Computers and Fluids 145 (2017) 167–179

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Simulation of real gas effects in supersonic methane jets using a tabulated equation of state with a discontinuous Galerkin spectral element method

F. Hempert[a,*], S. Boblest[b], T. Ertl[b], F. Sadlo[c], P. Offenhäuser[d], C.W. Glass[d], M. Hoffmann[e], A. Beck[e], C.-D. Munz[e], U. Iben[f]

[a] Robert Bosch GmbH, Robert-Bosch-Campus 1, 71272, Renningen, Germany
[b] Visualization Research Center, University of Stuttgart, Allmandring 19, 70569 Stuttgart, Germany
[c] Interdisciplinary Center for Scientific Computing, Heidelberg University, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
[d] High Performance Computing Center, University of Stuttgart, Nobelstrasse 19, 70569 Stuttgart, Germany
[e] Institute for Aerodynamics and Gas dynamics, University of Stuttgart, Pfaffenwaldring 21, 70569 Stuttgart, Germany
[f] Robert Bosch GmbH, Shvetsova Str. 41, 198095 St. Petersburg, Russia

ARTICLE INFO

Article history:
Received 28 June 2016
Revised 13 December 2016
Accepted 22 December 2016
Available online 23 December 2016

Keywords:
Discontinuous Galerkin
General equation of state
Navier–Stokes equations
Supersonic jet
Shocks

ABSTRACT

Modern direct gas injectors for natural-gas-powered internal combustion engines operate at ever-increasing pressures to maximize efficiency. At the operating point of current devices, real gas effects already become relevant, for example in the determination of mass flow rates. These effects have to be considered in the design process of such components. Motivated by this fact, we investigate the real gas effects of high-pressure supersonic methane jets. A discontinuous Galerkin spectral element method for computational fluid dynamics is used in combination with a tabulated equation of state for methane. We first evaluate self-similar profiles for subsonic jets to validate our simulation approach. Then we discuss our simulations of a supersonic jet, where we observe well-resolved shocks and transient structures. Additionally, real and ideal gas modeling are compared and we discuss the significant differences in general flow structure as well as density, pressure and mass flow that occur and make an ideal-gas description inappropriate for our simulation. The influence of pressure level and temperature on the behavior of the jet is discussed. We analyze the influence of the thermodynamic and viscous properties of the gas for the shock location and mass flow. The use of a tabulated equation of state makes jet simulations with occurring phase changes possible.

1. Introduction

One of the most promising alternatives to conventional combustion fuel for automotive vehicles is natural gas (NG) [1,2]. It allows both to further reduce pollutant emission and to decrease the operating costs, while this is very challenging with conventional fuel, e.g., gasoline. The carbon-to-hydrogen ratio of the used world energy supply has decreased in the past. This indicates a trend towards natural gas, which will most likely continue in the coming years [3]. Therefore, the automotive industry strives towards an accurate and efficient designing process for the advance development of NG injection systems. To enable this, the correct prediction of the structure and the development of compressible jets are of vital importance. Compressible supersonic jets have been a topic of research in many different areas, e.g., noise emission [4–7], hazardous effects of gas leaks [8] or different transient stages of jet development [9–11].

Further, direct gas injectors for natural-gas-powered internal combustion engines operate at high pressures [12,13] to maximize efficiency and reduce emissions. At these conditions, real gas effects such as the pressure-dependent compressibility or viscosity can no longer be neglected, i.e., the finite molecule volume and attraction forces between the gas molecules become relevant. In the present study we focus on two aspects of compressible jets: the transient behavior of jets and the influence of real gas effects.

Even today, direct numerical simulation of turbulent jets with high Reynolds numbers is beyond computational feasibility. However, Large-Eddy simulations (LES) have become possible in recent years both for simulations of ideal gases and also when real gas effects are taken into account. Yu et al. [14] investigated the

* Corresponding author.
E-mail address: fabian.hempert@de.bosch.com (F. Hempert).

http://dx.doi.org/10.1016/j.compfluid.2016.12.024
0045-7930/

transient behavior of a natural gas (NG) jet both experimentally and numerically in the ideal gas regime. They used planar laser-induced fluorescence for the experiment and compared the results with LES near the nozzle exit. The experimental and numerical results of their study are in good agreement and they concluded that the simulation provided detailed flow information of the underexpanded jet, e.g., temperature distribution and Mach number.

Vuorinen et al. [9] focus on the variation of inlet to outlet pressure ratios with ideal gas in a similar numerical setup as used in the aforementioned study. Their work comprises underexpanded jets with pressure ratios of 4.5–8.5 at Reynolds numbers of the order 7.5×10^4–1.4×10^5. The transient stages of the underexpanded jet are analyzed and key features are discussed. Further, with a proper orthogonal decomposition, coherent structures downstream of the Mach disk are identified.

To model real gas effects, cubic equations of state (EOS) are most commonly used in computational fluid dynamics (CFD) [15]. These equations are based on the idea of the Van-der-Waals equation and account for attractive forces between gas molecules and for their finite volume. Current state of the art approaches for the simulation of real gas are the Redlich–Kwong [16], Soave–Redlich–Kwong [17], and the Peng–Robinson [18] equation of state. Less frequently used approaches are the Beattie–Bridgeman [19] and Abel-Noble EOS [20].

Bonelli et al. [21] and Hamzehloo and Aleuferus [10] demonstrated the necessity of a proper real gas EOS for a hydrogen jet at high injection pressure $p_0 > 1 \times 10^7$ Pa. They found significant differences between the ideal gas, the Van-der-Waals equation, and the Redlich–Kwong formulation with respect to Mach number, pressure, and temperature within the flow regime.

Khaksarfard et al. [8] investigated the high pressure release of a hydrogen storage tank. The study compared the Beattie–Brideman equation and the Abel-Noble equation and evaluated the differences to the ideal gas formulation. The authors found significant differences in the flow structures, e.g., shock location and intensity and demonstrated the need for a real gas EOS, although, for the investigated case, the Abel-Noble formulation was preferred due to computation costs and numerical stability.

In conclusion, the predictive capabilities of simulations of underexpanded jets and their transient development have improved significantly over the last years. For the representation of the EOS in LES CFD frameworks, ideal gas or cubic EOS formulations are most commonly used. Some tabulated approaches have been employed in real gas applications [22–24].

In the present paper, we use a tabulated equation of state for methane, the main component of NG [25], which allows for an efficient and accurate representation of the fluid properties. We compare the efficiency and the accuracy with state-of-the-art EOS, e.g., Peng–Robinson and validate our method in a simulation of a self-similar jet. Further, a supersonic jet at different pressures and temperatures is analyzed in detail. The paper focuses on a study of the real gas effects at high pressures, which are gaining relevance for industrial applications.

2. Methodology

The currently used CFD code is an implementation of the Discontinuous Galerkin Spectral Element Method (DG-SEM). The DG-SEM algorithm is very well parallelizable and thus ideally suited for modern high-performance computing [26]. Details about this method can be found in, e.g, [27]. However, to make this paper self-contained, we give a short summary of its most important features.

2.1. Discontinuous Galerkin spectral element method

Starting point are the compressible Navier–Stokes equations for the computational domain Ω expressed in conservative form as

$$\boldsymbol{u}_t(\boldsymbol{x},t) + \nabla_x \cdot \boldsymbol{F}(\boldsymbol{u}(\boldsymbol{x},t), \nabla_x \boldsymbol{u}(\boldsymbol{x},t)) = \boldsymbol{0} \quad \forall \boldsymbol{x} \in \Omega, \tag{1}$$

where $\boldsymbol{u}$ is the vector of conservative variables and $\boldsymbol{F}$ contains the physical fluxes, i.e., advection and viscous fluxes. Within our code, the domain Ω is split into hexahedral grid cells. To facilitate the treatment of complex geometries we allow for curved grid cells.

Each of the grid cells is mapped to the reference element $E = [-1,1]^3$ with coordinates $\boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3)^\top$. The discretization operators are formulated in the reference element. This transformation yields

$$J(\boldsymbol{\xi})\boldsymbol{u}_t(\boldsymbol{\xi},t) + \nabla_\xi \cdot \mathcal{F}(\boldsymbol{u}(\boldsymbol{\xi},t), \nabla_x \boldsymbol{u}(\boldsymbol{\xi},t)) = \boldsymbol{0}, \tag{2}$$

where $J(\boldsymbol{\xi})$ is the Jacobian of the mapping and $\mathcal{F}$ are the transformed fluxes. In the weak formulation of discontinuous Galerkin (DG), which we apply, the conservation laws (2) are multiplied component-wise by a test function Φ and the resulting equations are integrated over the reference element E. Here, integration by parts is applied to the second integral:

$$\int_E J\boldsymbol{u}_t \Phi \,\mathrm{d}\boldsymbol{\xi} + \oint_{\partial E} (\mathcal{F}^* \cdot \boldsymbol{n}) \Phi \,\mathrm{d}S_\xi - \int_E \mathcal{F} \cdot \nabla_\xi \Phi \,\mathrm{d}\boldsymbol{\xi} = 0, \tag{3}$$

where $\boldsymbol{n}$ is the transformed normal vector of the corresponding interface to the reference base and $\mathcal{F}^*$ is the boundary flux. Within the reference element, the solution is approximated as

$$\boldsymbol{u}_h(\boldsymbol{\xi},\boldsymbol{t}) = \sum_{i,j,k=1}^{N+1} \hat{\boldsymbol{u}}_{ijk}(t)\psi_{ijk}(\boldsymbol{\xi}) \quad \text{with } \psi_{ijk} = l_i(\xi^1) l_j(\xi^2) l_k(\xi^3), \tag{4}$$

by using a polynomial tensor product basis of degree N in each spatial direction. Here, $\hat{\boldsymbol{u}}_{ijk}$ are the nodal degrees of freedom (DOFs), and $l_i(\boldsymbol{\xi})$ are the one-dimensional Lagrange interpolation polynomials defined by the set of Gauss nodes $\{\xi_i\}_{i=1}^{N+1}$. DG-SEM uses a collocation technique which means that the interpolation and integration points are equal. Furthermore, the DG method uses the same basis and test functions $\Phi = \psi$.

In the DG method, adjacent elements are coupled by fluxes through their interfaces. Adjacent elements have different states at the interface because the approximate solution may be discontinuous over element boundaries. A numerical flux function is used to approximate the boundary flux $\mathcal{F}^*$ with information from the state $\boldsymbol{u}$ of an element and the state $\boldsymbol{u}^+$ of the neighbor element, as well as the respective gradients. In the present study, we use the local Lax–Friedrichs (LF) method [28]. The evaluation of the volume integral and surface integral determines the time derivative $\partial\hat{\boldsymbol{u}}_{ijk}/\partial t$ for each degree of freedom (DOF). Time integration is done with an explicit 3rd-order accurate Runge-Kutta scheme.

To derive the viscous flux terms, the governing equations are rewritten as a corresponding system of first order equations [29]. Here, an additional variable $\boldsymbol{S}$ is introduced as an estimate for the solution gradients:

$$\begin{aligned} &\boldsymbol{S} - \nabla_x \boldsymbol{u} = 0, \\ &\boldsymbol{u}_t + \nabla_x \cdot \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{S}) = 0. \end{aligned} \tag{5}$$

In addition, we define the boundary flux

$$\mathcal{F}^*(\boldsymbol{u}^+, \boldsymbol{u}^-, \boldsymbol{S}^+, \boldsymbol{S}^-) = \mathcal{G}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \mathcal{H}^*(\boldsymbol{u}^+, \boldsymbol{u}^-, \boldsymbol{S}^+, \boldsymbol{S}^-), \tag{6}$$