QGDFoam* implementing the QGD numerical algorithm was developed in [1] and is now being under maintenance. The main advantages of the QGD-algorithm are:

- single numerical approach for flows with different speeds: from subsonic to hypersonic, from laminar to fully developed turbulence;
- explicit schemes with simple approximation of spatial terms yielding good scalability for HPC computations;
- single set of tuning parameters for all kind of simulation: relaxation time  $\tau$ , maximum Courant number  $Co^{max}$  and additional viscosity coefficient  $\Delta^{QGD}$ .

Recommendations for choosing values of  $\tau$ ,  $Co^{max}$  and  $\Delta^{QGD}$  were deduced analytically for a few simple special cases (one-dimensional, with uniform orthogonal grids, at particular Mach and Re numbers and so on) [2]. Though, there is no justified procedure for a general case which can be applied to simulate complex geometries with arbitrary Ma and Re numbers, such as turbulent wakes after wind mills[3] on polyhedral meshes. This work aims at the further studying the adjustment of tuning parameters in the QGD-algorithm, for a case of industrial interest, namely the supersonic flow around a blunt body. A secondary objective of the research is the measurement of parallel execution efficiency of the *QGDFoam* solver.

# QGDFOAM — THE OPEN-SOURCE IMPLEMENTATION OF QGD NUMERICAL ALGORITHM

Modern commercial and open-source general-purpose CFD programs are dominated by three types of numerical algorithms:

- operator splitting techniques (PISO, SIMPLE and other) for viscous subsonic flows;
- meshless methods for unsteady external subsonic flows around moving bodies, such as Vortex Method (see [4]);
- Godunov-type or flux-splitting methods for inviscid flows.

Each of those approaches imposes strict restrictions on the application area. Justified by assumptions about the numerical solution, these restrictions do not allow to simulate regimes where both subsonic and supersonic types of flow are present. These restrictions are removed in a new system of equations known as regularized or quasi-gas dynamic equations.

In order to extend the scope of application of QGD numerical algorithms and to simplify the process of verification and validation, *QGD Foam* solver has been developed on top of OpenFOAM® platform. It encompasses the library *libQGD* for approximation of  $\tau$ -terms and solver *QGD Foam* for the simulation of compressible viscous perfect-gas flows.

## QGD equations

Like Navier-Stokes equations, the QGD system describes the evolution of gas density  $\rho$ , velocity  $\vec{U}$  and pressure  $p$ , as functions of space co-ordinates and time. But in contrast with the Navier-Stokes equations, the QGD parameters are regarded as averaged, or smoothed values over some small time interval. Such smoothing of the gas dynamic parameters  $\rho$ ,  $p$  and  $\vec{U}$  leads to the appearance of additional dissipative terms in the corresponding equations with an additional dissipative coefficient, which has the dimension of a time and is denoted as  $\tau$ , e.g. [5, 2, 6, 7]. It is related to the averaging time. When  $\tau$  tends to zero, the system of QGD equations, which describes the evolution of smoothed gas-dynamic parameters, reduces to Navier-Stokes equations. The dissipative nature of  $\tau$ -terms is ensured by the existence of the non-negative dissipative function for QGD equations system.

The QGD system in Cartesian coordinates writes:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{j}_m = 0, \quad (1)$$

$$\frac{\partial \rho \vec{U}}{\partial t} + \nabla \cdot (\vec{j}_m \otimes \vec{U}) + \nabla p = \nabla \cdot \hat{\Pi}, \quad (2)$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\vec{j}_m h^{tot}) + \nabla \cdot \vec{q} = \nabla \cdot (\hat{\Pi} \cdot \vec{U}). \quad (3)$$

Here, for simplicity reasons, external forces and heat sources are omitted, and the gas is supposed to be ideal. The full system can be found, for example, in [2]. The total energy per unit volume  $\rho e$  and the total specific enthalpy  $h^{tot}$  are defined as  $\rho e = \rho u + \rho \frac{1}{2}(\vec{U} \cdot \vec{U})$  and  $h^{tot} = e + p/\rho$ , where  $u$  is specific internal energy. The mass flux density  $\vec{j}_m$  is given by:

$$\vec{j}_m = \rho(\vec{U} - \vec{w}), \quad \vec{w} = \frac{\tau}{\rho} (\nabla \cdot (\rho \vec{U} \otimes \vec{U}) + \nabla p). \quad (4)$$

The viscous stress tensor  $\hat{\Pi}$  and the heat flux  $\vec{q}$  write

$$\hat{\Pi} = \hat{\Pi}_{NS} + \tau \vec{U} \otimes \rho(\vec{U} \cdot \nabla \vec{U}) + \frac{1}{\rho} \nabla p + \tau \hat{I}(\vec{U} \cdot \nabla p + \gamma p \nabla \cdot \vec{U}), \quad (5)$$

$$\hat{\Pi}_{NS} = \mu(\nabla \vec{U} + (\nabla \vec{U})^T) - \hat{I} \frac{2}{3} \nabla \cdot \vec{U}, \quad (6)$$

$$\vec{q} = \vec{q}_{NS} - \tau \vec{U} \rho(\vec{U} \cdot \nabla u + p \vec{U} \cdot \nabla(\frac{1}{\rho})), \quad \vec{q}_{NS} = -\kappa \nabla T. \quad (7)$$

Here,  $\gamma$  is the adiabatic exponent,  $\hat{I}$  the unit tensor. The internal energy per unit mass for a perfect-gas is  $u = p/(\rho(\gamma - 1))$ . Pressure, density and temperature  $T$  are linked through the perfect-gas equation-of-state (EoS):

$$p = \rho RT, \quad (8)$$

where  $R$  is the unit-mass perfect gas constant. The thermal conductivity is given by

$$\kappa = \frac{\mu C_p}{\text{Pr}}, \quad (9)$$

where  $\text{Pr}$  is the Prandtl number,  $C_p$  is the specific heat capacity at constant pressure. The dynamic viscosity  $\mu$  contained in expressions (5)–(9) for  $\hat{\Pi}$  and  $\vec{q}$  is defined as a function of temperature:

$$\mu = \mu(T), \quad (10)$$

### QGD equations approximation

The system of QGD equations (1)–(3) has been approximated with the Finite Volume Method (FVM) implemented in open-source library OpenFOAM<sup>®</sup>. The choice of this approach is justified by the:

- successful application of FVM on staggered irregular triangular meshes for QGD equations, see [2];
- conservativeness and boundedness properties of Finite Volume Method.

This implementation of FVM uses co-located storage with compact stencil. According to this approach the computational domain is approximated as a set of non-overlapping volumes of arbitrary shape, connected to each other only through one common face, [8]. The unknown gas dynamics fields are averaged over cell volumes. The averaged values are stored at cell centers  $P$ .

Balance equations describing the flow are approximated in the integral formulation using the Ostrogradsky-Gauss theorem to replace the volume integral with a surface integral. The mean value theorem is used to calculate the surface and volume integrals of approximated functions.

Traditionally, in most of OpenFOAM<sup>®</sup> applications operator-splitting techniques are used (such as PISO, SIMPLE and their modifications) to solve pressure-velocity linked equations at low speeds, and Euler or multi-stage Runge-Kutta schemes with approximate Riemann solvers (Kurganov—Tadmor, HLLC, AUSM+) for high-speed flows [9]. The matrix-coupled approach is used more rarely for pressure-velocity coupled simulations of incompressible fluids. The hybrid pressure-based and Kurganov—Tadmor approach was proposed [10].

QGD equations contain terms which need a special approximation procedure in OpenFOAM<sup>®</sup>. These terms are denoted as “QGD”-fluxes. They have been approximated with the Least-Squares Method on unstructured meshes, which is second-order accurate in space.

The semi-implicit approach, similar to [9] was used to approximate QGD equations. According to procedure [9] diffusive terms are approximated using an implicit scheme, while other terms are approximated using an explicit scheme. It has been shown that the explicit approximation of QGD equations is conditionally stable.

*Mass conservation equation*

$$\rho^n = \rho^o - \frac{\Delta t}{V} \sum_f \Phi_f(\rho^o)$$

$$\Phi_f(\rho^o) = \rho_f \vec{U}_f \cdot \vec{S}_f - \tau_f \left( \left[ \nabla \cdot (\rho \vec{U} \otimes \vec{U}) \right]_f + [\nabla p]_f \right) \cdot \vec{S}_f$$

*Momentum balance equation*

$$\begin{aligned}
(\rho \vec{U})^p &= (\rho \vec{U})^o - \frac{\Delta t}{V} \sum_f \Phi_f((\rho \vec{U})^o) \\
\frac{\rho^n \vec{U}^n - \rho^o \vec{U}^o}{\Delta t} - \frac{\rho^n \vec{U}^p - \rho^o \vec{U}^o}{\Delta t} &= \frac{1}{V} \sum_f \vec{S}_f \cdot \hat{\Pi}_f^{NS} \\
\Phi_f((\rho \vec{U})^o) &= \Phi_f(\rho^o) \vec{U}_f + p_f \vec{S}_f - \vec{S}_f \cdot \hat{\Pi}_f^{QGD} \\
\hat{\Pi}_f^{QGD} &= \tau_f \vec{U}_f \otimes (\rho_f \vec{U}_f \cdot [\nabla \vec{U}]_f + [\nabla p]_f) + \\
&\quad \tau_f \hat{I}(\vec{U}_f \cdot [\nabla p]_f + \gamma_f p_f [\nabla \cdot \vec{U}]_f) \\
\hat{\Pi}_f^{NS} &= (\mu \nabla \vec{U}^n)_f + \mu_f \left( (\nabla \vec{U}^o)_f^T - \frac{2}{3} \hat{I}(\nabla \cdot \vec{U}^o)_f \right)
\end{aligned}$$

*Energy balance equation*

$$\begin{aligned}
(\rho e)^p &= (\rho e)^o - \frac{\Delta t}{V} \sum_f \Phi_f((\rho e)^o) \\
\frac{\rho^n u^n - \rho^o u^o}{\Delta t} - \frac{\rho^n u^p - \rho^o u^o}{\Delta t} &= \frac{1}{V} \sum_f \left( \frac{\kappa}{C_v} \right)_f \frac{\delta u^n}{\delta \vec{n}_f} |\vec{S}_f| + \\
&\quad \frac{1}{V} \sum_f \hat{\Pi}_f^{NS} \cdot \vec{U}_f^o \\
\Phi_f((\rho e)^o) &= \Phi_f(\rho^o)_f h_f^{tot} - \hat{\Pi}_f^{QGD} \cdot \vec{U}_f \cdot \vec{S}_f - \\
&\quad \tau_f \vec{U}_f \rho_f (\vec{U}_f \cdot [\nabla u]_f + p_f \vec{U}_f \cdot \left[ \nabla \frac{1}{\rho} \right]_f) \cdot \vec{S}_f,
\end{aligned}$$

where superscript  $n$  refers to the new time layer,  $o$  – to the old time layer,  $p$  – to the predicted values.  $V$  is the volume of computational cell,  $\Delta t$  – the time step,  $\vec{S}_f$  – the area of the face between two adjacent cells, multiplied by normal vector value  $\vec{n}_f$ . If not mentioned explicitly, values in expressions are evaluated from previous time layer.

Within this approach, the time step is limited only by the CFL criterion:

$$CFL = (|\vec{U}| + C_s) \frac{\Delta t}{\Delta h} \leq CFL^{max} \quad (11)$$

## CLOSURE RELATIONS FOR $\tau$ -COEFFICIENT

The QGD system, compared with the Navier-Stokes one includes an additional dissipative coefficient  $\tau$ . Due to the of construction of the QGD system, this coefficient must be small, producing a small contribution of the additional terms compared with the other ones.

### Derivation of $\tau$ -coefficient for a rarefied gas

The value of  $\tau$  can be determined using the kinetic derivation of the QGD system or by comparing the additional  $\tau$  terms in the continuity equation (1) with the classical descriptions of self-diffusion, thermodiffusion or barodiffusion effects, see [5], [2]. For a perfect gas, all these approaches lead to

$$\tau = \frac{\mu}{p S c}, \quad (12)$$

where  $Sc$  is the Schmidt number, that is of the order of 1. So  $\tau$  is close to the so-called maxwellian relaxation time  $\tau_{Max} = \mu/p$ , that is close to a mean free time for the gas particles, and  $\tau$  can be estimated as

$$\tau \sim \tau_{Max} \sim \frac{\lambda}{C_s}. \quad (13)$$

Here  $\lambda$  is the mean free path of the gas particles, and  $C_s$  is the sound velocity. A more general formula for  $\tau$  was proposed by Sheretov [5] in the form

$$\tau = \gamma \frac{\mu}{Sc C_s^2 \rho}, \quad (14)$$

### **$\tau$ -coefficient as regularizer of a numerical solution**

For dense gases and liquids,  $\tau$  value is negligibly small, and the role of additional terms in QGD equations becomes negligible compared with that of viscous Navier-Stokes terms. Hence, for computational purpose,  $\tau$  should be increased to make it act as an efficient algorithm regularizer.

One of the natural approaches for the numerical implementation of  $\tau$ -terms consists in replacing the mean free path  $\lambda$  in (13) by the computational space step  $\Delta_h$  in the form

$$\tau = \alpha \frac{\Delta_h}{C_s}, \quad (15)$$

where  $\alpha = const$  is a small numerical factor between 0 and 1 for tuning the computational solution.

Despite of its simplicity, the implementation of QGD equations with  $\tau$  terms in the form (15) allows to use the central differences approximation for all spatial derivatives without stabilizing the algorithm by limiting procedures means of any kind. The Courant stability of the explicit in time central-difference QGD schemes is ensured by the  $\tau$ -terms.

### **Different approaches for the approximation of QGD coefficients**

The value of  $\tau$  may be chosen in a more sophisticated ways, depending on the problem under consideration. Several examples used in previous computations are listed below.

In problems with non-negligible variations of Knudsen numbers, a combination of (12) and (15) was used in the form

$$\tau = \frac{\mu}{p Sc} + \alpha \frac{\Delta_h}{C_s}. \quad (16)$$

For flows with high Reynolds and Mach numbers the  $\tau$  dissipation included in the QGD system can be insufficient to stabilize the solution. In this case additional dissipation can be included in the Navier-Stokes viscous stress tensor to increase the viscosity coefficient as

$$\mu \rightarrow \mu + \Delta^{QGD} p \tau, \quad (17)$$

where  $\Delta^{QGD}$  is a positive tuning coefficient (denoted as  $Sc^{QGD}$  in some works).

For Euler flows, where molecular viscosity and conductivity  $\mu = \kappa = 0$ ,  $\tau$  is calculated using (15). Thus the dissipative coefficients  $\mu = p\tau$  and heat conductivity (9) are regarded as artificial.

Basic values of tuning coefficients are  $\alpha = 0.5$ ,  $\Delta^{QGD} = 1$ ,  $Pr = 1$ . Numerical dissipation and diffusion could be adjusted by decreasing  $\alpha$  and  $\Delta^{QGD}$  coefficients down to values at which the solution becomes unstable or begins to oscillate. The value of  $Pr$  must be kept equal to 1 in most cases.

### **Varying QGD coefficients**

Introducing a dependence of the tuning coefficient from Mach number  $Ma$  in (17) as  $\Delta^{QGD} = \Delta^{QGD}(Ma)$  allows to vary the level of artificial dissipation. For example, it can be increased in the vicinity of shock waves and decreased in the boundary layers.

In this work, the value of additional QGD viscosity coefficient is related to the marker of shocks as follows:

$$\Delta^{QGD} = \frac{|\nabla \rho|}{\rho} \Delta_h \quad (18)$$

For stability reasons, the calculated value is bounded between given minimum  $\Delta_{min}^{QGD}$  and  $\Delta_{max}^{QGD}$  limits.

For wall boundary treatment, standard wall functions from OpenFOAM models were used to replace  $\Delta^{QGD}$  at surfaces where no-slip condition is imposed:

$$\Delta_w^{QGD} = (\nu_t \rho)_w, \quad (19)$$

where  $\nu_t$  is a turbulent viscosity, calculated from wall-function law, subscript “w” stands for “wall”.  $\tau$  is calculated using relation (15).

## VERIFICATION OF THE QGDFOAM SOLVER

The *QGDFoam* solver has been tested for several 1D and 2D cases [1]. Results are briefly presented below. They prove the applicability of the QGD numerical algorithm to solve properly sub-, trans- and supersonic flows in inviscid and viscous regimes. This property makes the QGD numerical algorithm peculiarly valuable for flows where both viscous and high speed effects, such as shocks, simultaneously. *QGDFoam* showed a more diffusive behavior than *rhoCentralFoam* solver with TVD flux limiters, but it is less diffusive than Godunov-type methods, such as HLL.

### 1D verification cases

This section considers the Riemann problems discussed in, e.g., [11] – [12]. They reflect the characteristic features of unsteady gas flows with strong shock waves that are difficult for numerical simulation. The initial data for the Riemann problems are listed in the Table 1 according to the notations used in [12]. Specifically, the flow parameters on the left and right sides of the discontinuity are denoted by subscripts *L* and *R*, respectively. The time at which the plots are shown is denoted by  $t_{fin}$ .

The boundary conditions are the same as the corresponding initial conditions at the ends of the computational domain. In all computations,  $\gamma = 1.4$ , except for the Noh problem (Test 3) with  $\gamma = 5/3$ . The length of the computational domain is equal to 1, from  $x = -0.5$  to  $x = +0.5$ . The discontinuity is placed at  $x = 0$ .

We compare the results obtained by *QGDFoam* and *rhoCentralFoam* solvers. All solutions for *QGDFoam* solver can be obtained with regularization parameter  $\alpha = 0.4$  and numerical coefficient  $\Delta^{QGD} = 1$ . *QGDFoam* and *rhoCentralFoam* solvers use a constant Courant number *Co* and a variable time step.

**TABLE 1.** Initial conditions for Riemann problems

Test	$\rho_L$	$u_L$	$p_L$	$\rho_R$	$u_R$	$p_R$	$t_{fin}$
1	1	0.75	1	0.125	0	0.1	0.2
2	1	-2	0.4	1	2	0.4	0.15
3	1	1	$10^{-6}$	1	-1	$10^{-6}$	1
3a	1	-19.597	1000	1	-19.597	0.01	0.012
4	5.999	19.597	460.894	5.999	-6.196	46.095	0.035
5	1.4	0	1	1	0	1	2
6	1.4	0.1	1	1	0.1	1	2
7	0.126	8.904	782.928	6.591	2.265	3.154	0.0039

A comparison of  $L_1$ -norm of error, computed for Tests 1 – 7 is presented in Table 2.  $L_1$ -norm of error has been calculated on the self-similar solutions for 1D inviscid gas flow equations and a numerical approximation obtained on a uniform mesh from *QGDFoam* or *rhoCentralFoam* solvers, respectively :

$$L_1 = \frac{1}{N} \sum_{i=1}^N |a_i^{exact} - a_i^{num}|, \quad (20)$$

where  $N$  — number of computational points,  $a_i^{exact}$  — exact solution (density, velocity, etc.) at the  $i$ -th computational point,  $a_i^{num}$  — numerical solution at the  $i$ -th computational point calculated using *rhoCentralFoam* or *QGDFoam* solvers. For tests 1, 3, 3a, 5, 6, 7 density  $\rho$  was used to measure  $L_1$ -norm and for tests 2 and 4 — internal energy  $u$ .

The comparison [1] shows that QGD algorithm is competitive with *rhoCentralFoam* method, and is even better for a number of cases. For tests 1 and 2 the QGD Courant number exceeds the *rhoCentralFoam* *Co*. For tests 3a and 7 the *Co* values are equal. For tests 2 and 3 the entropy tail in the QGD formulation is smaller than in *rhoCentralFoam*

**TABLE 2.** Comparison of  $L_1$ -norm computed for Tests 1 – 7 using *rhoCentralFoam* (RCF) and *QGDFoam* and solvers

Test No	1	2	3	3a	4	5	6	7
<i>RCF</i>	0.0024	0.2726	0.0287	0.6935	1.3929	0.0103	0.0103	0.0532
<i>QGD</i>	0.0065	0.2909	0.0368	0.6849	3.6953	0.0021	0.0116	0.0775

one. In test 5 QGD solution coincides with the analytical solution. Nevertheless, additional tuning of the *QGDFoam* and *rhoCentralFoam* settings may bring further improvements for both methods.

## 2D verification cases

The properties of the *QGDFoam* solver were studied in comparison with standard OpenFOAM® solvers in the following cases:

- supersonic inviscid flow over forward-facing step in a channel;
- subsonic viscous laminar flow over backward-facing step in a channel;
- supersonic underexpanded jet flow with Mach reflection;
- supersonic overexpanded jet flow with Mach reflection.

Verification of *QGDFoam* solver in the forward-facing step showed its adequacy in representing inviscid supersonic flows. The method was clearly capable to reproduce the formation of secondary waves reflecting from the upper wall of the channel and upper surface of the step. Behind the rarefaction wave, over the corner of the ledge, the gas density is at its minimum, and near the contact discontinuity, after the triple point over the ledge, the gas density is at its maximum. The results showed that *QGDFoam* density distribution seems less smoothed compared with *rhoCentralFoam* upwind and more smoothed compared with *rhoCentralFoam* TVD 2<sup>nd</sup> order.

Standard OpenFOAM® high-speed flows solvers, based on Godunov-type methods (*rhoCentralFoam*, for example) fail to compute low-speed viscous flow problems. To assess properties of *QGDFoam*, the subsonic laminar flow of a backward-facing step was considered. Results of *QGDFoam* calculations were compared with experiment and another OpenFOAM® solver *simpleFoam* for pure incompressible viscous flows. The comparison [1] shows that for Reynolds numbers up to 300, the results of both methods are in a good agreement with the experiment. For  $Re_h \geq 400$ , the difference increases as the flow becomes three-dimensional, see [2].

The implemented QGD algorithm was able to resolve Mach reflection for the Ladenburg test, which is used often for assessing gas dynamics codes. The algorithm implemented in *QGDFoam* with  $\Delta^{QGD} = 1.0$  is more diffusive than Kurganov-Tadmor scheme with 2<sup>nd</sup> order TVD approximation of convective fluxes, but far less diffusive than HLL scheme (Kurganov-Tadmor (KT) scheme with pure upwind). The *QGDFoam* results can be improved by adjusting artificial viscosity with the  $\Delta^{QGD}$  coefficient. For example, QGD algorithm produces a solution similar to KT with TVD by setting  $S c^{QGD} = 0.15$ .

For the last test, an overexpanded nozzle was simulated with several shock-cells after the exit. Mesh convergence of the center-line time-averaged pressure distribution was compared with the experiment. The QGD solution converges to experimental data (first two shock cells) very well. Moreover, the QGD algorithm resolved accurately the 3rd shock cell which is located in the beginning of the transitional region, where Kelvin-Helmholtz instability waves start to emerge. Further discrepancies between calculation and experiment can be explained by both assumption of flow axisymmetry or insufficient space discretization.

## VALIDATION OF THE *QGDFOAM* SOLVER

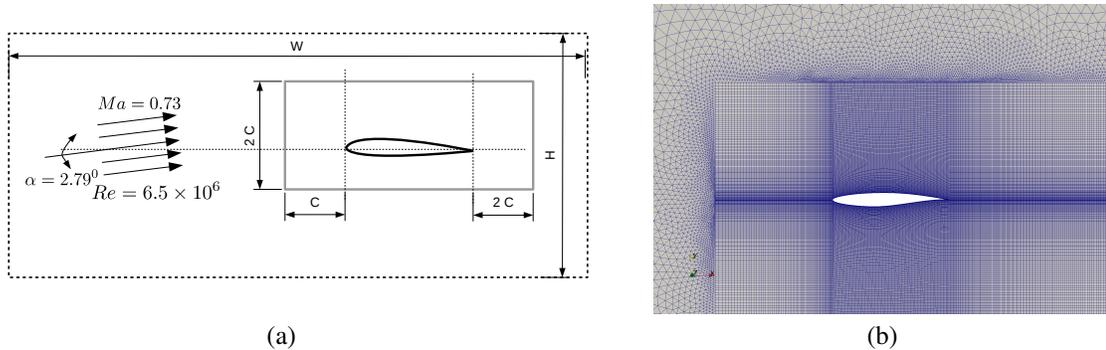
The capability of the developed solver to simulate high Reynolds number compressible flows over blunt bodies is considered. As a first step, the proposed approach for using varying  $\Delta^{QGD}$  coefficient and near-wall functions is verified for the case of high-speed flow over the RAE 2822 airfoil. The parameter settings determined in the former case are used for the industrial-size simulation of a 3D supersonic flow over blunt body.

## Transonic flow over the RAE 2822 airfoil

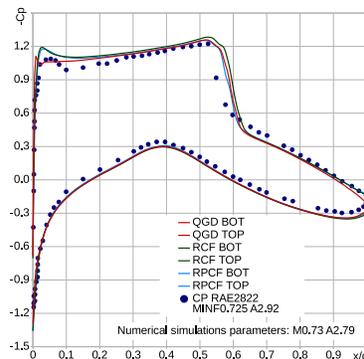
Considering the flow around the RAE 2822 airfoil, the results of the *QGDFoam* simulations have been compared with the experimental data [13] and with simulations of OpenFOAM® solvers *rhoCentralFoam* [9] and *rhoPimpleCentralFoam* [14]. Experimental flow parameters were  $Ma = 0.725$ , angle of attack  $2.92^\circ$  and  $Re = 6.5 \times 10^6$ . According to recommendations from [15], the simulations were carried with  $Ma = 0.73$ , angle of attack of  $2.79^\circ$ . Since the flow is turbulent and unsteady, one of the standard OpenFOAM® Large Eddy Simulation models was chosen to calculate sub-grid scale viscosity, namely the Smagorinsky model [16] for the two-dimensional flow. The turbulent boundary layer is accounted for using standard OpenFOAM® wall functions.

The computational domain is shown in the Fig. 1 and extends from -6 chord lengths on the left-hand side to 7 chord lengths on the right-hand side and from -6 chord lengths at the bottom to 7 chord lengths at the top. The inner region with quadrilateral elements has the size  $4C \times 2C$ , where  $C$  is the chord length of the airfoil. The computational mesh was built with quadrilateral elements in the region near the airfoil. The rest of the domain was meshed with triangular elements. The total number of elements is  $\approx 109k$ .

The gas is assumed to be perfect with an adiabatic exponent of 1.4. Subsonic inflow free stream conditions are imposed at the vertical left and at the horizontal bottom boundaries with a fixed velocity and fixed temperature value. Subsonic outflow free stream conditions were imposed at the vertical right and at the horizontal top boundaries. A no-slip impermeable wall boundary condition is imposed at the surface of the airfoil. The simulation was performed until a physical time such that the flow passed the domain 6 times. To get a steady-state distribution, the pressure field was averaged in time for the window equal to 1 passage of flow through the domain. The comparison between the experimentally measured pressure distribution and the numerically calculated values using *QGDFoam*, *rhoCentralFoam* and *rhoPimpleCentralFoam* are presented in Fig. 2.



**FIGURE 1.** Sketch (a) and grid lines (b) of the computational domain for the case of transonic flow over RAE 2822 airfoil. Here  $C$  is the chord length,  $W = 13C$  is the width of the domain and  $H = 12C$  is the height of the domain.

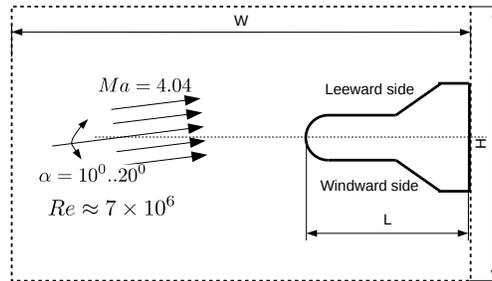


**FIGURE 2.** Comparison of experimentally measured pressure distribution (pressure coefficient) and calculated values using OpenFOAM® solvers *QGDFoam* (QGD), *rhoCentralFoam* (RCF) and *rhoPimpleCentralFoam* (RPCF).

From the pressure distribution it can be seen that for this kind of flow all solvers operate almost identically. However for the top surface the pressure distribution achieved by the *QGDfoam*-solver is slightly better. Discrepancies between the simulation and the experiment at the leading edge of the top surface can be explained by the presence of highly non-orthogonal cells in this region.

### Supersonic flow over a blunt cylinder-flare model

A high-supersonic flow around a hemispherical-nose cylinder with a conical flare at different Mach numbers [17] is studied, see Fig. 3. The gas is assumed to be perfect with an adiabatic exponent of 1.4. The Reynolds number is  $Re = 7 \times 10^6$  with Mach numbers  $Ma = 2.95$  and  $Ma = 4.04$ . To calculate sub-grid scale viscosity the Smagorinsky LES model [16] was used. At body walls a logarithmic wall function was applied. The computational mesh was built using the *snappyHexMesh*-utility and two different cell sizes, 1/192 and 1/384 of the body length, resulting in  $\approx 2.2$  and  $\approx 7.2$  million cells, respectively. The settings for the numerical schemes were set similar to the previous case. Supersonic inflow free stream conditions are imposed at the vertical left and at the horizontal bottom planes with a fixed velocity, fixed pressure and fixed temperature value. No-slip impermeable wall boundary conditions are imposed at the surface of the model. Supersonic outflow free stream conditions were imposed at other external planes of computational domain.



**FIGURE 3.** Sketch of the computational domain for the case of a supersonic flow over a blunt cylinder-flare model. The length  $L$  of the model is 127mm,  $W = 10L$  is the width of the domain and  $H = 12L$  is the height of the domain.

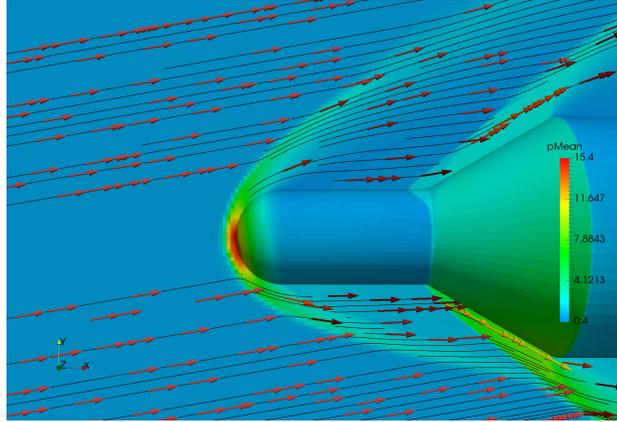
Flow field visualization and distribution of calculated pressure fields on the surface of the object are presented in Fig. 4. A comparison of the simulation results (viscous and inviscid) and the experimental data [17] are presented in Fig. 5 and 6. The zone of interest is located at the leeward side of the body between two incident shocks. The experimental distribution of static pressure clearly shows the presence of the separation zone, where the subsonic turbulent unsteady flow is expected. Inviscid models based on Euler or laminar Navier-Stokes equations are unable to predict such kind of flows and produce a pressure spike near the second incident shock (see Fig. 6). Application of standard wall-functions allows to avoid this spike, however resulting in very steep pressure gradient near the end of separation zone. The same behavior could be observed for the RAE2822 airfoil test case. To resolve this region the grid should be much more refined. It is seen from figures that application of standard wall-functions with QGD algorithm allows to resolve near-surface flow without introducing viscous layer cells.

### SCALABILITY OF THE SOLVER

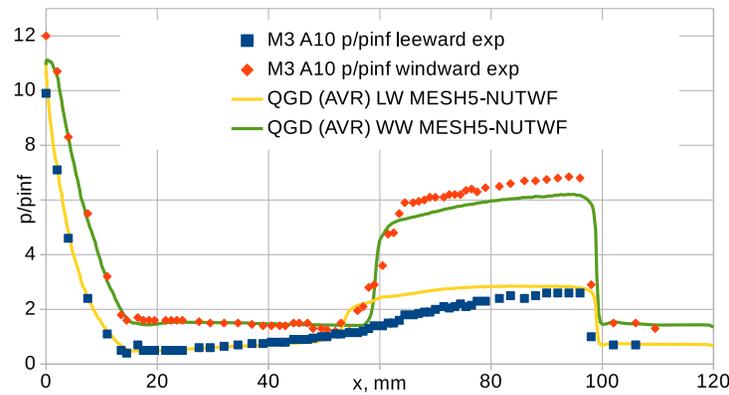
The scalability of the solver has been measured for the NRC "Kurchatov institute" HPC cluster, using the fine grid for the case of supersonic flow over blunt cylinder-flare model. The computational cluster comprises 192 computing nodes with eight Intel(R) Xeon(R) CPU E5345 2.33GHz processors, resulting in a total of 1536 computing cores.

**TABLE 3.** Parallel performance of the developed solver

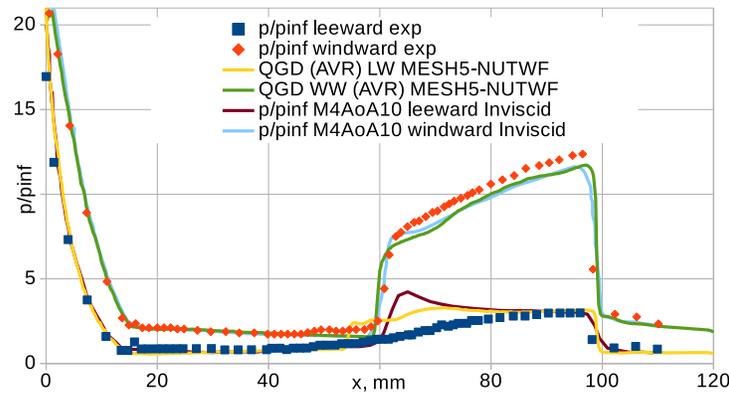
$N_{CPU}$	768	512	256	128	64
CPU time per step,s	0.66	1.05	2.28	2.94	9.67



**FIGURE 4.** Pressure field, stream lines and velocity field distribution around cylinder-flare blunt body model at  $M=4$  and angle of attack  $10^\circ$



**FIGURE 5.** Comparison of experimentally measured pressure distribution (normalized by free-stream pressure) and calculated values using *QGD* solver:  $Ma=3$ , angle of attack  $10^\circ$ .



**FIGURE 6.** Comparison of experimentally measured pressure distribution (normalized by free-stream pressure) and calculated values using the inviscid model [17] and the *QGD* solver:  $Ma=4$ , angle of attack  $10^\circ$ .

It can be seen from Table 3 that performance speedup is almost linear except a strange behavior for the configuration with 128 CPUs. Doubtful results for 128 computing cores have been reproduced several times and

could be explained by hardware issues.

## CONCLUSION

Explicit solver for regularized gas dynamics equations *QGD Foam* has been verified for various 1D and 2D cases. Verification showed that QGD algorithm implementation is able to simulate both viscous and inviscid flows at all speeds. QGD algorithm is more diffusive than TVD schemes with Godunov fluxes, however the solution tends to the analytical one when grid size decreases. Today *QGD Foam* is the only OpenFOAM solver which is able to simulate subsonic, transonic and supersonic flows using a single numerical scheme.

The solver has been tested for 2D & 3D cases with large Re numbers ( $\approx 6 \times 10^6$ ): flow around RAE2822 airfoil at transonic speed and supersonic flow around a blunt flare-body at  $Ma=3$  and  $Ma=4$ .

Wall functions were used to account for near-wall effects in high-speed flows. Effective QGD viscosity coefficient has been calculated using logarithmic wall law. Comparisons of the computed pressure distribution showed acceptable agreement with experimental data.

NRC Kurchatov Institute HPC cluster performance scalability study of the solver showed linear speed-up for number of cores from 128 to 768.

## ACKNOWLEDGMENTS

This study was supported by the Program of Fundamental Research of the Presidium of the Russian Academy of Sciences No. 26

Calculations of this research have been carried out using computing resources of the federal collective usage center Complex for Simulation and Data Processing for Mega-science Facilities at NRC Kurchatov Institute, <http://ckp.nrcki.ru/>

The authors are grateful to J.-C. Lengrand for fruitful discussions and remarks to the paper presentation.

## REFERENCES

- [1] M. V. Kraposhin, E. V. Smirnova, T. G. Elizarova, and M. A. Istomina, *Computers & Fluids* **166**, 163–175 (2018).
- [2] T. G. Elizarova, *Quasi-Gas Dynamic equations* (Springer, 2009).
- [3] S. Strijhak, J. Redondo, and J. Tellez, “Multifractal analysis of a wake for a single wind turbine,” in *Topical Problems of Fluid Mechanics 2017* (Institute of Thermomechanics AS CR v.v.i., 2017).
- [4] K. S. Kuzmina, I. K. Marchevsky, and E. P. Ryatina, “Open source code for 2d incompressible flow simulation by using meshless lagrangian vortex methods,” in *2017 Ivannikov ISPRAS Open Conference (ISPRAS)* (IEEE, 2017).
- [5] Y. V. Sheretov, *Continuum Dynamics under Spatiotemporal Averaging* (SPC “Regular and Chaotic Dynamics”, Moscow-Izhevsk, 2009). (In Russian).
- [6] T. G. Elizarova and Y. V. Sheretov, *J. Computational Mathematics and Mathematical Physics* **41**, 219–234 (2001).
- [7] T. Elizarova, *Computational Mathematics and Mathematical Physics* **51**, 1973–1982 (2011).
- [8] H. Jasak, “Dynamic mesh handling in OpenFOAM,” in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition* (American Institute of Aeronautics and Astronautics, 2009).
- [9] C. J. Greenshields, H. G. Weller, L. Gasparini, and J. M. Reese, *International journal for numerical methods in fluids* **63**, 1–21 (2010).
- [10] M. Kraposhin, A. Bovtrikova, and S. Strijhak, *Procedia Computer Science* **66**, 43–52 (2015).
- [11] T. G. Elizarova and E. V. Shilnikov, *Comput. Math. and Math. Phys.* **49**, 532 – 548 (2009).
- [12] R. Liska and B. Wendroff, *SIAM J. Sci. Comput* **25**, 995 – 1017 (2003).
- [13] P. Cook, M. McDonald, and M. Firman, “Aerofoil RAE 2822 – Pressure Distributions, and Boundary Layer and Wake Measurements. Experimental Data Base for Computer Program Assessment,” Tech. Rep. (J. Barche, ed., AGARD, 1979) AGARD-AR-138.

- [14] M. V. Kraposhin, M. Banholzer, M. Pfitzner, and I. K. Marchevsky, [International Journal for Numerical Methods in Fluids](#) **88**, 79–99 (2018).
- [15] D. Huff, J.-C. Wu, and L. Sankar, “Analysis of viscous transonic flow over airfoil sections,” Tech. Rep. (National Aeronautics and Space Administration, Lewis Research Center, 1987) report No. E-3340.
- [16] J. Smagorinsky, [Monthly Weather Review](#) **91**, p. 99 (1963).
- [17] E. Houtman, W. Bannik, and B. Timmerman, “Experimental and Computational Study of a Blunt Cylinder-Flare Model in High Supersonic Flow,” Tech. Rep. (TU Delft, 1995) report No. LR-796.