Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 220 (2007) 856–878
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An improved gas-kinetic BGK finite-volume method for three-dimensional transonic flow

Georg May *, Balaji Srinivasan, Antony Jameson

Department of Aeronautics and Astronautics, Stanford University, Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035, USA

Received 7 June 2005; received in revised form 5 March 2006; accepted 28 May 2006
Available online 17 July 2006

Abstract

During the past decade gas-kinetic methods based on the BGK simplification of the Boltzmann equation have been employed to compute fluid flow in a finite-difference or finite-volume context. Among the most successful formulations is the finite-volume scheme proposed by Xu [K. Xu, A gas-kinetic BGK scheme for the Navier–Stokes equations and its connection with artificial dissipation and Godunov method, J. Comput. Phys. 171 (48) (2001) 289–335]. In this paper we build on this theoretical framework mainly with the aim to improve the efficiency and convergence of the scheme, and extend the range of application to three-dimensional complex geometries using general unstructured meshes. To that end we propose a modified BGK finite-volume scheme, which significantly reduces the computational cost, and improves the behavior on stretched unstructured meshes. Furthermore, a modified data reconstruction procedure is presented to remove the known problem that the Chapman–Enskog expansion of the BGK equation fixes the Prandtl number at unity. The new Prandtl number correction operates at the level of the partial differential equations and is also significantly cheaper for general formulations than previously published methods. We address the issue of convergence acceleration by applying multigrid techniques to the kinetic discretization. The proposed modifications and convergence acceleration help make large-scale computations feasible at a cost competitive with conventional discretization techniques, while still exploiting the advantages of the gas-kinetic discretization, such as computing full viscous fluxes for finite volume schemes on a simple two-point stencil.
© 2006 Elsevier Inc. All rights reserved.

MSC: 65M99; 82C40; 76P05

Keywords: Navier–Stokes equations; BGK equation; Kinetic scheme; Finite-volume scheme

1. Introduction

Most applications of computational fluid dynamics are based on the solution of the Euler or Navier–Stokes equations. Both can be derived from the Boltzmann equation, which governs the evolution of a single gas

* Corresponding author. Tel.: +1 650 723 9564.
E-mail address: georgmay@stanford.edu (G. May).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.05.027

distribution function describing the entire fluid state. In fact, the Euler and Navier–Stokes equations are only the zeroth and first order approximation in the Chapman–Enskog expansion of the Boltzmann equation.

Because of its high dimensionality and the complicated collision integral the solution of the Boltzmann equation is impractical for all but the simplest problems. Furthermore, the Navier–Stokes equations are sufficiently accurate for very many applications. Nevertheless, the gas-kinetic description of flow physics offers an interesting alternative in the construction of numerical schemes. It seems rational to apply the discretization to the more fundamental quantity, the distribution function, rather than the derived quantities, the primitive or conservative variables. In a finite-volume context numerical fluxes can be computed from the distribution function. If such a distribution function is constructed up to the Navier–Stokes level of accuracy, the discretization allows for the computation of the complete Navier–Stokes fluxes directly as a moment of the distribution function, which is usually obtained on a simple two-point discretization stencil. This means that the terms which give rise to the rate-of-strain tensor and heat flux vector are computed as natural parts of a gas distribution function on the same stencil as convective fluxes, avoiding additional viscous discretization. This facilitates mesh-transparent formulations of numerical schemes considerably, which makes the gas-kinetic route particularly attractive for the implementation on general unstructured meshes.

There is a rich body of literature on kinetic schemes for conservation laws. A brief theoretical introduction can be found in [1]. The benefit of kinetic formulation and representation of conservation laws is not limited to the construction of kinetic schemes, although this is our primary interest here. The kinetic approach may also be used to simplify the theoretical analysis of conservation laws, which, in particular for scalar conservation laws, has led to useful results [2]. An early example for a gas-kinetic scheme for inviscid flow is the equilibrium flux method by Pullin [3]. Deshpande pioneered the use of kinetic flux-vector splitting (KFVS), based on the collisionless Boltzmann equation, for the Euler equations [4]. Chou and Baganoff [5] proposed a KFVS scheme for the Navier–Stokes equations. A scheme based on the BGK simplification of the Boltzmann equation was proposed by Prendergast and Xu [6], which was later analyzed and modified by Xu et al. [7,8]. This scheme has been very successful by introducing the BGK equation as a good compromise between the collisionless Boltzmann equation and the full Boltzmann equation. The BGK equation contains a simple modeling of the molecular collision process, and in fact allows the recovery of the Euler and Navier–Stokes equations by means of Chapman–Enskog expansion, much like the full Boltzmann equation (barring inexact transport coefficients). Furthermore, the BGK equation possesses an analytic solution, which can be directly used for the construction of numerical fluxes. Recently Ohwada has given an analysis of the scheme [9], which established and quantified its consistency and accuracy. We refer to this scheme simply as the BGK scheme.

Gas-kinetic schemes have considerable potential on general unstructured meshes, for which one aims to employ discretization techniques which do not depend on the mesh topology, i.e. are mesh transparent. For conventional schemes this is straight forward only for the convective terms of the Navier–Stokes equations, while the viscous discretization typically depends on the mesh topology and also the choice of control volume in a finite-volume scheme (cell-centered vs. cell-vertex schemes). The BGK scheme allows a stable viscous discretization on a universal next-neighbor stencil regardless of the mesh topology. Our overall formulation of the scheme is designed for general three-dimensional structured and unstructured meshes, and we will present results for different types of meshes in two and three dimensions.

The only obvious disadvantage of the BGK scheme is the fact that it is extremely expensive in terms of floating point operations per flux computation compared to other finite-volume schemes, in particular in a multidimensional formulation [10]. It is probably the main reason why the scheme has not been very often considered for practical applications. We address this issue by proposing a modification of the scheme, which significantly reduces the computational cost, in particular for three-dimensional flow. Furthermore, the issue of convergence acceleration has not been given much attention. We present a multigrid formulation for the BGK scheme, which dramatically accelerates convergence to a steady state and, along with appropriate parallelization, is indispensable for large-scale computations of steady flow. Together with the proposed modifications this makes the BGK route viable for nontrivial three-dimensional computations.

A subject of debate has been the fact that schemes based on the BGK equation will fix the Prandtl number at unity. Different methods for removing this problem have been proposed. We present an alternative to the methods proposed in [8,11] which allows one to set the correct Prandtl number by a very simple modification of the construction of the gradients of the distribution function, reducing the computational cost. In fact the

proposed method requires almost no extra computation (one floating point operation per gradient construction).

The paper is organized as follows. In Section 2 we briefly review some general concepts to illustrate the gas-kinetic route to finite-volume schemes for the Euler and Navier–Stokes equations. In Section 3 we briefly recapitulate the derivation of the BGK scheme, before we present and motivate the aforementioned modifications. The modified reconstruction which allows one to set the correct Prandtl number is outlined in Section 4. After discussing algorithmic details in Section 5, we present basic validation results in Section 6, and results for two- and three-dimensional flow computations in Sections 7 and 8, respectively.

2. Gas-kinetic numerical schemes

In a gas-kinetic description of fluid flow all relevant flow variables are defined as moments of a distribution function. Let ρ be the density, (U, V, W) the velocity vector in Cartesian coordinates, and E the total energy. The conservative variables, given by

$$q = (\rho, \rho U, \rho V, \rho W, E)^{\mathrm{T}}, \tag{1}$$

can be written as

$$q(x, y, z, t) = \int \phi f(x, y, z, u, v, w, t, \xi)\, \mathrm{d}u\, \mathrm{d}v\, \mathrm{d}w\, \mathrm{d}\xi, \tag{2}$$

where f is the distribution function, u, v, w are the phase-space velocities and ξ stands for internal degrees of freedom. In the following we restrict ourselves to monoatomic gases for the sake of simplicity. For polyatomic gases internal degrees of freedom cause only minor differences in the formalism, see the discussion in [12]. For monoatomic gases the vector ϕ can be written

$$\phi = \left(1, u, v, w, \tfrac{1}{2}(\vec{u} \cdot \vec{u})\right)^{\mathrm{T}}, \tag{3}$$

where $\vec{u} = (u, v, w)^{\mathrm{T}}$. The evolution of the distribution function is given by the Boltzmann equation

$$\partial_t f + \vec{u} \cdot \vec{\nabla} f = Q(f, f). \tag{4}$$

Here $Q(f,f)$ is the collision integral. Bhatnagar, Gross, and Krook first introduced the concept of replacing the collision integral with a simple relaxation term, thus arriving at the BGK equation [13]:

$$\partial_t f + \vec{u} \cdot \vec{\nabla} f = -\frac{f - g}{\tau}. \tag{5}$$

Here g is the equilibrium distribution, given by the Maxwellian distribution

$$g(\vec{x}, \vec{u}) = f_0 = \rho(\vec{x}) \left(\frac{\lambda(\vec{x})}{\pi}\right)^{3/2} \mathrm{e}^{-\lambda(\vec{x})((\vec{U}-\vec{u})\cdot(\vec{U}-\vec{u}))}, \tag{6}$$

where

$$\lambda = \frac{\rho}{2p} \tag{7}$$

and p is the pressure. We prefer to use the notation f_0 for the equilibrium distribution to emphasize that this function is in fact the zeroth order term in the Chapman–Enskog expansion of both the Boltzmann and BGK equation. In a rough rationalization the BGK equation can be thought of as modeling the distribution function under the assumption that the flow will be driven to local equilibrium on a time scale τ, the collision time. It can be found by Chapman–Enskog expansion that the Navier–Stokes equations can be recovered from the BGK equation with a viscosity coefficient of

$$\mu = \tau p, \tag{8}$$

which relates the collision time and thus the molecular collision process directly to the macroscopic dynamic viscosity.

We are interested in exploiting the BGK equation in the framework of finite-volume methods. The key idea behind using gas-kinetic methods in this context is to compute fluxes at interfaces between discrete control volumes from reconstructed distribution functions in adjacent cells. Consider the Navier–Stokes equations in integral form for a control volume V_i

$$\int_{V_i} \frac{\mathrm{d}q}{\mathrm{d}t}\,\mathrm{d}V + \int_{\partial V_i} \vec{F}\cdot\mathrm{d}\vec{A} = 0, \tag{9}$$

where $\vec{F}(q)$ is the Navier–Stokes flux vector. If it is assumed that the local direction x and corresponding phase-space velocity u are normal to the face (one can work with rotated coordinates at each face), one obtains upon formally integrating Eq. (5) against ϕ and comparison with Eq. (9):

$$\vec{F}\cdot\mathrm{d}\vec{A} = (F_\rho, F_{\rho u}, F_{\rho v}, F_{\rho w}, F_E)^{\mathrm{T}} = \int u\phi f(\vec{x_l}, \vec{u}, t)\,\mathrm{d}\Xi\,\mathrm{d}A. \tag{10}$$

Here $\Xi = \mathrm{d}u\,\mathrm{d}v\,\mathrm{d}w$ denotes the velocity space and $\vec{x}_l = (x_l, y_l, z_l)^{\mathrm{T}}$ is the face coordinate. In deriving Eq. (10) we have made use of the fact, that the moments with ϕ of the right-hand side of both the Boltzmann and the BGK equation vanish. For this reason the elements of the vector ϕ are sometimes called the collisional invariants.

The flux computed in this manner will be valid provided that the reconstructed distribution function is consistent with the exact distribution function at least to the Euler or Navier–Stokes order of accuracy, depending on which equation is solved. The fundamental task of a gas-kinetic scheme is thus to construct such a distribution function. Numerical fluxes can then be computed according to Eq. (10), and finally Eq. (9) may be integrated in time to advance the numerical solution.

3. The BGK finite volume method

In this section we present a modified formulation of the BGK scheme designed for three-dimensional unstructured meshes. We initially restrict the analysis to one dimension to demonstrate the essential steps and outline the proposed modifications. Subsequently we consider the multidimensional extension.

Ohwada showed [14,9] that in order to approximate the Euler or Navier–Stokes equations via a gas-kinetic formulation one aims to approximate a modified gas-kinetic equation along with initial conditions, obtained by means of Chapman–Enskog expansion, based on the Boltzmann or BGK equation, depending on which equation is chosen as a starting point. If the starting point is the BGK equation, the modified kinetic equation can be approximated by

$$\partial_t f + u\partial_x f = -f_1 \tag{11}$$

with the initial condition given by the first-order Chapman–Enskog distribution

$$f(x,u,0) = f_0(x,u) + \tau f_1(x,u). \tag{12}$$

The Chapman–Enskog state f_0 is given by the Maxwellian distribution, see Eq. (6). The first nonequilibrium state is given by

$$f_1 = -(\partial_t f_0 + u\partial_x f_0). \tag{13}$$

Expanding the analytical solution of this equation for small Δt, one can write to the order of $\tau\Delta t^2$ in truncation error:

$$f(x,u,t) = f_0(x,u,0) - ut\partial_x f_0(x,u,0) + \tau f_1(x,u,0) - tf_1(x,u,0). \tag{14}$$

The truncation error in this simplification is the same as the one incurred in the approximated modified kinetic equation (11). It is this distribution function that we construct at a cell interface between two control volumes, given the estimates of the macroscopic variables to the left and right of an interface.

In the following we assume without loss of generality that any such interface is located at $x_0 = 0$. At the interface, a logical left cell is associated with $x < 0$ and a logical right cell with $x > 0$. We assume that cell-averaged states Q^{l} and Q^{r} and piecewise constant gradients $\partial_x^{\mathrm{l}}, \partial_x^{\mathrm{r}}$ are available in both cells. One can then reconstruct the left and right states $Q^{\mathrm{l}}, Q^{\mathrm{r}}$ at the face as

$$Q^{\mathrm{l}} = Q^{1} + (x_0 - x_1)\partial_x^{\mathrm{l}} q, \quad Q^{\mathrm{r}} = Q^{2} + (x_0 - x_2)\partial_x^{\mathrm{r}} q. \tag{15}$$

In practice the gradients will have to be limited to preserve monotonicity. We will address this in Section 5.4. The reconstruction is schematically depicted in Fig. 1.

The Xu–Prendergast BGK scheme can be obtained by first solving the BGK equation (5) analytically:

$$f(x,u,t) = \frac{1}{\tau}\int_0^t f_0(x',u,s)\mathrm{e}^{-\frac{-(t-s)}{\tau}}\mathrm{d}s + \mathrm{e}^{-\frac{t}{\tau}} f(x-ut,u,0), \tag{16}$$

where $x' = x - u(t-s)$. Eq. (16) represents a nonlinear average between the initial nonequilibrium state $f(x,u,0)$ and an equilibrium state $f_0(x,u,t)$. The weight on the equilibrium state increases with t, which reinforces the interpretation of the BGK equation, as modeling the relaxation to a local equilibrium state on a time scale τ.

3.1. Approximation of the equilibrium state

To approximate the equilibrium state under the integral sign in Eq. (16), a relaxation state $\bar{Q}$ is postulated with a corresponding distribution $\bar{f}_0$ at the face. As $t > 0$, the characteristic line $x_0 - ut$ falls to the left of the interface for $u > 0$, and to its right for $u < 0$. The relaxation state can thus be approximated along the characteristic ut as

$$\bar{f}_0(-ut,u,t) = \bar{f}_0(0,u,0) - ut\partial_x^{\mathrm{l}}\bar{f}_0 H(u) - ut\partial_x^{\mathrm{r}}\bar{f}_0(1-H(u)) + t\partial_t\bar{f}_0, \tag{17}$$

where $H(x)$ is the Heaviside function with $H(x) = 1$ for $x \geqslant 0$ and $H(x) = 0$ otherwise.

The equilibrium distribution function at the face is obtained by using the conservation constraint

$$\int \phi\bar{f}_0\,\mathrm{d}\Xi = \int_{u>0}\phi f_0^{\mathrm{l}}\,\mathrm{d}\Xi + \int_{u<0}\phi f_0^{\mathrm{r}}\,\mathrm{d}\Xi, \tag{18}$$

where f_0^{l} and f_0^{r} represent Maxwellians evaluated with the left and right states, Q^{l} and Q^{r}. The splitting of the phase space is introduced so as to respect the gas-kinetic characteristic lines during reconstruction. It can thus be viewed as gas-kinetic upwinding. The spatial derivatives to the left and right of a face are assumed to be distinct, consistent with a piecewise linear reconstruction in each cell. The derivatives are obtained on either side of the interface by simply differentiating Eq. (6) and using the chain rule, according to

$$\partial_x f_0 = \partial_q f_0 \partial_x q. \tag{19}$$

For the macroscopic state $\bar{Q}$ corresponding to the relaxation function $\bar{f}_0$ no gradients are available from the initial reconstruction. One can use the approximation:

$$\begin{aligned}\partial_x^{\mathrm{l}}\bar{q} &= \frac{\bar{Q} - Q^{\mathrm{l}}}{\|x_0 - x_1\|},\\ \partial_x^{\mathrm{r}}\bar{q} &= \frac{Q^{2} - \bar{Q}}{\|x_0 - x_2\|}.\end{aligned} \tag{20}$$

Notice that the gradients have thus changed from the initial reconstruction to reflect the modeling of the collision process leading to the relaxation state $\bar{f}_0$. Note also that $\bar{f}_0$ is continuous at the face. A more detailed

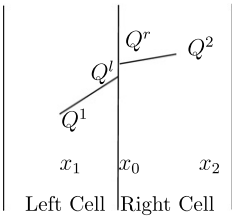

Fig. 1. Linear reconstruction.

description of the computation of the derivatives is given in Appendix A. For notational convenience we denote the spatial reconstruction by

$$\overline{\partial_x f_0} = \partial_x^{\mathrm{l}} \bar{f}_0 H(u) + \partial_x^{\mathrm{r}} \bar{f}_0 (1 - H(u)). \tag{21}$$

We discuss the time derivative of $\bar{f}_0$ in Section 3.3. This distribution function is used for the equilibrium state f_0 in Eq. (16). Given the functional dependence in Eq. (17), the integral in Eq. (16) can be evaluated analytically.

3.2. Approximation of the initial nonequilibrium state

The initial state must be approximated in a way consistent with the first-order Chapman–Enskog expansion, Eq. (12). In the Xu–Prendergast BGK scheme the KFVS discretization is chosen, which uses the information from either side of the face by introducing a kinetic splitting and evaluating the functional dependence along the characteristic line on either side of the face as:

$$f(-ut, u, 0) = \begin{cases} f_0^{\mathrm{l}} - ut\partial_x f_0^{\mathrm{l}} - \tau\{\partial_t f_0^{\mathrm{l}} + u\partial_x f_0^{\mathrm{l}}\}, & u > 0, \\ f_0^{\mathrm{r}} - ut\partial_x f_0^{\mathrm{r}} - \tau\{\partial_t f_0^{\mathrm{r}} + u\partial_x f_0^{\mathrm{r}}\}, & u < 0. \end{cases} \tag{22}$$

One of the solubility constraints of the Chapman–Enskog expansion requires that the higher order terms f_i, where $i > 0$, vanish upon integration over the velocity space against the collisional invariants. Hence the time derivatives are obtained as

$$\int \phi \partial_t f_0^{\mathrm{l/r}} \,\mathrm{d}\Xi = -\int \phi u \partial_x f_0^{\mathrm{l/r}} \,\mathrm{d}\Xi, \tag{23}$$

which is equivalent to enforcing the Euler equations. The spatial derivatives follow from the gradients of the initial reconstruction by chain rule. Note that for the initial state, *both* the state and the derivative are discontinuous. We introduce the notation:

$$\widehat{f_0} = f_0^{\mathrm{l}} H(u) + f_0^{\mathrm{r}} (1 - H(u)), \tag{24}$$

$$\widehat{\partial_x f_0} = \partial_x f_0^{\mathrm{l}} H(u) + \partial_x f_0^{\mathrm{r}} (1 - H(u)). \tag{25}$$

The same notation holds for the time derivatives obtained via Eq. (23).

3.3. The time derivative of the relaxation state

The time derivative in the relaxation state is finally computed using the following constraint:

$$\int_0^{\Delta t} \int \phi (f(0, u, t) - \bar{f}_0(0, u, t)) \,\mathrm{d}\Xi \,\mathrm{d}t = 0. \tag{26}$$

This constraint seeks to mimic the condition that the right-hand side of the BGK equation vanish in integration against the collisional invariants, i.e. $\int \phi(f - f_0)\,\mathrm{d}\Xi$ at all times, by ensuring that on average, over a time step, the condition is satisfied for the full distribution function and the relaxation part $\bar{f}_0$ at the face. Since all the other terms are known, this expression can be solved for the time derivative, which we denote as $\widetilde{\partial_t f_0}$, to give:

$$\int \phi \widetilde{\partial_t f_0} \,\mathrm{d}\Xi = \int \phi (\alpha_1 u \widehat{\partial_x f_0} + \alpha_2 u \overline{\partial_x f_0} + \alpha_3 \{\widehat{\partial_t f_0} + u \widehat{\partial_x f_0}\}) \,\mathrm{d}\Xi. \tag{27}$$

The coefficients α_i can be obtained by straight forward time integration, which shall be omitted here. We have made use of the fact that

$$\int \phi (\bar{f}_0 - \widehat{f_0}) \,\mathrm{d}\Xi = 0 \tag{28}$$

by construction. The time derivative is then computed using the procedure in Appendix A.

3.4. The final form of the distribution function and some comments

Integrating the relaxation state in time, according to Eq. (16), we can write the distribution function as

$$f(0,u,t) = \left(1-\mathrm{e}^{-\frac{t}{\tau}}\right)\{\bar{f}_0 - ut\overline{\partial_x f_0}\} + \mathrm{e}^{-\frac{t}{\tau}}\{\widehat{f_0} - ut\widehat{\partial_x f_0}\} - \tau\left(1-\mathrm{e}^{-\frac{t}{\tau}}\right)\{\widetilde{\partial_t f_0} + u\overline{\partial_x f_0}\} - \tau\mathrm{e}^{-\frac{t}{\tau}}\{\widehat{\partial_t f_0} + u\widehat{\partial_x f_0}\} + t\{\widetilde{\partial_t f_0} + u\overline{\partial_x f_0}\}. \tag{29}$$

One can easily identify the corresponding terms in Eq. (14). The first two terms approximate the equilibrium distribution along the characteristic line ut using a nonlinear average between the continuous and discontinuous reconstruction techniques. The second and third term model the nonequilibrium distribution f_1, which gives rise to the viscous terms using a similar average. Finally the last term approximates the last term in Eq. (14).

It is intriguing to note the nature of the approximation of the terms which form the viscous fluxes, the third and fourth terms in Eq. (29). For moderate Reynolds numbers it is actually the third term which dominates, not the one which comes from the direct approximation of the initial f_1 state, Eq. (22). The spatial derivative in the this term comes from the approximation of the relaxation state along the characteristic line $-ut$, see Eqs. (17) and (21), and the time derivative is the term computed by Eq. (27). One can appreciate how the terms that eventually form the viscous fluxes come together from all parts of the analytical solution of the BGK equation, or the approximation thereto. In the derivation they were not meant to approximate an f_1 state, and the constraint $\int \phi f_1 \,\mathrm{d}\Xi = 0$ will not be satisfied by this approximation.

3.5. The modified BGK scheme

We propose two modifications to the classical BGK scheme which address the construction of the initial nonequilibrium terms (the fourth term in Eq. (29)), and the computation of the time derivative of the relaxation state, i.e. the term $\widetilde{\partial_t f_0}$.

In the BGK scheme the initial state is identical to the distribution function used in the KFVS method [5], which is based on the collisionless Boltzmann equation $\partial_t f + u\partial_x f = 0$, and hence has no relaxation state. It has sometimes been found that the KFVS scheme does not give very good results for viscous flow. The BGK scheme's superior modeling of viscous flow comes from the blending of the initial KFVS state and the continuous relaxation state. In fact, for most Reynolds number regimes the contribution of the KFVS part is negligible in the BGK scheme. This and cost considerations, which will be addressed below, provide the motivation to rethink the computation of the nonequilibrium terms. We start by writing instead of Eq. (22):

$$f(-ut) = \widehat{f_0} - \widehat{\partial_x f_0} - \tau\{\overline{\partial_t f_0} + u\overline{\partial_x f_0}\}. \tag{30}$$

While the first two terms remain unchanged, the averaging $\widehat{\partial_x f_0}$ in the nonequilibrium terms is replaced with the averaging $\overline{\partial_x f_0}$, and the new time derivative $\overline{\partial_t f_0}$ is introduced, which is obtained as

$$\int \phi\overline{\partial_t f_0}\,\mathrm{d}\Xi = -\int \phi u\overline{\partial_x f_0}\,\mathrm{d}\Xi. \tag{31}$$

We point out that the right-hand side of Eq. (31) is also part of the right-hand side in the computation of the time derivative $\widetilde{\partial_t f_0}$, see Eq. (27), where it is blended with other terms. The old formulation computes a complete f_1 state to the left and right of the face separately, which is then discretized by kinetic upwinding when computing fluxes. In the new formulation the upwinding discretization takes place when the f_1 state at the face is computed, by Eq. (31). This makes the viscous terms of the initial state more akin to the third term in Eq. (29), which dominates except when τ is of the order of t or larger. This is the case only for very low Reynolds numbers or whenever τ is artificially increased for the purpose of adding dissipation, e.g. for shock capturing. Validation of the new formulation should reflect this by including low Reynolds number computations and transonic test cases.

In the modified formulation the nonequilibrium terms now cancel by construction when computing $\widetilde{\partial_t f_0}$ according to Eq. (26). One can write

$$\int \phi \widetilde{\partial_t f_0}\,\mathrm{d}\Xi = \int \phi u(\alpha_1 \overline{\partial_x f_0} + \alpha_2 \widehat{\partial_x f_0})\,\mathrm{d}\Xi. \tag{32}$$

Furthermore, the first term on the right-hand side has already been computed for Eq. (31), so that only the two moments resulting from the averaging $\widehat{\partial_x f_0}$ need to be computed. At this point one can write the distribution function as

$$\begin{aligned} f(x,u,t) = {} & \left(1-\mathrm{e}^{-\frac{t}{\tau}}\right)\{\bar{f}_0 - ut\overline{\partial_x f_0}\} + \mathrm{e}^{-\frac{t}{\tau}}\{\widehat{f_0} - ut\widehat{\partial_x f_0}\} - \tau\left(1-\mathrm{e}^{-\frac{t}{\tau}}\right)\{\widetilde{\partial_t f_0} + u\overline{\partial_x f_0}\} \\ & - \tau\mathrm{e}^{-\frac{t}{\tau}}\{\overline{\partial_t f_0} + u\overline{\partial_x f_0}\} + t\{\widetilde{\partial_t f_0} + u\overline{\partial_x f_0}\}. \end{aligned} \tag{33}$$

Although this is only an intermediate step we use this function for validation purposes in order to isolate the effects of the proposed modifications. We shall refer to the scheme based on this distribution function as the BGKg scheme, alluding to the use of the same averaging procedure for the nonequilibrium terms of the initial distribution as for the equilibrium state, which is sometimes denoted as g.

The only terms in which $\widetilde{\partial_t f_0}$ appears, are meant to approximate a nonequilibrium state, i.e. an f_1 state, where the corresponding spatial expansion is given by $\overline{\partial_x f_0}$. Any f_1 state should satisfy $\int \phi f_1\,\mathrm{d}\Xi = 0$. Hence it is not only computationally less expensive, but also appears physically reasonable to use this constraint, which means using Eq. (31), and hence $\widetilde{\partial_t f_0} = \overline{\partial_t f_0}$. We can thus further rewrite the distribution function as

$$f(x,u,t) = \left(1-\mathrm{e}^{-\frac{t}{\tau}}\right)\{\bar{f}_0 - ut\overline{\partial_x f_0}\} + \mathrm{e}^{-\frac{t}{\tau}}\{\widehat{f_0} - ut\widehat{\partial_x f_0}\} - \tau\{\overline{\partial_t f_0} + u\overline{\partial_x f_0}\} + t\{\overline{\partial_t f_0} + u\overline{\partial_x f_0}\}. \tag{34}$$

This formulation will be denoted BGKgg, to reflect the fact that the overbar averaging, Eq. (31), is now used for both the viscous terms, which collapse into one.

Eq. (26), which was originally used to compute $\widetilde{\partial_t f_0}$, was meant to mimic the condition that $\int \phi(f - f_0)\,\mathrm{d}\Xi = 0$, i.e. that the right-hand side of the BGK equation vanish upon integration against ϕ. However, at the level of the partial differential equation, this constraint actually means that the *current* time-dependent equilibrium state f_0 is the only part which survives an integration against ϕ, and all other parts of f vanish. In the new formulation, it has been ensured that all approximated f_1 states vanish individually. The only terms which survive integration against ϕ are the first two terms in Eq. (34), which are indeed the approximation to the most current f_0 state, including the equilibrium portion of the initial state. In the original formulation only the relaxation state $\bar{f}_0$ has been considered the "current" f_0 state.

Incidentally, all formulations are conservative in the usual sense, which is ensured by using a conservative formulation and discretization. More precisely, for any domain Ω, which is subdivided into discrete control volumes the condition $\int_\Omega \mathrm{d}Q/\mathrm{d}t\,\mathrm{d}\Omega = \int_{\partial\Omega} F \cdot \mathrm{d}\partial\Omega$ holds, irrespective of which way to compute the f_1 states or the term $\widetilde{\partial_t f_0}$ is chosen.

To appreciate the reduction in computational cost, consider Table 1. Most of the computational effort lies in computing the moments of terms involving spatial or temporal expansions, for example the integral on the right-hand side in Eq. (31), where two such moments have to be computed, one for each half of the velocity space. We have included here the cost of a full three-dimensional reconstruction, which has not been discussed so far. The one-dimensional formulation outlined here can be used for multidimensional computations as well, although a small error in the viscous terms is incurred. We discuss the multidimensional extension below, and point out how the additional overhead can be avoided without producing any error. The costs of the BGKg and BGKgg schemes thus remain the ones given in Table 1. The simplified formulation thus reduces the computational effort significantly.

Table 1
The number of moments coming from terms involving spatial and temporal expansions, which have to be taken for one flux computation for the standard BGK scheme and the two modified versions, BGKg and BGKgg

Moments for	BGK3D	BGK	BGKg	BGKgg
$\widetilde{\partial_t f_0}$	12	6	2	0
Nonequilibrium terms (initial state)	6	2	2	2
Fluxes	13	7	6	5
Additional terms for 3d	4	–	–	–
Total	35	15	10	7

3.6. Analysis of the multidimensional formulation

The schemes outlined above are applicable to higher dimensions, using Eq. (10) to compute normal fluxes for a finite-volume scheme, provided that at each face the local coordinate x is aligned with the face normal (which suggests the use of a locally rotated coordinate system). However, a slight error in viscous terms is incurred. While the heat flux vector can be accurately represented by normal gradients alone, for the viscous terms in the momentum equation both normal and tangential derivatives are present even in the normal component of the viscous flux. One might consider a more elaborate multidimensional formulation, such as the one proposed recently by Xu et al. [10]. For simplicity we consider only the two-dimensional case. The extension to three dimensions is straight forward. In rotated coordinates with normal and tangential directions x and y, and corresponding macroscopic velocities U and V, the rate-of-strain tensor is given by

$$\sigma = \begin{pmatrix} 2\mu\partial_x U + \lambda(\partial_x U + \partial_y V) & \mu(\partial_x V + \partial_y U) \\ \mu(\partial_x V + \partial_y U) & 2\mu\partial_y V + \lambda(\partial_x U + \partial_y V) \end{pmatrix}. \tag{35}$$

In the rotated frame the face normal is simply $\underline{n} = (1,0)^{\mathrm{T}}$. The viscous fluxes thus depend only on the components in the first column of the rate-of-strain tensor

$$F_{\mathrm{vis,mom}} = \sigma \cdot \underline{n} = \begin{pmatrix} 2\mu\partial_x U + \lambda(\partial_x U + \partial_y V) \\ \mu(\partial_x V + \partial_y U) \end{pmatrix}. \tag{36}$$

We point out here that the second coefficient of viscosity is fixed at $\lambda = -2/5\mu$ by the Chapman–Enskog expansion of the BGK equation. This deviates from the value $-2/3\mu$, which is usually used, but is correct only for monoatomic gases. In practice this difference is of minor importance. The corresponding first nonequilibrium term in the Chapman–Enskog expansion of the gas-distribution function is given by

$$\tau f_1 = \tau\{\partial_t f_0 + u\partial_x f_0 + v\partial_y f_0\}. \tag{37}$$

In the continuous limit, the gas-kinetic formulation is entirely equivalent to the macroscopic one, so that

$$\tau \int u \begin{pmatrix} u \\ v \end{pmatrix} f_1 \, \mathrm{d}\Xi = \sigma \cdot \underline{n} \tag{38}$$

holds exactly. If the tangential derivative $v\partial_y f_0$ is omitted we have

$$\tau \int u \begin{pmatrix} u \\ v \end{pmatrix} \{\partial_t f_0 + u\partial_x f_0\} \, \mathrm{d}\Xi = \begin{pmatrix} 2\mu\partial_x U + \lambda\partial_x U \\ \mu\partial_x V \end{pmatrix}. \tag{39}$$

By comparing to Eq. (36), it is obvious that for the quasi-1D expansion even in the continuous limit the full rate-of-strain tensor is not recovered, and the viscous flux at a given face depends on the local rotation. It is important to note that this holds for any type of mesh. In [10] it was stressed that a multidimensional reconstruction has to be considered for unstructured meshes. However, even for Cartesian meshes a quasi-1D reconstruction does not reproduce the exact rate-of-strain tensor in the continuous limit. It should be pointed out, that unlike conventional schemes, where the difference between Eqs. (39) and (36) is computationally insignificant, the BGK model becomes almost prohibitively expensive, certainly in the three-dimensional case, if the gas-kinetic reconstruction is carried out in all dimensions, see Table 1.

For inviscid flow, a quasi-1D formulation should certainly suffice, since the difficulties involve only the viscous terms. For viscous flow in higher dimensions the difference between the multidimensional reconstruction and the quasi-1D reconstruction is small for a wide class of problems, even on highly stretched unstructured meshes. Accordingly the use of a three-dimensional gas-kinetic reconstruction seems excessive. We propose instead to augment the normal reconstruction by a simple central average of the tangential components of the rate-of strain tensor. The necessary gradients are readily available from the reconstruction, and the stencil for the viscous discretization remains unchanged.

A similar rationale has been used for conventional viscous discretization as well. It is well known that a simple central averaging of the gradients for evaluation of the rate-of-strain tensor leads to odd–even decoupling, and possibly instability. For cell-vertex schemes a discretization technique, which replaces the normal

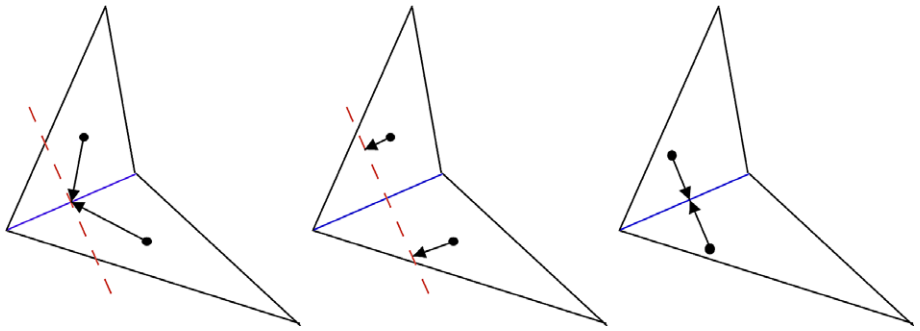

Fig. 2. Schematic illustration of the correction of the normal gradient for stretched meshes in two dimensions. Left: The standard BGK scheme uses Eq. (20). Middle: The modified schemes extrapolate to a projection onto the face normal direction first. Right: Approximate the normal gradient using the new state and the reconstructed state at the face.

component of the central average with a directional difference approximation, has been proposed and used successfully [15]. This correction provides an effective recoupling of the discretization stencil, which is given here by the gas-kinetic normal expansion. Very few additional floating point operations are needed to add the tangential gradients.

For the relaxation state $\bar{Q}$ the classical BGK scheme suggests the computation of the normal gradient by using a difference approximation between the relaxation state and the cell-averaged states, see Eq. (20). This assumes that the face normal is aligned with the direction connecting face centroid and cell centroid, which is true only for Cartesian meshes. For all other meshes the expansion suggested by Eq. (20), gives inconsistent values. This is a serious deficiency of the BGK scheme, since according to Eq. (29) the spatial gradients of the relaxation state are in fact the dominant contribution to the physical viscous terms for most flow regimes. We demonstrate below that this inconsistent expansion can have disastrous effects for stretched unstructured meshes and is in fact of far greater importance than including tangential derivatives. It seems most fitting to remove this problem by extrapolating the cell-averaged values to correspond to a point on the line normal to the face and use the difference between the new state and the relaxation state to compute the gradient. This approach is schematically depicted in Fig. 2. We emphasize that this correction is needed regardless of whether the quasi-1D or a multidimensional formulation is chosen.

4. Transport coefficients

By setting the collision time according to Eq. (8) viscous fluxes with the correct coefficient of viscosity can be computed directly by integrating the distribution function obtained from the BGK equation. However, for the Chapman–Enskog expansion of the BGK equation, the Prandtl number is then fixed at one, so that the heat conduction coefficient will take an incorrect value.

For accurate heat transfer prediction a correction is needed. Chae et al. [11] and Xu [8] suggested modifying the energy flux as

$$F_{\mathrm{E}}^{\mathrm{new}} = F_{\mathrm{E}} + \left(\frac{1}{Pr} - 1\right) q, \tag{40}$$

where the heat flux

$$q = \frac{1}{2} \int (\vec{U} - \vec{u}) \cdot (\vec{U} - \vec{u}) f \, \mathrm{d}\Xi \tag{41}$$

needs to be approximated in some suitable fashion.

In this paper we propose a different method, which corrects the heat flux at the level of the partial differential equations instead. The spatial derivative of the Maxwellian can be obtained by straight forward differentiation of Eq. (6):

$$\frac{\partial_x f_0}{f_0} = \frac{\rho_x}{\rho} - \frac{3}{2}\frac{\partial_x T}{T} + \frac{1}{2RT^2}(u-U)^2\partial_x T + \frac{1}{RT}(u-U)\partial_x U. \tag{42}$$

For simplicity we restrict the following analysis to one dimension. The principle, however, applies to higher dimensions as well.

We define the following modified gradient of the Maxwellian

$$\frac{\partial_x f_0}{f_0} = \frac{\rho_x}{\rho} - \frac{3}{2Pr}\frac{\partial_x T}{T} + \frac{1}{2RT^2 Pr}(u-U)^2\partial_x T + \frac{1}{RT}(u-U)\partial_x U, \tag{43}$$

where the temperature gradient is scaled by the Prandtl number. We will show in the following that this modified gradient in fact leads to a first-order Chapman–Enskog term with the correct Prandtl number.

We have already pointed out that enforcing the constraint

$$\int \phi(\partial_t f_0 + u\partial_x f_0)\,\mathrm{d}\Xi = 0 \tag{44}$$

is equivalent to enforcing the Euler equations, or put differently, computing the time derivatives from the right-hand side of the Euler equations. A straight forward calculation shows that substituting the modified Maxwellian gradient leads, upon taking moments, to the modified Euler equations:

$$\begin{aligned}
&\partial_t \rho = \partial_x(\rho U),\\
&\rho\partial_t U = -\rho U\partial_x U - \frac{\rho R\partial_x T}{Pr} + \partial_x \rho RT,\\
&\frac{3}{2}\rho\partial_t T = -\frac{3}{2}\frac{\rho U}{Pr}\partial_x T - \rho T\partial_x U.
\end{aligned} \tag{45}$$

Substituting these modified gradients in the first-order Chapman–Enskog term gives

$$\tau(\partial_t f_0 + u\partial_x f_0) = -\tau f_0\left(\left[\frac{5}{2}(u-U) - \frac{(u-U)^3}{2RT}\right]\frac{1}{T}\frac{\partial_x T}{Pr} - \frac{2}{3RT}(u-U)^2\partial_x U\right). \tag{46}$$

Here R is the specific gas constant. A similar derivation in three dimensions leads to

$$\tau(\partial_t f_0 + u\partial_x f_0) = \tau f_0\left(\frac{P_{ij}}{p}\frac{1}{2RT}c_i c_j + \frac{q_i}{pT}c_i\left\{1 - \frac{2}{5}\frac{c^2}{2RT}\right\}\right), \tag{47}$$

where $c_i = u_i - U_i$, $P_{ij} = -\mu(\partial_j u_i + \partial_i u_j - 2/3\delta_{ij}\partial_k u_k)$, and $q_i = -\frac{\mu c_p}{Pr}\partial_i T$. Since the heat flux is now proportional to the coefficient of heat conduction $k = \mu c_p/Pr$ one obtains the correct Navier–Stokes terms upon taking the appropriate moments.

As outlined in Appendix A, in practical computations, the spatial derivatives are computed using the relation

$$f_0 + x\partial_x f_0 = f_0(1 + ax), \tag{48}$$

where

$$a = a_0 + a_1 u + a_2 v + a_3 w + \frac{a_4}{2}(u^2 + v^2 + w^2). \tag{49}$$

Xu gives a reconstruction for two-dimensional flows in [8]. The reconstruction for three dimensions is rather similar. After some algebra (see Appendix A) one can see that the coefficients are given by:

$$a_4 = -\partial_x \lambda, \tag{50}$$

$$a_3 = 2\lambda\partial_x W - 2Wa_4, \tag{51}$$

$$a_2 = 2\lambda\partial_x V - 2Va_4, \tag{52}$$

$$a_1 = 2\lambda\partial_x U - 2Ua_4, \tag{53}$$

$$a_0 = -2Ua_1 - 2Va_2 - 2Wa_3 + \frac{\partial_x \rho}{\rho} - \frac{5}{2\lambda}a_4 - (U^2 + V^2 + W^2)a_4. \tag{54}$$

The computation is performed in an efficient recursive fashion, reusing the already computed coefficients. The scaling of the temperature gradient with the inverse Prandtl number can be set by scaling the coefficient a_4, which is related to the temperature gradient via

$$\partial_x \lambda = \partial_x \left(\frac{1}{2RT} \right) = -\frac{1}{2R} \frac{\partial_x T}{T}. \tag{55}$$

Here the gas constant is assumed to be constant, although the method can be used for non-constant R as well. It is sufficient to set $\widetilde{a_4} = a_4/Pr$, and it can be appreciated how the scaling affects the reconstruction through the recursion. Thus, only one additional floating point operation is needed for each (spatial) slope reconstruction. It also appears more straightforward to set the right Prandtl number when constructing the nonequilibrium terms, rather than computing the energy flux at the wrong Prandtl number and substituting the correct heat flux afterwards.

5. Algorithmic details

5.1. Practical implementation of the flux computation

If one uses the method of lines to separate the spatial discretization from the time integration, Eq. (9) may be considered as a starting point for the practical implementation of the finite-volume BGK scheme. The key step, which is at the same time the only place where the method differs from a conventional finite-volume scheme, is the flux evaluation, which is given by Eq. (10). The functional form of the time-dependent distribution function is either of Eqs. (29), (33), or (34). In practical implementations only moments of the distribution function are needed, see Eq. (10), which can be computed by exact integration. Note that, if the spatial and temporal gradients of the reconstructed distribution function are computed as outlined in Appendix A, all moments that need to be computed can be expressed as

$$\int_I u^l v^m w^n f_0 \, \mathrm{d}\Xi \tag{56}$$

for some nonnegative integer numbers l, m, n, and $I = \widetilde{I} \times \mathbb{R} \times \mathbb{R}$. Depending on which integral is computed we have either $\widetilde{I} = \mathbb{R}$, $\widetilde{I} = \mathbb{R}^+$, or $\widetilde{I} = \mathbb{R}^-$, where limits corresponding to the positive and negative real axis result from the kinetic upwinding on the normal phase-space velocity. These types of integrals can be precomputed before each flux computation using efficient recursion formulas [8], and their contributions can then be simply added to evaluate Eq. (10), which is used in Eq. (9).

5.2. The multigrid formulation

For large-scale computations of steady flow multigrid techniques are almost indispensable. A nonlinear multigrid method for nonlinear equations can be outlined as follows. Consider the general nonlinear PDE

$$\mathscr{L}(u) = \mathscr{F}. \tag{57}$$

A numerical estimate of the solution, v_h say, will not satisfy this equation exactly. Assume a correction δv_h can be found such that

$$\mathscr{L}_h(v_h + \delta v_h) = \mathscr{F}_h.$$

Then, after linearization,

$$\mathscr{A}_h \delta v_h + R_h = 0,$$

where $R_h = L_h(v_h) - F_h$ is the residual and A_h is the Jacobian of the nonlinear operator L_h. On the coarse grid the above equation can be replaced by

$$\mathscr{A}_{2h} \delta v_{2h} + I_{2h}^h R_h = 0, \tag{58}$$

where I^h_{2h} represents the aggregation or restriction operator. To avoid using the Jacobian explicitly add

$$\mathscr{L}_{2h}(v_{2h}) - \mathscr{F}_{2h} - R_{2h} = 0$$

to Eq. (58) to get

$$\mathscr{L}_{2h}(v_{2h} + \delta v_{2h}) - \mathscr{F}_{2h} + I^h_{2h} R_h - R_{2h} = 0. \tag{59}$$

This is formally identical to the original equation, except for an additional source term involving the residuals on the coarse and fine mesh. The solution can thus be advanced on the coarse mesh using the same routines as on the fine mesh. Let $v_{2h} + \delta v_{2h} = v^+_{2h}$. After v^+_{2h} has been computed on the coarse grid the corrected solution on the fine grid can be written

$$v^+_h = v_h + I^{2h}_h (v^+_{2h} - v_{2h}),$$

where I^{2h}_h is an interpolation operator.

We combine this multigrid strategy with Runge–Kutta time stepping, as proposed by Jameson [16], which yields a very computationally efficient framework. In principle the same numerical methods that are used on the finest mesh in a multigrid sequence can be used on the coarser meshes as well. However, the numerical schemes used on the coarse meshes do not affect the converged solution, but only the rate of convergence. It has been found, in particular for irregular unstructured meshes, that a more diffusive flux computation on coarse meshes has advantages. In fact, it often pays to smooth the residual correction in Eq. (59) before adding it to the next finer mesh. Keeping these constraints in mind, it seems unreasonable to employ the full gas-kinetic formulation on coarse meshes. Instead we have chosen a simple conventional method for the convective fluxes, using central averages with scalar diffusion according to

$$F_{\text{Riemann}} = \tfrac{1}{2}(F^{\mathrm{L}} + F^{\mathrm{R}}) - \alpha(q^{\mathrm{R}} - q^{\mathrm{L}}). \tag{60}$$

Here F^{L} and F^{R} are the projected inviscid fluxes evaluated with cell-averaged values from the left and right cells, q are the conservative variables, and α is a diffusion coefficient proportional to the spectral radius of the flux Jacobian. A simple central discretization may used for the viscous terms.

5.3. Adaptive diffusion

For viscous flow the collision time is proportional to the dynamical viscosity, according to Eq. (8), i.e. $\tau = \mu/p$. For inviscid flow one can set a lower value corresponding to numerical diffusion, $\tau = \beta_1 \Delta t$, where in practice $\beta_1 \ll 1$, to provide a small amount of background diffusion.

The BGK scheme generates dissipation by virtue of using a discontinuous reconstruction. It has been common practice, however, to augment the collision time by a pressure-dependent term, designed to make the physical viscosity proportional to the mesh width, whenever shocks are present:

$$\tau = \tilde{\mu}/p + \beta_2 \left| \frac{p^{\mathrm{L}} - p^{\mathrm{R}}}{p^{\mathrm{L}} + p^{\mathrm{R}}} \right| \Delta t, \tag{61}$$

where $\tilde{\mu}$ is given by Eq. (8) or by $\tilde{\mu} = \beta_1 \Delta t p$ for viscous and inviscid flow, respectively (the variables can be evaluated with the relaxation state at the face). The rationale behind the additional term is that the numerical viscosity should scale as the characteristic mesh length in underresolved regions. Eq. (61) indeed makes the *physical* viscosity proportional to this characteristic length (provided the time step is proportional to the spectral radius of the flux Jacobian). This has two main effects at shock locations: firstly the physical viscous terms are proportional to the increased coefficient of viscosity; secondly the discontinuous reconstruction is more heavily weighted.

If the additional increase of the physical viscosity is necessary or sufficient for adequate shock capturing is not at all obvious. For some transonic flow cases, in particular in three dimensions, it was not used in our implementations, while for most two-dimensional computations we generally found it necessary to augment the inherent dissipation generated by the BGK scheme by increasing τ. This simply suggests that the best treatment of dissipation is still an open problem for this class of scheme.

5.4. Limiting

To promote monotonicity the gradient-based data reconstruction has to be limited using a limiter function ϕ:

$$q(x) = Q_i + \phi(q)\nabla q \cdot (x - x_i). \tag{62}$$

For unstructured meshes the limiter due to Venkatakrishnan [17] has proved very successful, and has been used for this work. For one-dimensional results and results on structured meshes, Van Leer's limiter [18] has been used.

The nature of the gas-kinetic reconstruction warrants a more careful look at the limited reconstruction. In the standard formulation *all* gradients in the construction of the distribution functions are limited. This, however, means that the nonequilibrium terms, which give rise to viscous terms, will in fact not be evaluated with the correct gradients. This sets the gas-kinetic scheme apart from conventional approaches where usually only the reconstruction for the convective terms is limited, whereas the viscous terms are evaluated with exact gradients. For the BGKgg scheme an option to not use data limiting for the nonequilibrium terms can be easily included. The moments for the expansion $\widetilde{\partial_x f_0}$ have to be computed twice in this case, once with and once without limiting, which increases the total number of moments from seven to eight, see Table 1, which is still significantly cheaper than the standard BGK scheme. Due to the intricate nature of the gas-kinetic reconstruction in the standard BGK scheme, in particular the expansion $\widetilde{\partial_t f_0}$, it is extremely cumbersome and expensive to provide for such an option. For cases in which viscous phenomena are resolved properly, the BGK scheme thus may suffer from degraded accuracy. We provide examples showing how the limiting can affect the solution for the BGK scheme in Section 7.

6. Numerical validation

6.1. Validation of the Prandtl number correction

Several test cases have been computed to validate the Prandtl number correction. Fig. 3 shows the velocity and heat flux distribution across a resolved normal shock for two different Prandtl numbers computed with the BGKg scheme. The upstream Mach number is $M = 10$. We compare the results with the analytical solution of the one-dimensional Navier–Stokes equations [8]. Excellent agreement can be observed. We note that the Navier–Stokes equations are not well suited for such high degrees of nonequilibrium, so the results shown here cannot be expected to represent accurate physics. However, the aim of these computations is merely to demonstrate that the gas-kinetic scheme solves the Navier–Stokes equations at the specified Prandtl number.

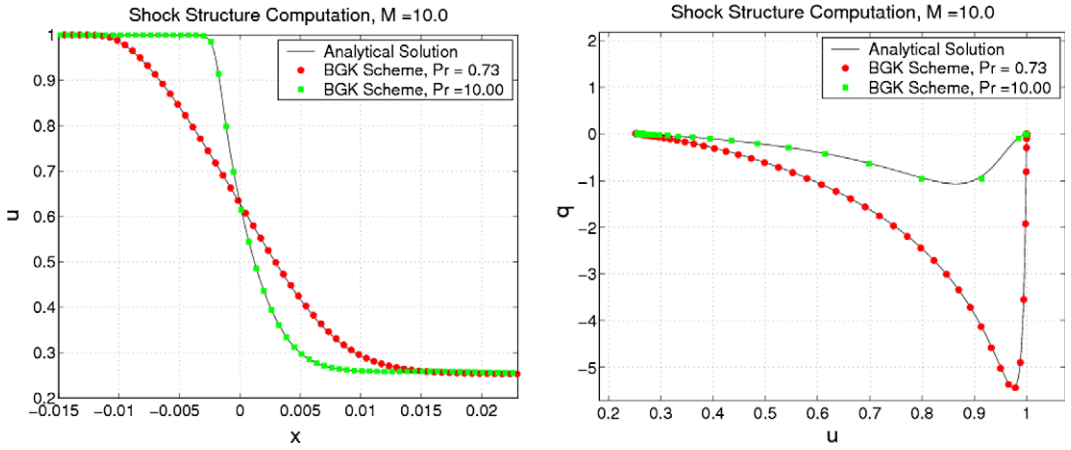

Fig. 3. Velocity (left) and heat flux (right) across a strong normal shock ($M = 10$). Note that only every other point has been plotted.

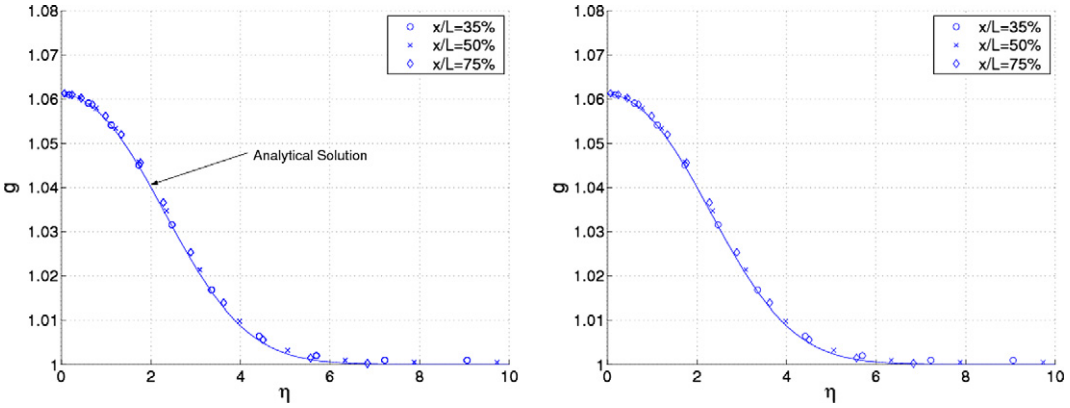

Fig. 4. Dimensionless temperature for compressible boundary layer at $M = 0.6$, $Pr = 0.72$. Standard BGK scheme (left) and BGKg scheme, Eq. (33).

Fig. 4 shows the similarity solution for a zero-pressure-gradient compressible boundary layer on a flat plate at $M = 0.6$ and $Pr = 0.72$ in terms of the dimensionless temperature T/T_0, where T_0 is the freestream temperature. The profiles are plotted in similarity coordinates and compared to the exact similarity solution at several stations along the plate. We observe very good agreement for both the standard BGK scheme and the BGKg scheme, according to Eq. (33), which are both shown in the figure. For this well-resolved case the differences between the schemes are negligible. This is also true, although not shown, for the BGKgg scheme.

6.2. Isentropic nozzle flow: a convergence study

As a preliminary test case, and for the purpose of carrying out a convergence study, consider isentropic, quasi one-dimensional nozzle flow. For this case the governing equations can be written

$$\partial_t Q + \partial_x F + S = 0, \tag{63}$$

where

$$Q = \begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix}, \quad F = \begin{pmatrix} \rho U \\ \rho U^2 + p \\ U(E+p) \end{pmatrix}, \quad S = \frac{1}{A}\frac{\mathrm{d}A}{\mathrm{d}x}\begin{pmatrix} \rho U \\ \rho U^2 \\ U(E+p) \end{pmatrix} \tag{64}$$

and the dependent variables are defined as in Section 2. The cross-section of the nozzle, A, is defined on the interval $[0, x_{\text{out}}]$, where we have taken $x_{\text{out}} = 10$, and

$$A(x) = \begin{cases} d_{\text{in}} + \xi(x)^2(3 - 2\xi(x))(d_{\text{t}} - d_{\text{in}}), & x < x_{\text{t}}, \\ d_{\text{t}} + \eta(x)^2(3 - 2\eta(x))(d_{\text{out}} - d_{\text{t}}), & x \geqslant x_{\text{t}}, \end{cases} \tag{65}$$

where $\xi(x) = x/x_{\text{t}}$ and $\eta(x) = (x - x_{\text{t}})/(x_{\text{out}} - x_{\text{t}})$. The coordinate $x_{\text{t}} = 3.75$ is the location of the nozzle throat, while the nozzle diameters at the inlet, outlet and throat are given by $d_{\text{in}} = 1$, $d_{\text{out}} = 1.25$, and $d_{\text{t}} = 0.875$, respectively.

We choose a nozzle exit Mach number of $M = 0.3$, which leads to a maximum Mach number in the nozzle throat of $M \approx 0.46$ for the chosen nozzle geometry. At the inflow we extrapolate the outgoing Riemann invariant and fix the total enthalpy and entropy function $s = \rho^\gamma/p$, while at the outflow the entropy and total enthalpy are extrapolated and the pressure is fixed. The exact solution for the Mach number can be obtained from the well known area-Mach number relation for isentropic nozzle flow, which can be found in many standard text books, e.g. [19]. Table 2 displays the maximum error in the Mach number for the BGKgg scheme in mesh refinement, where we have denoted the number of mesh cells with N. We compare the data to a conventional finite volume scheme, which uses a central average for the fluxes, augmented by a CUSP construction of artificial diffusion [20]. It can be seen that the BGKgg scheme reaches second order of accuracy, once the

Table 2
Comparison of the maximum error in the Mach number between the BGKgg scheme and the CUSP scheme for isentropic nozzle flow

N	BGKgg		CUSP	
	L_∞(error)	Order	L_∞(error)	Order
20	5.920739e − 04		7.531804e − 04	
30	3.093076e − 04	1.601359	4.037851e − 04	1.537548
40	1.740776e − 04	1.998161	2.375062e − 04	1.844706
50	1.044453e − 04	2.289281	1.513558e − 04	2.019152
70	5.049497e − 05	2.160028	8.068882e − 05	1.869495
100	2.409334e − 05	2.074545	3.744039e − 05	2.152801

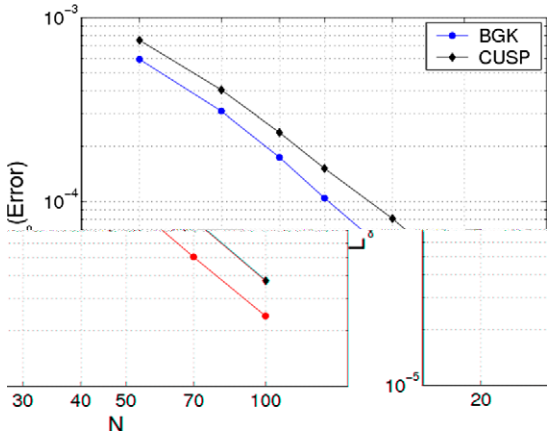

Fig. 5. Comparison of the maximum error in the Mach number between the BGKgg scheme and the CUSP scheme for isentropic nozzle flow.

asymptotic region is reached. A graphical comparison between the two schemes is shown in Fig. 5, where the error is plotted on a log scale against the number of cells. Here it is clearly visible that both schemes converge at approximately the same rate, while the error levels for the BGKgg scheme are somewhat lower, compared to the conventional CUSP scheme.

7. Two-dimensional results

In this section we focus on basic validation of the modified versions of the BGK scheme for inviscid and viscous flow. While multigrid was implemented and used, we defer the discussion related to multigrid to Section 8, where we present three-dimensional results on unstructured meshes.

The computations in this section use triangular meshes in cell-centered discretization, i.e. the primary mesh elements are used as control volumes. On numerous occasions we compare the BGK scheme to results obtained with a conventional method, for which we have chosen a finite-volume scheme using a central average with CUSP construction of artificial dissipation [20] for the convective terms, and a central discretization for the viscous terms. More precisely, in the conventional implementation the viscous terms are computed by first transferring the relevant variables from the cell centroids to the nodes, computing primitive gradients there using a least-squares method, and then averaging the gradients on edges (i.e. the faces of the triangular control volumes) to compute the rate-of-strain tensor. The reconstruction is identical for both schemes and uses a least-squares estimate of the gradient along with Venkatakrishnan's limiter. In fact, aside from the flux computation routines, there is no difference between the code that employs the conventional method and the gas-kinetic version.

We first present two-dimensional results for inviscid transonic flow. Fig. 6 shows the computed pressure contours for a NACA0012 profile in transonic flow at $M = 0.8$ and an incidence of $\alpha = 1.25°$. The CUSP solu-

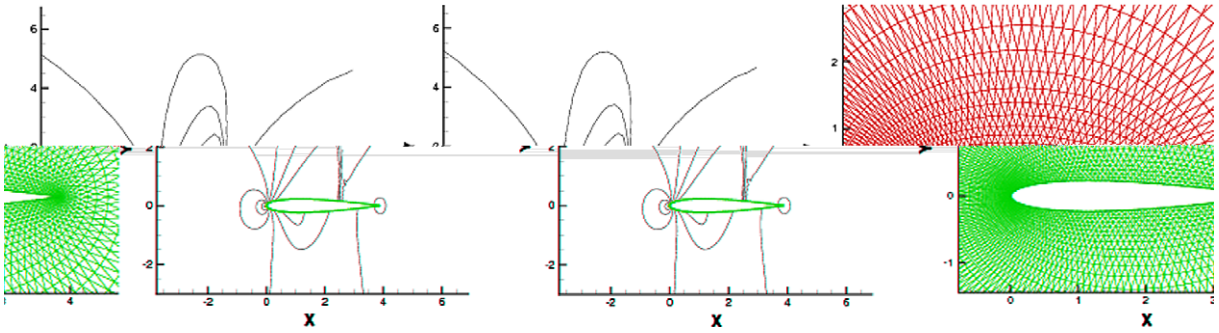

Fig. 6. Inviscid transonic flow around the NACA0012 airfoil at $M = 0.8$, $\alpha = 1.25°$. Contour plots of pressure for the CUSP scheme (left) and the BGKgg scheme (middle). The mesh is shown on the right.

tion is compared to the BGKgg scheme (incidentally, for inviscid flow the initial nonequilibrium state can be omitted, and hence the BGK and BGKg schemes are identical). The shock capturing capabilities are virtually identical for all schemes. The computation was carried out on a mesh with 10,240 triangles. Fig. 7 compares the pressure distributions along the airfoil for the same test case. We have used $\beta_1 = 10^{-2}$ and $\beta_2 = 1$ in Eq. (61).

For two-dimensional viscous validation we first demonstrate the necessity of using a consistent approximation for the normal gradient of the relaxation state $\bar{f}_0$, i.e. applying the correction of the normal derivative described in Section 3.6. Fig. 8 shows the streamwise velocity of a zero-pressure gradient boundary layer at $Re = 2000$, $M = 0.2$ computed on a stretched triangular mesh using the standard BGK scheme. The figures overplot the solution sampled at every 10% length of the plate in similarity coordinates. It can be seen that using a consistent approximation of the gradient of the relaxation state is of far greater importance than using the multidimensional reconstruction. Computing viscous fluxes with incorrect projected gradients is more detrimental than omitting tangential expansions, and the correction must be carried out for both one-dimensional and multidimensional reconstruction. The same is true for the BGKg and BGKgg schemes, which use the same gradients to approximate the viscous terms.

For two-dimensional viscous validation of the modified scheme we focus on low Reynolds number computations to amplify the contribution of the viscous terms, along with transonic flow conditions, to lend some importance to the generation of artificial dissipation as well. Some of the test cases we present have also been considered in [21], and have been previously used as benchmarks for viscous flow [22]. Fig. 9 shows the dimensionless pressure distribution p/p_∞, where p_∞ is the freestream pressure, for the NACA0012 profile at $M = 0.8$ and $\alpha = 10°$ for Reynolds numbers $Re = 73$ and $Re = 500$. One can see that although the contribution of the

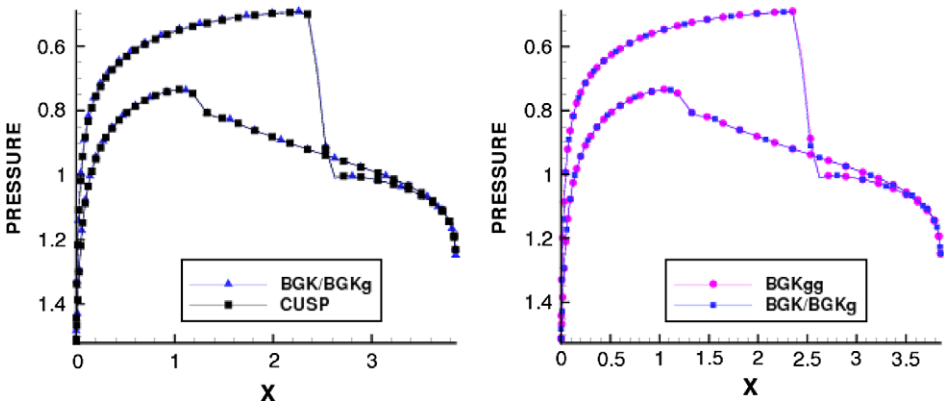

Fig. 7. Inviscid transonic flow around the NACA0012 airfoil at $M = 0.8$, $\alpha = 1.25°$. Comparison of the dimensionless pressure p/p_∞ along the airfoil.

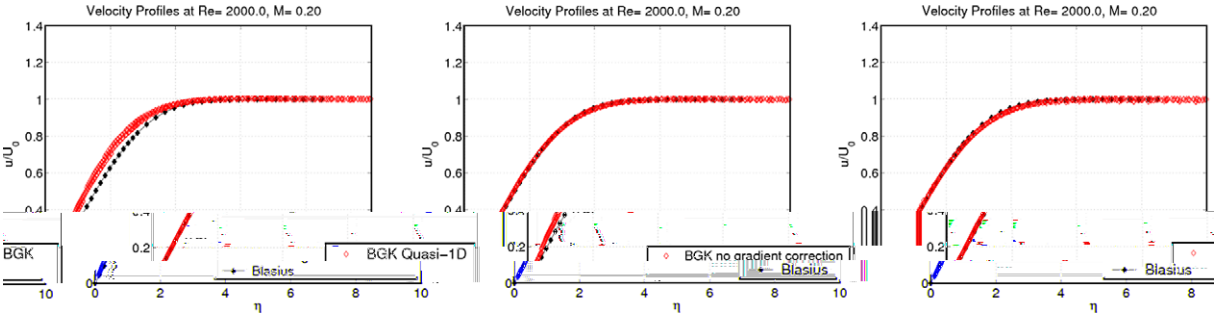

Fig. 8. Streamwise velocity for zero-pressure-gradient boundary layer on a stretched triangular mesh. Left: Multidimensional reconstruction and Eq. (20) without correction of the cell-averaged state. Middle: Multidimensional reconstruction with gradient correction. Right: Quasi-1D reconstruction with gradient correction.

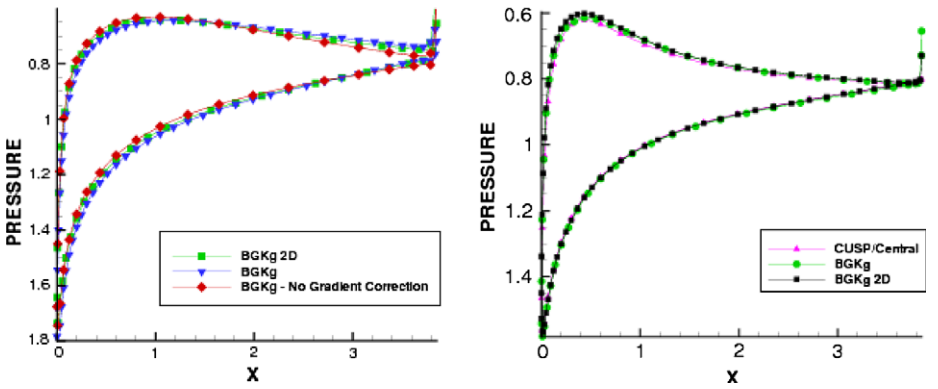

Fig. 9. Left: NACA0012 profile at $Re = 73$, $M = 0.8$, and $\alpha = 10°$. Comparison of quasi-1D and 2D reconstruction, and quasi-1D using Eq. (20) without correction of the cell-averaged state. Right: NACA0012 profile at $Re = 500$, $M = 0.8$, and $\alpha = 10°$. Comparison of quasi-1D and 2D reconstruction.

viscous terms is very significant for these test cases the difference between the quasi-1D and 2D reconstruction is small. Tangential gradients have been omitted. We observe that using the correct normal expansion for $\overline{\partial_x f_0}$ is of greater importance than including tangential components.

Fig. 10 demonstrates the influence of limiters on the solution for $Re = 500$. A contour plot of the Mach number distribution is shown, computed with the BGKg scheme, along with a comparison of the dimension-

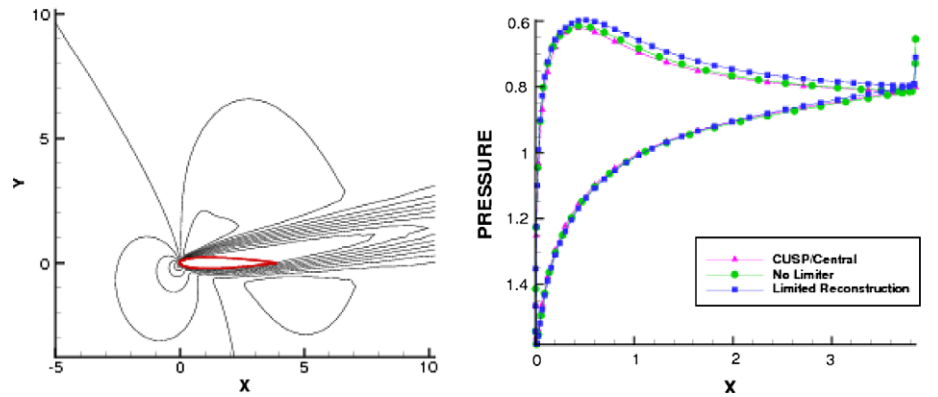

Fig. 10. NACA0012 profile at $Re = 500$, $M = 0.8$, and $\alpha = 10°$. Mach number contours computed with the modified BGK scheme without limiters (left), and comparison of the pressure distribution along the airfoil for the CUSP scheme with central viscous discretization and the BGKg scheme with and without limiter (right).

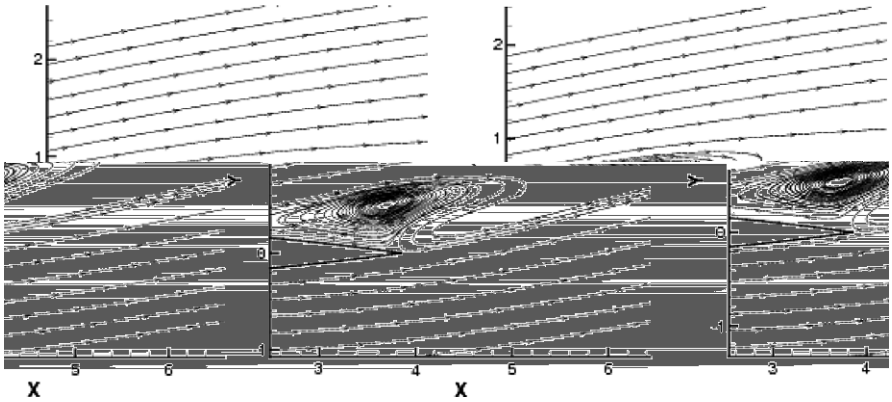

Fig. 11. NACA0012 profile at $Re = 500$, $M = 0.8$, and $\alpha = 10°$. Separation on the suction side. The BGKg scheme (left) and the CUSP scheme with central viscous discretization (right).

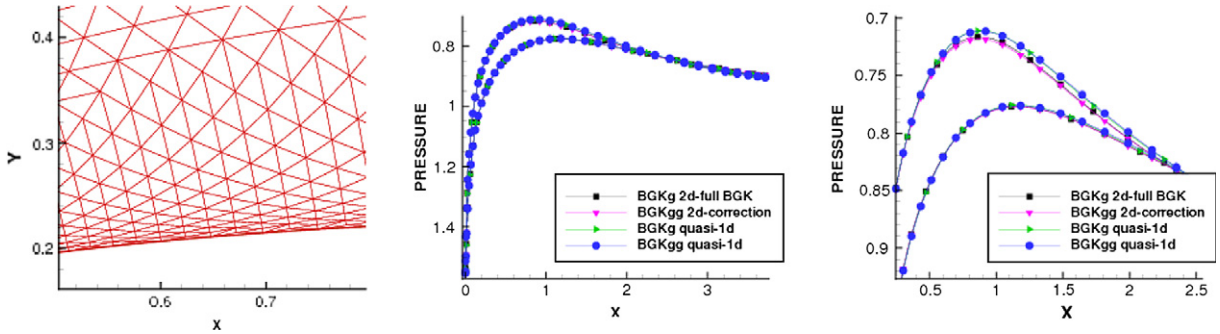

Fig. 12. NACA0012 profile at $Re = 1500$, $M = 0.8$, and $\alpha = 1.25°$. Left: Magnified view of the mesh inside the boundary layer. Middle: Pressure distribution along the profile for various reconstruction methods. Right: Magnified view of the pressure distribution on the suction side to amplify the difference between the quasi-1d and 2d reconstruction techniques.

less pressure distribution between the CUSP scheme (with central discretization of viscous terms), and BGKg scheme with and without limiters. The influence of data limiting on the BGK scheme has already been pointed out by Xu [8], and can be clearly seen in the pressure distribution (although not shown the same behavior is observed for the BGKgg scheme). We point out here that the CUSP solution was computed using limiters as well, but due to the fact that the viscous terms use unlimited gradients, the solution is not greatly affected. As outlined in Section 5.4, an advantage of the BGKgg scheme is that at least the option of not using data limiting for the nonequilibrium terms can be easily included at only slightly increased computational cost.

Fig. 11 shows a close-up view of a large separation zone, which is present for these flow conditions on the suction side of the airfoil. The size and shape of the recirculation region is very similar for the all schemes. We show exemplarily the CUSP and BGKg solutions.

Fig. 12 shows the solution for the NACA0012 profile at $M = 0.8$, $Re = 1500$, and incidence $\alpha = 1.25°$. To resolve the boundary layer, the mesh becomes increasingly stretched. However, the difference between the quasi-1d and the 2d reconstruction methods remains small. In the magnified view of the pressure distribution on the suction side it can be seen that the 2d construction outlined in Section 3.6 and the full gas-kinetic 2d reconstruction yield virtually identical results, for both the BGKg and the BGKgg scheme.

8. Three-dimensional results

We focus mainly on the multigrid implementation on unstructured meshes in this section. The BGK method has been embedded into the Flo3xx computational aerodynamics architecture, documented in a pre-

vious publication [23], which solves the three-dimensional Euler and Navier–Stokes equations on arbitrary meshes in finite-volume formulation.

The Onera M6 test case at $M = 0.84$, $\alpha = 3.06°$ has been chosen to validate three-dimensional inviscid transonic flow using the BGKg method (which is identical to the standard BGK scheme for inviscid flow) and the BGKgg scheme with multigrid. Fig. 13 shows the convergence history of the lift and the density residual on a tetrahedral mesh with 94,000 nodes. Also shown is a comparison of the convergence history for the lift coefficient between the BGK(g) scheme and the CUSP scheme, which serves to show that the BGK scheme is certainly competitive with conventional schemes in terms of convergence to a steady state. We have used $\beta_1 = 10^{-2}$ and, in fact, $\beta_2 = 0$ for these computations. This means that no artificial dissipation is added, and we rely solely on the dissipation generated by the gas-kinetic upwinding. The shock capturing behavior is excellent, as is shown in Fig. 14, where the pressure distribution in terms of the pressure coefficient is shown for the wing section $y/b = 0.2$ taken from computations on tetrahedral meshes. Both the CUSP method and the BGK schemes are shown and can be seen to give very similar results. Both show a good shock resolution. In particular the complete absence of oscillatory behavior, especially for the BGK scheme should be noted. The shock resolution on a mesh of moderate size, such as the 316,000 node mesh is excellent.

Fig. 15 shows the convergence history of a single-grid and a three-level multigrid solution for a computation of a Falcon Business Jet configuration in transonic inviscid flow, along with contour plots of the density. The multigrid overhead has been taken into account in the work units used to compare the convergence. The speedup due to multigrid is most dramatic in integrated quantities, such as the coefficient of lift, shown in Fig. 15. We point out that solutions for the BGKg scheme and the BGKgg scheme are virtually identical.

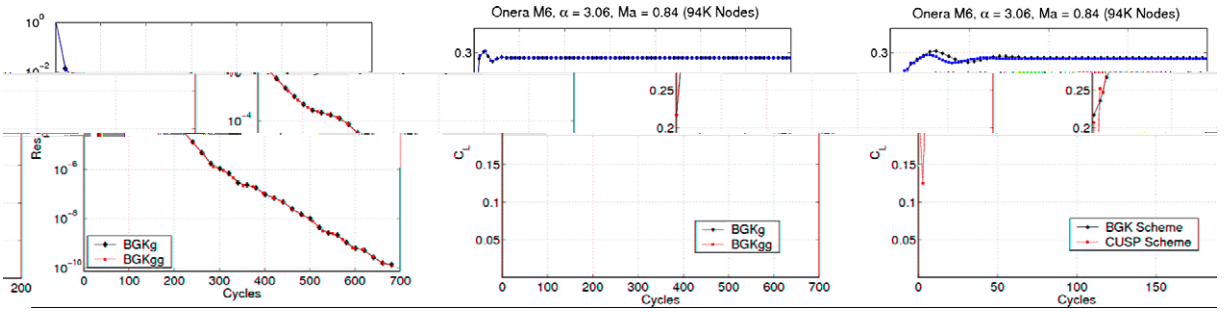

Fig. 13. Left and middle: Convergence history of the density residual and lift coefficient for the BGKg and BGKgg scheme. Right: Comparison of the convergence history for the CUSP and BGK(g) scheme on a magnified scale. Tetrahedral mesh using a three-level multigrid sequence. Coarse meshes use conventional first-order fluxes.

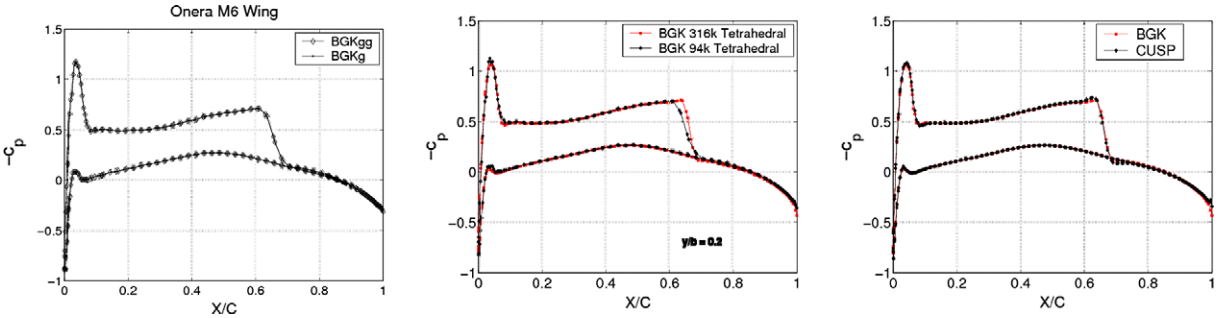

Fig. 14. Pressure coefficient for the Onera M6 Wing at $M = 0.84$, $\alpha = 3.06$ at wing section $y/b = 0.2$. Left: Comparison of the BGKg and BGKgg scheme on 94,000 node mesh. Middle: BGKg scheme on different meshes (94,000 and 316,000 nodes). Right: BGKg and CUSP scheme on the 316,000 node mesh.

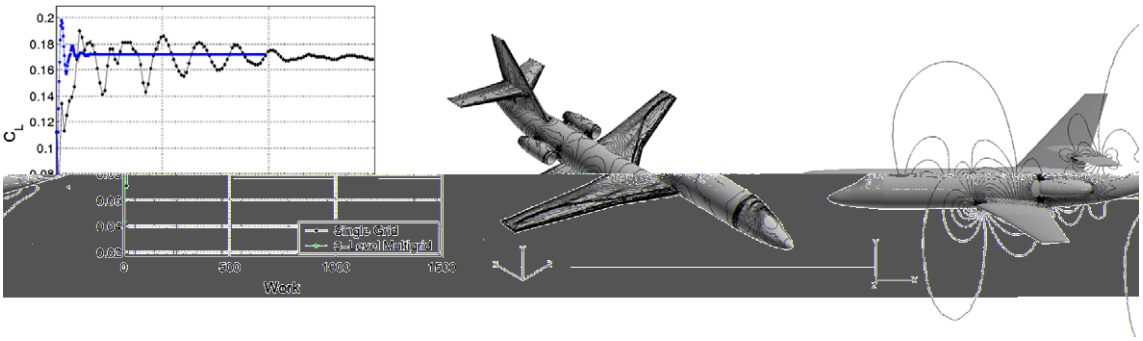

Fig. 15. Falcon Business Jet at $M = 0.8$ and $\alpha = 2°$. Left: Convergence history of the coefficient of lift for single grid and multigrid computations. Middle and right: Density contours.

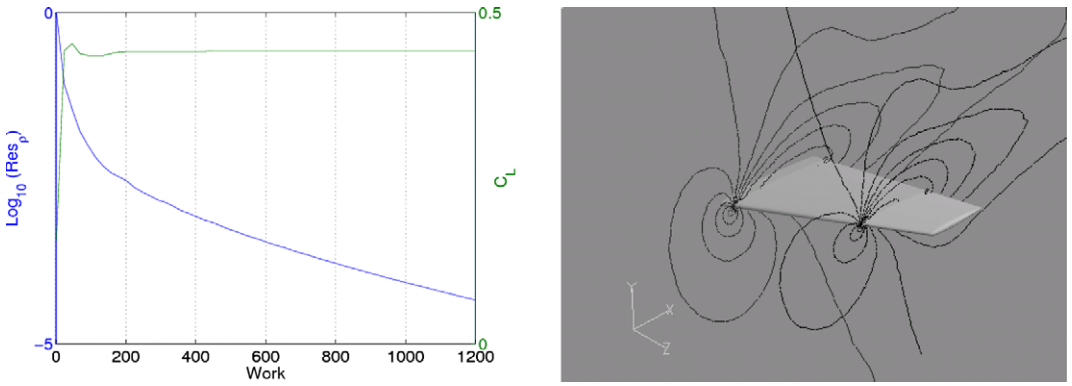

Fig. 16. Onera M6 Wing at $Re = 73$, $M = 0.8$, and $\alpha = 3.06°$. Left: Convergence history of density residual and coefficient of lift. Right: Contours of pressure coefficient.

All schemes are stable at CFL = 10 with an explicit 5 Stage Runge–Kutta time stepping technique with residual averaging and local time stepping.

Basic viscous validation for a low Reynolds number is shown in Fig. 16, where the solution for the Onera M6 Wing at $M = 0.8$, $\alpha = 3.06°$ and $Re = 73$ is shown. In terms of the Mach number and angle of attack this is a standard test case, however computed at much lower Reynolds number to emphasize the contribution of the viscous terms. Note that the convergence history has been plotted against work units, which introduces a factor of about 1.4 in increase of cost compared to single grid solutions, made up in equal parts by the computations on coarser meshes and the cost of performing one extra residual evaluation on the fine mesh for the coarse mesh forcing function in Eq. (59). Note how the coefficient of lift is essentially converged when the residual has been reduced by about three orders of magnitude. The mechanism of convergence acceleration is an effective increase of the speed with which information is propagated in the flow field. Typically, solutions in multigrid computations do not tend to creep, and the steady state is reached at much higher levels of residuals compared to single grid computations.

9. Conclusion

Modifications to the standard BGK scheme have been proposed and examined. All schemes have been shown to give good results overall. There is very little difference between the different flavors of the method in terms of results, but enormous differences in CPU time. Large-scale computations in three dimensions are made feasible by the modified formulations (along with multigrid), with the consequence that the otherwise prohibitive cost of gas-kinetic computations is drastically reduced. The BGKgg method is the least com-

putationally expensive of the tested schemes, and should therefore be preferred. In particular when compared to the full three-dimensional BGK scheme, the savings in CPU time are dramatic.

Appendix A. Reconstruction

In this section we give explicit formulas for the reconstruction procedures needed for the spatial gradients and time derivatives. Very similar formulas have been given by Xu for the two-dimensional case [8]. We include the three-dimensional formulas merely for convenience. The spatial gradients of the Maxwellian distribution can be evaluated by direct differentiation. We write

$$f_0 + x\partial_x f_0 = f_0(1 + ax), \tag{A.1}$$

where

$$a = \frac{1}{f_0}\frac{\partial f_0}{\partial x} = \frac{\partial(\ln f_0)}{\partial x}. \tag{A.2}$$

This leads to

$$\begin{aligned}\frac{\partial(\ln f_0)}{\partial x} &= \frac{\partial}{\partial x}\left\{\ln\rho + \frac{5}{2}\ln\left(\frac{\lambda}{\pi}\right) - \lambda(\vec{u}-\vec{U})\cdot(\vec{u}-\vec{U})\right\} \\ &= \frac{1}{\rho}\frac{\partial\rho}{\partial x} + \frac{5}{2\lambda}\frac{\partial\lambda}{\partial x} - \frac{\partial\lambda}{\partial x}(\vec{u}-\vec{U})\cdot(\vec{u}-\vec{U}) - 2\lambda(\vec{u}-\vec{U})\cdot\frac{\partial\vec{U}}{\partial x},\end{aligned} \tag{A.3}$$

where the differentiation of the velocity vector is to be taken componentwise.

Since in practical implementations only moments of the distribution function are needed it pays to group the expansion coefficients along with the collisional invariants, i.e. the elements of ϕ:

$$a = a_0 + a_1 u + a_2 v + a_3 w + \frac{a_4}{2}(u^2 + v^2 + w^2). \tag{A.4}$$

Suppose we are given the gradients of the conservative variables Q and the primitive variables as input. The following recursive action can be used to compute the expansion coefficients from Eq. (A.3):

$$\partial_x U_i = \frac{1}{\rho}(\partial_x Q_{i+1} - U_i\partial_x Q_1), \quad i = 1,2,3, \tag{A.5}$$

$$\partial_x p = (\gamma-1)\left\{\partial_x Q_5 - \frac{1}{2}\partial_x Q_1(U^2+V^2+W^2) - \frac{1}{2}\vec{U}\cdot\partial_x\vec{U}\right\}, \tag{A.6}$$

$$\partial_x\lambda = \partial_x\left(\frac{\rho}{2p}\right) = \frac{\lambda}{\rho}\partial_x Q_1 - \frac{2\lambda^2}{\rho}\partial_x p, \tag{A.7}$$

$$a_4 = -\partial_x\lambda, \tag{A.8}$$

$$a_3 = 2\lambda\partial_x W - 2Wa_4, \tag{A.9}$$

$$a_2 = 2\lambda\partial_x V - 2Va_4, \tag{A.10}$$

$$a_1 = 2\lambda\partial_x U - 2Ua_4, \tag{A.11}$$

$$a_0 = -2Ua_1 - 2Va_2 - 2Wa_3 + \frac{1}{\rho}\partial_x Q_1 - \frac{5}{2\lambda}a_4 - (\vec{U}\cdot\vec{U})a_4. \tag{A.12}$$

This routine can be used to compute the time derivatives as well. It can be appreciated that the time derivative of the Maxwellian is given by Eq. (A.3) with the space derivatives replaced by time derivatives. Since all time derivatives are computed by the conservation constraint

$$\int \phi\partial_t f_0\,\mathrm{d}\Xi = \int \phi u\partial_x f_0\,\mathrm{d}\Xi, \tag{A.13}$$

we can interpret the right-hand side as the macroscopic derivative which the derivative of the Maxwellian has to reproduce:

$$\int \phi u \partial_x f_0 \, \mathrm{d}\Xi = \partial_t \widetilde{Q}. \tag{A.14}$$

Eqs. (A.5)–(A.12) can then be used with the time derivatives $\partial_t \widetilde{Q}$ to compute the equivalent of the set of coefficients (A.4) for the time derivatives.

References

[1] E. Godlewski, Numerical Approximation of Hyperbolic Systems of Conservation Laws, Springer, Berlin, 1996.
[2] B. Perthame, Kinetic Formulation of Conservation LawsOxford Lecture Series in Mathematics, Oxford University Press, New York, 2002.
[3] D.I. Pullin, Direct simulation methods for compressible inviscid gas flow, J. Comput. Phys. 34 (1980) 231–244.
[4] S.M. Deshpande, A second order accurate, kinetic-theory based, method for inviscid compressible flows, Tech. Paper No. 2613, NASA Langley, 1986.
[5] S.Y. Chou, D. Baganoff, Kinetic flux-vector splitting for the Navier–Stokes equations, J. Comput. Phys. 130 (1996) 217–230.
[6] K. Prendergast, K. Xu, Numerical hydrodynamics from gas-kinetic theory, J. Comput. Phys. 109 (1993) 53–66.
[7] K. Xu, L. Martinelli, A. Jameson, Gas-kinetic finite-volume methods, flux vector splitting and artificial diffusion, J. Comput. Phys. 120 (48) (1995) 48–65.
[8] K. Xu, A gas-kinetic BGK scheme for the Navier–Stokes equations and its connection with artificial dissipation and Godunov method, J. Comput. Phys. 171 (48) (2001) 289–335.
[9] T. Ohwada, S. Kobayashi, Management of discontinuous reconstruction in kinetic schemes, J. Comput. Phys. 197 (2004) 116–138.
[10] K. Xu, M. Mao, L. Tang, A multidimensional gas-kinetic BGK scheme for hypersonic viscous flow, J. Comput. Phys. 203 (2005) 405–421.
[11] D. Chae, C. Kim, O. Rho, Development of an improved gas-kinetic BGK scheme for inviscid and viscous flows, J. Comput. Phys. 158 (2000) 1.
[12] K. Xu, Gas-kinetic schemes for unsteady compressible flow simulations, Vki for fluid dynamics lecture series, 1998-03, 1998.
[13] P. Bhatnagar, E. Gross, M. Krook, A model for collision processes in gases i: small amplitude processes in charged and neutral one-component systems, Phys. Rev. 94 (1954) 511.
[14] T. Ohwada, On the construction of kinetic schemes, J. Comput. Phys. 177 (2002) 156–175.
[15] A. Haselbacher, J. Blazek, Accurate and efficient discretization of Navier–Stokes equations on mixed grids, AIAA J. 38 (11) (2000) 2094–2102.
[16] A. Jameson, Solution of the Euler equations for two dimensional transonic flow by a multigrid method, Appl. Math. Comput. 13 (1983) 327–356.
[17] V. Venkatakrishnan, Convergence to steady state solutions of the Euler equations on unstructured grids with limiters, J. Comput. Phys. 118 (1995) 120–130.
[18] B. van Leer, Towards the ultimate conservative difference scheme iv: a new approach to numerical convection, J. Comput. Phys. 23 (1995) 276–299.
[19] J.D. Anderson, Fundamentals of Aerodynamics, McGraw-Hill, New York, 2001.
[20] A. Jameson, Analysis and design of numerical schemes for gas dynamics 2: artificial diffusion and discrete shock structure, Int. J. Comp. Fluid. Dyn. 5 (1995) 1–38.
[21] F. Bassi, S. Rebay, A high-order accurate discontinuous finite-element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (1997) 267–279.
[22] INRIA, Gamm Workshop, December 4–6, 1985, Nice, France. Numerical simulation of compressible Navier–Stokes equations-external 2d flows around a naca0012 airfoil, Centre de Rocquefort, de Rennes et de Sophia-Antipolis, 1986.
[23] G. May, A. Jameson, Unstructured algorithms for inviscid and viscous flows embedded in a unified solver architecture: Flo3xx, AIAA Paper 2005-0318, 2005.