A pressure-based solver for low-Mach number flow using a discontinuous Galerkin method

Hennink, Aldo; Tiberga, Marco; Lathouwers, Danny

DOI
10.1016/j.jcp.2020.109877

Publication date
2021

Document Version
Final published version

Published in
Journal of Computational Physics

Citation (APA)
Hennink, A., Tiberga, M., & Lathouwers, D. (2021). A pressure-based solver for low-Mach number flow using a discontinuous Galerkin method. *Journal of Computational Physics*, *425*, Article 109877. https://doi.org/10.1016/j.jcp.2020.109877

Journal of Computational Physics 425 (2021) 109877

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A pressure-based solver for low-Mach number flow using a discontinuous Galerkin method

Aldo Hennink *, Marco Tiberga, Danny Lathouwers

Department of Radiation Science and Technology, Delft University of Technology, The Netherlands

ARTICLE INFO

Article history:
Received 8 November 2019
Received in revised form 22 September 2020
Accepted 26 September 2020
Available online 2 October 2020

Keywords:
Low-Mach
Variable properties
Discontinuous Galerkin
Pressure correction

ABSTRACT

Over the past two decades, there has been much development in discontinuous Galerkin methods for incompressible flows and for compressible flows with a positive Mach number, but almost no attention has been paid to variable-density flows at low speeds. This paper presents a pressure-based discontinuous Galerkin method for flow in the low-Mach number limit. We use a variable-density pressure correction method, which is simplified by solving for the mass flux instead of the velocity. The fluid properties do not depend significantly on the pressure, but may vary strongly in space and time as a function of the temperature.

We pay particular attention to the temporal discretization of the enthalpy equation, and show that the specific enthalpy needs to be 'offset' with a constant in order for the temporal finite difference method to be stable. We also show how one can solve for the specific enthalpy from the conservative enthalpy transport equation without needing a predictor step for the density. These findings do not depend on the spatial discretization.

A series of manufactured solutions with variable fluid properties demonstrate full second-order temporal accuracy, without iterating the transport equations within a time step. We also simulate a Von Kármán vortex street in the wake of a heated circular cylinder, and show good agreement between our numerical results and experimental data.

1. Introduction

Several low-speed flows of practical importance are compressible, that is, the velocity is not divergence-free. This can occur due to mixing, or due to a temperature-dependent density near a heat source. An example is heat transfer in low-Mach number flows of supercritical fluids, where all fluid properties vary strongly with the temperature, but do not depend significantly on the pressure. Most flow solvers use either a pressure-based approach and assume a divergence-free velocity field, or a fully compressible (density-based) formulation. Neither of these methods is directly applicable to compressible flows in the low-Mach number limit.

Density-based solvers can be used to simulate zero-Mach flows by approximating the flow with a low, non-zero Mach number (e.g., [46], [11]). This has often been used for heat transfer in supercritical fluids at low speeds (e.g., [14], [4]). This is expensive for several reasons. First, the temporal discretization needs to resolve acoustic effects, and the resulting linear systems tend to be very stiff. Second, the system of transport equations is solved in a coupled way, which is more expensive than using a time-splitting method, though the performance may be improved with suitable preconditioning [31]. Finally,

* Corresponding author.
E-mail address: a.hennink@protonmail.ch (A. Hennink).

https://doi.org/10.1016/j.jcp.2020.109877
0021-9991/

the fluid properties are evaluated as a function of two thermodynamic variables (usually the density and the volumetric enthalpy), so that a spline interpolation costs far more memory, thus complicating massively parallel calculations [14].

There is also substantial experience with discontinuous Galerkin (DG) discretizations for incompressible flows. These are either based on the introduction of artificial compressibility (e.g. [12]), or they solve for the pressure (e.g. [41,8,37,35]). The artificial compressibility method can be more than second-order accurate in time, though it requires the system of transport equations to be solved in a coupled manner (e.g., [6,5]). By choosing entropy variables as the unknowns, the DG method can also be formulated in a general way for both compressible and incompressible flows, at the cost of great complexity (e.g., [34]). There is, however, almost no literature on solving the low-Mach number equations with a pressure-based discontinuous Galerkin method.

The only previous work of which we are aware is by Klein et al. [24,25], who used a SIMPLE scheme to march the transport equations forward in time, iterating the equations within each time step. This required under-relaxation in order for the iteration to converge. They solved for the velocity, so that a predictor for the density is needed in the temporal derivative of the momentum equation.

We avoid this by solving for the mass flux rather than the velocity. Another advantage of this approach is that the divergence term in the continuity equation does not have to be weighed by the density, so that the divergence matrix does not depend on the density. This makes the transport equations less tightly coupled, and it simplifies the pressure correction method, because the pressure matrix is constant for each time step.

The temporal discretization of the enthalpy equation requires special care. The convective term in the primitive enthalpy equation cannot be upwinded with a finite element method if the velocity is not divergence-free. We therefore discretize the enthalpy equation in conservative form, although we solve for the specific enthalpy, which is a primitive variable. This causes a complication in the temporal discretization, which features the unknown density at a new time step. A similar issue occurs in multispecies transport, and Najm et al. [28] devised a wide-spread two-step iterative method to stabilize the temporal scheme. This has subsequently been adapted to handle the strong property variations in supercritical fluids [29].

We present an alternative method that does not use any predictor solves or iterations to handle the unknown density at a new time step. We show that the error in our approximation can be made negligible compared to the error in the finite difference scheme, and that the method can be made stable by offsetting the specific enthalpy with a constant.

The paper is structured as follows. Section 2 defines the mathematical problem. Section 3 and 4 explain the spatial discretization and the time-splitting scheme, paying particular attention to the polynomial solution spaces (Section 3.1) and the discretization of the viscous stress (Section 3.2.1), which differ slightly from previous literature. Section 5 and 6 treat the density predictor in the enthalpy equation in great detail, and illustrate the results with numerical examples; these sections do not depend on the spatial discretization, and they can be read independently. Section 7 verifies the numerical method and the implementation for the coupled transport equations with a series of progressively more challenging manufactured solutions, and validates our approach by reproducing experimental data for flow past a heated circular cylinder. Finally, we draw our conclusions based on the results in Section 8.

2. Governing equations

The transport equations in the low-Mach number limit are

$$\frac{\partial \rho}{\partial t} + \partial_k m_k = 0 \,, \tag{1a}$$

$$\frac{\partial m_i}{\partial t} + \partial_k \left(u_k m_i\right) = \partial_k \tau_{ik} - \partial_i p + F_i \,, \tag{1b}$$

$$\frac{\partial \rho h}{\partial t} + \partial_k \left(m_k h\right) = -\partial_k q_k + Q \,. \tag{1c}$$

Here t is the time, ρ is the density, u is the velocity, $m := \rho u$ is the mass flux, p is the pressure, h is the specific enthalpy, and F and Q are known external sources. Assuming a Newtonian fluid, the stress tensor is

$$\tau_{ij} = \mu \left(\partial_i u_j + \partial_j u_i - \frac{2}{3} \partial_k u_k \delta_{ij} \right) \,. \tag{2}$$

The heat flux is

$$q_i = -k\, \partial_i T = -\frac{k}{c_p} \partial_i h \,, \tag{3}$$

where T is the temperature. k is the thermal conductivity, and c_p is the specific heat capacity. The last equality is technically an approximation because it neglects the dependence of the temperature on the pressure, but this is highly accurate for low-Mach number flows, even for strongly variable fluid properties in supercritical fluids [33]. The fluid properties (ρ, μ, c_p, and k) are a function of T, but do not depend significantly on the pressure.

We consider two types of boundaries:

- Dirichlet boundaries, denoted by Γ^{D}, on which the mass flux and the temperature (and, therefore, the enthalpy and the fluid properties) are given, that is, $m_i = m_i^{\mathrm{D}}$ and $T = T^{\mathrm{D}}$;
- outflow boundaries, denoted by Γ^{N}, where $(\tau_{ik} - p\ \delta_{ik})\, n_k = f_i^{\mathrm{N}}$ and $k\ n_l \partial_l T = q^{\mathrm{N}}$ are prescribed.

3. Spatial discretization

This section details the spatial discretization of the enthalpy and mass flux transport equations on a domain Ω with outward normal n and space vector $r = [x, y] \in \Omega$. The domain is meshed into a set of elements $\mathcal{T}$. We denote by $\mathcal{F}^{\mathrm{i}}$, $\mathcal{F}^{\mathrm{D}}$, and $\mathcal{F}^{\mathrm{N}}$ the sets of internal faces, Dirichlet boundary faces, and Neumann boundary faces. The set of faces of an element $T \in \mathcal{T}$ is $\mathcal{F}_T$, and $\mathcal{T}_F$ is the set of neighbors of face F. Each internal face $F \in \mathcal{F}^{\mathrm{i}}$ has a normal vector n^F with an arbitrary but fixed direction. The jump and average operators are defined as

$$[\![x]\!] := x^+ - x^- \quad \text{and} \quad \{x\} := \frac{1}{2}\left(x^- + x^+\right), \tag{4}$$

where

$$x^{\pm}(r \in F) := \lim_{\epsilon \downarrow 0} x(r \mp \epsilon\, n^F) \tag{5}$$

indicate the values at the two sides. On boundary elements, both the jump and the average equal the internal numerical value, and n^F coincides with the outward normal of Ω.

The solution space within each element is spanned by all polynomials up to an order $\mathcal{P}$. We denote the order of the polynomials for the unknown X by $\mathcal{P}_X$. For the numerical examples in this paper, the polynomial order is the same on all elements, though this is not a requirement of the numerical method.

3.1. Solutions spaces for enthalpy and pressure

In this paper the basis functions for the enthalpy and the pressure are always of the same order, that is, $\mathcal{P}_h = \mathcal{P}_p$. Neglecting the source term, the enthalpy equation in its conservative form is

$$\frac{Dh}{Dt} = \frac{1}{\rho}\partial_l\left(k\partial_l T\right), \tag{6}$$

where D/Dt is the total derivative. If $\partial\rho/\partial h < 0$ for all h, then this becomes

$$\frac{1}{\rho}\frac{D\rho}{Dt} = \frac{\partial\rho}{\partial h}\frac{1}{\rho^2}\partial_l\left(k\partial_l T\right) = -\frac{\beta}{\rho c_p}\partial_l\left(k\partial_l T\right) = -\alpha\beta\,\frac{1}{k}\partial_l\left(k\partial_l T\right), \tag{7}$$

where $\beta := -(1/\rho)\partial\rho/\partial T$ is the thermal expansibility, and $\alpha := k/\rho c_p$ is the thermal diffusivity. Compare the continuity equation

$$\frac{1}{\rho}\frac{D\rho}{Dt} = -\partial_k u_k\,. \tag{8}$$

Regardless of the Péclet number or the compressibility, the right-hand-sides of Eqs. (7) and (8) can be arbitrarily small in any part of the domain, in which case the enthalpy equation and the continuity equation are almost equivalent. Since the continuity equation in its weak form is weighed by the pressure basis functions, it seems inconsistent to use different polynomial orders for the pressure and the enthalpy if $\partial\rho/\partial h < 0$.

Furthermore, our numerical experiments (not shown here) demonstrate that, for low diffusivity, the spatial discretization is unstable if $\mathcal{P}_h > \mathcal{P}_p$, even if the enthalpy is a passive scalar, the density is constant, and we simulate a laminar steady state. The reason for the instability is that the convective velocity in the enthalpy discretization must satisfy the continuity equation up to the order $\mathcal{P}_h$, in a weak sense. This observation contrasts with Klein et al. [24], who used a polynomial order for the temperature that was higher than for the pressure. This discrepancy seems to be because their tests were done at a low Prandtl number of 0.7. We also obtained accurate results with $\mathcal{P}_h > \mathcal{P}_p$ for the 2D buoyancy-driven cavity at various Richardson numbers, but the instability manifests itself when the thermal diffusion vanishes. We will show this in future work.

3.2. Momentum discretization

We solve for the mass flux m from the conservative transport equation (1b). Klein et al. [24] have previously solved the same conservative equation by taking the velocity u as the unknown, but this presents difficulties. First, one would have to handle the strongly fluctuating density in the temporal derivative $\partial(\rho u)/\partial t$. Second, solving for u would complicate the pressure correction scheme, as we will see in Section 4. Another alternative is to solve for u from the primitive equation

$$\rho\left(\frac{\partial u_i}{\partial t} + u_k \partial_k u_i\right) = \partial_k \tau_{ik} - \partial_i p + F_i \,, \tag{9}$$

as in common for incompressible flows. (See [17,1] for examples with a variable density.) Unfortunately we cannot upwind the convective term in Eq. (9), because it cannot be rewritten as $\partial_k (u_i u_k)$ when the compressibility $(\partial_k u_k)$ is non-negligible. Our approach of taking m as the unknown in the conservative equation slightly complicates the implicit treatment of the viscous term, which is linear in $\partial_i u$, not $\partial_i m$. Section 3.2.1 details how this can be handled with a DG method.

The semi-discrete weak form of the momentum equation is

Find $m \in V$, such that, for all $v \in V$,

$$\int_\Omega v_k \frac{\partial m_k}{\partial t} + a^{\text{conv}}(u, m_k, v_k) + a^{\text{visc}}(m, v) = l^{\text{conv}}(u, m_k^{\text{D}}, v_k) + l^{\text{visc}}(v) - a^{\text{div}}(v, p) + \int_\Omega F_k v_k \,, \tag{10}$$

where V is the solution space of the mass flux. The divergence operator a^{div} is (see, e.g., [41], [21, pp. 92], or [36, pp. 250–252])

$$a^{\text{div}}(v, q) = -\sum_{T\in\mathcal{T}} \int_T q\, \partial_k v_k + \sum_{F\in\mathcal{F}^{\text{i}}\cup\mathcal{F}^{\text{D}}} \int_F \{q\}\, [\![v_k]\!]\, n_k^F \,, \tag{11}$$

where q is a scalar that lies in the solution space of the pressure.

3.2.1. Discretization of the viscous stress

To derive a discretization for the viscous term, rewrite the viscous stress in terms of the mass flux as $\tau = L^{\text{visc}}(m)$, where

$$L_{ij}^{\text{visc}}(m) = \frac{\mu}{\rho}\left(\partial_i m_j + \partial_j m_i - \frac{2}{3}(\partial_k m_k)\,\delta_{ij} - d_i m_j - d_j m_i + \frac{2}{3}(d_k m_k)\delta_{ij}\right) \,, \tag{12}$$

is a linear operator, and

$$d_i := \frac{1}{\rho}\,\partial_i \rho \,. \tag{13}$$

We use a generalization of the symmetric interior penalty (SIP) method, given by the discrete bilinear operator

$$\begin{aligned} a^{\text{visc}}(w, v) = &\sum_{T\in\mathcal{T}} \int_T L_{kl}^{\text{visc}}(w)\, \partial_l v_k + \sum_{F\in\mathcal{F}^{\text{i}}\cup\mathcal{F}^{\text{D}}} \int_F \eta^F\, [\![w_k]\!]\, [\![v_k]\!] \\ &- \sum_{F\in\mathcal{F}^{\text{i}}\cup\mathcal{F}^{\text{D}}} \int_F \left([\![v_k]\!] \left\{ L_{kl}^{\text{visc}}(w) \right\} + [\![w_k]\!] \left\{ L_{kl}^{\text{visc}}(v) \right\} \right) n_l^F \end{aligned} \tag{14}$$

and

$$l^{\text{visc}}(v) = \sum_{F\in\mathcal{F}^{\text{D}}} \int_F \left(\eta^F\, m_k^{\text{D}}\, v_k - m_k^{\text{D}}\, L_{kl}^{\text{visc}}(v)\, n_l \right) + \int_{\Gamma^{\text{N}}} f_k^{\text{N}} v_k \,. \tag{15}$$

This reduces to the regular SIP method for constant-density, constant-viscosity flows when substituting $(\mu/\rho)\partial_l w_k$ for $L_{kl}^{\text{visc}}(w)$. Compared to other interior penalty methods and the local DG method, the advantages of the SIP method are the optimal convergence rate for all polynomial orders, its adjoint consistency, and its compact small stencil [18].

This discretization of the viscous term can be compared to what is usually done for compressible flows, where the system of equations (1) is solved for a full state vector $U := [\rho, m, \rho h]^{\text{T}}$. In that case all elliptic terms in Eqs. (1) can be written as $\partial_k(G(U)\partial_k U))$, where $G(U)$ is a homogeneity tensor that does not contain any gradients of the unknowns. (See, e.g., [18,23].) This tensor is then kept fixed during an iteration step, while $\partial_i U$ can be treated implicitly. Eq. (12) shows that each term of the viscous stress tensor is bilinear in U: they are products of m and ρ. The last three terms contain a gradient of ρ, and therefore the mass flux is evaluated in the homogeneity tensor, and frozen within an iteration step. This differs from our implicit diffusion discretization, where ρ is frozen in each term in Eq. (12), and m is treated implicitly.

Our approach also differs from that of Klein et al. [24], in that we treat all terms in the viscous stress (Eq. (2)) in a time-implicit manner, whereas they only do this for the first term $(\mu\partial_i u_j)$. In our treatment the velocity components are coupled. We have no a priori estimate for the difference in magnitude between the effects of the first term $(\mu\partial_i u_j)$ and its transpose $(\mu\partial_j u_i)$ on the viscous force $\partial_i \tau_{ij}$, especially when the viscosity varies strongly in space. Note the gradients in the effective viscosity would increase greatly if a large eddy simulation (LES) model were included.

Following Hillewaert [20, pp. 30], we set the penalty parameter to

$$\eta^F = \max_{T\in\mathcal{T}_F}\left(C_T\, \text{card}(\mathcal{F}_T)\, \frac{\|F\|_{\text{leb}}}{\|T\|_{\text{leb}}} \right) \max_{T\in\mathcal{T}_F} (K|_T) \,, \tag{16}$$

where $K = \mu/\rho$ is the diffusion parameter, $\text{card}(\mathcal{F}_T)$ is the number of faces of element T, and $\|\cdot\|_{\text{leb}}$ denotes the Lebesgue measure (which is the length, area, or volume in 1, 2, or 3 dimensions). The factor C_T depends on the type of elements in the mesh: for a polynomial order $\mathcal{P}$, $C_T = (\mathcal{P}+1)^2$ for quadrilaterals and hexahedra, $C_T = (\mathcal{P}+1)(\mathcal{P}+2)/2$ on triangles, and $C_T = (\mathcal{P}+1)(\mathcal{P}+3)/3$ for tetrahedra. We compute the penalty parameter pointwise, rather than averaging over the face.

3.2.2. Discretization of convective term

The discretization of the convective term is given by

$$\begin{aligned} a^{\text{conv}}(b,\phi,\psi) = &-\sum_{T\in\mathcal{T}}\int_T \psi\, b_k \partial_k \phi + \sum_{F\in\mathcal{F}^{\text{i}}}\int_F [\![\psi]\!]\, H^F(b,\phi) \\ &+ \int_{\Gamma^{\text{N}}} (n_k b_k)\,\psi\,\phi + \int_{\Gamma^{\text{D}}} \max\left(0, n_k b_k^{\text{D}}\right)\psi\,\phi\,, \end{aligned} \tag{17}$$

$$l^{\text{conv}}(b,\phi,\psi) = -\int_{\Gamma^{\text{D}}} \min\left(0, n_k b_k^{\text{D}}\right)\psi\,\phi\,. \tag{18}$$

Here ϕ and ψ are scalars, b is the convective field, and H^F is the numerical flux function on a face F. We use the Lax-Friedrichs flux, given by

$$H^F(b,\phi) = \frac{1}{2}[\![\phi]\!]\,\alpha^F(b) + n_k^F\{\phi b_k\} \tag{19}$$

and

$$\alpha^F(b) = f \max_{T\in\mathcal{T}_F} \left| n_k^F b_k \right|_T\,. \tag{20}$$

The constant f is $f=2$ when convecting the mass flux, and $f=1$ when convecting a scalar, such as the enthalpy [9]. Note that we evaluate α^F pointwise in the integral in Eq. (17), rather than averaging over the face.

It is well known that imposing a Dirichlet boundary condition for the velocity is not stable at an outlet of a convection-dominated flow, so that we would normally have $n_k b_k^{\text{D}} < 0$ on Γ^{D}. Here we nevertheless include the Dirichlet boundary term in a^{conv}, because we will use it in the Taylor-Green vortex in Section 7.2, as is standard practice for that laminar benchmark case (e.g., [13]).

3.3. Enthalpy discretization

We solve for the specific enthalpy h because it fixes the thermodynamic state for a given thermodynamic pressure p_{th}. This is not true for the volumetric enthalpy ($H := \rho h$): the pair (p_{th}, H) does not uniquely determine the density or the other fluid properties. Choosing h as the unknown also simplifies the time-implicit treatment of the Fourier heat flux.

As with the mass flux equation, we solve the enthalpy equation in conservative form (Eq. (1c)), because we cannot deal with the convective term $(u_k\partial_k h)$ in the primitive transport equation when $\partial_k u_k \neq 0$. Given a solution space W for the enthalpy, which is also the solution space for the pressure, the semi-discrete weak form of the enthalpy equation is

Find $h \in W$, such that, for all $v \in W$,

$$\int_\Omega v\frac{\partial(\rho h)}{\partial t} + a^{\text{conv}}(m,h,v) + a^{\text{SIP}}(h,v) = l^{\text{conv}}(m,h^{\text{D}},v) + l^{\text{SIP}}(v) + \int_\Omega Q\; v\,, \tag{21}$$

where a^{SIP} and l^{SIP} are standard SIP bilinear and linear forms to discretize the Fourier heat flux. The SIP penalty parameter is as in Eq. (16), with a diffusion coefficient $K = k/c_p$. Note that the convective discretization is the same as for the mass flux, except that the convective field is m, rather than $u = m/\rho$. (That is, $b = m$ in Eqs. (17)-(20).) This is another advantage of solving for the pair (m, h).

4. Pressure correction method

We discretize the temporal derivatives in both the enthalpy and momentum transport equations with standard backward-difference formulae (BDF), thereby following previous DG literature (e.g., [41,24,26]). For the mass flux this is straightforward: for a constant time step size δt,

$$\frac{\partial m}{\partial t} \approx \frac{\gamma_0}{\delta t} m^n + \sum_{i=1}^{q} \frac{\gamma_i}{\delta t} m^{n-i}\,, \tag{22}$$

Table 1
Coefficients for the backward difference formula of various orders.

	γ_0	γ_1	γ_2	γ_3
BDF1	1	−1		
BDF2	3/2	−2	1/2	
BDF3	11/6	−3	3/2	−1/3

where m^n is the mass flux at time step n. The weights $\{\gamma_i\}_{i=0}^{q}$ are listed in Table 1. The temporal discretization of the enthalpy equation requires more care, as we will explain later in Section 5.

A pressure-correction scheme is used to split the continuity and the momentum equations, so that they can be solved in a segregated way. The algorithm to find the solution vectors $\mathbf{m}^n$, $\mathbf{h}^n$, $\mathbf{p}^n$ at a new time step n is as follows.

1. Obtain predictors for ρ^*, $(k/c_p)^*$, and m^* with a second-order extrapolation from previous time steps:
$$(\cdot)^* = 2(\cdot)^{n-1} - (\cdot)^{n-2} . \tag{23}$$
2. Solve for the enthalpy $\mathbf{h}^n$ at the new time step, using the above predictors for the diffusion constant ($(k/c_p)^*$) and the convective field (m^*).
Linearize the implicit time term with the predictors using either of the methods that are explained in Section 5.
3. Find a predictor for the mass flux from a linear system that corresponds to the semi-discrete weak form (10):
$$\left(\frac{\gamma_0}{\delta t} M + N\right) \hat{\mathbf{m}} = -D^{\intercal} \mathbf{p}^{n-1} + \mathbf{f} . \tag{24}$$
Here M is the mass matrix, N contains the implicit parts of the diffusion and convection discretizations, and $\mathbf{f}$ collects all explicit terms (i.e., boundary conditions, the external force, and the explicit terms in the temporal finite difference scheme). The matrix N contains the viscosity and the density, which are evaluated at the new time step as a function of h^n. The convective field is estimated as m^*/ρ^n.
4. Solve the pressure Poisson equation
$$\frac{\delta t}{\gamma_0} A\, \delta\mathbf{p} = D\hat{\mathbf{m}} - \mathcal{G}\left[\left(\frac{\partial \rho}{\partial t}\right)^n\right] , \tag{25}$$
where A is the pressure matrix, for which we derive an expression below, and $\mathcal{G}[\cdot]$ denotes the Galerkin projection onto the solution space. The temporal derivative of the density is estimated with a second-order finite difference scheme:
$$\left(\frac{\partial \rho}{\partial t}\right)^n \approx \frac{1}{\delta t}\left(\frac{3}{2}\rho^n - 2\rho^{n-1} + \frac{1}{2}\rho^{n-2}\right) . \tag{26}$$
5. Correct the pressure and the mass flux:
$$\mathbf{m}^n = \hat{\mathbf{m}} - \frac{\delta t}{\gamma_0} M^{-1} D^T \delta\mathbf{p} , \tag{27}$$
$$\mathbf{p}^n = \mathbf{p}^{n-1} + \delta\mathbf{p} . \tag{28}$$

The test cases in Section 7.3 will show full second-order temporal accuracy in the enthalpy and the mass flux, even if the fluid properties are non-trivial functions of the enthalpy.

To find an expression for the pressure matrix A, use Eq. (25) to eliminate $\delta\mathbf{p}$ in Eq. (27), and left-multiply by D, giving
$$D\mathbf{m}^n = D\hat{\mathbf{m}} - DM^{-1}D^{\intercal}A^{-1}\left(D\hat{\mathbf{m}} - \mathcal{G}\left[\left(\frac{\partial \rho}{\partial t}\right)^n\right]\right) . \tag{29}$$
Clearly this satisfies the semi-discrete continuity equation
$$-D\mathbf{m} + \mathcal{G}\left[\frac{\partial \rho}{\partial t}\right] = 0 \tag{30}$$
if $A = A^{\mathrm{LDG}} = DM^{-1}D^{\intercal}$. As explained by Shahbazi et al. [41], A^{LDG} is effectively a local DG discretization for a diffusion operator, so we can replace it by an SIP diffusion operator
$$A^{\mathrm{LDG}} \approx A^{\mathrm{SIP}} , \tag{31}$$
which has a smaller stencil. A consequence of using this SIP pressure matrix is that Eq. (30) does not hold exactly.

Equal-order discretizations (i.e., $\mathcal{P}_p = \mathcal{P}_m$) are known to be unstable for finite element methods. Cockburn et al. [10] addressed this problem in the context of a constant kinematic viscosity and using the A^{LDG} discretization by adding a term of the form

$$\sum_{F\in\mathcal{F}^i} \frac{\gamma_0 \|F\|_{\mathrm{leb}}}{\nu} \int_F [\![\phi]\!]\,[\![q]\!] \tag{32}$$

for basis and test functions ϕ and q to the pressure matrix. We set $\gamma_0 = 1$, and adjust the above penalty term to a variable viscosity by taking ν inside the integral, and taking the minimal value of ν on both sides of the face.

In the test cases that follow, we have found no noticeable difference between using LDG and SIP pressure discretizations, which is in line with previous findings (e.g., [25, pp. 33–45]). Interestingly, Shahbazi [40] has successfully used the SIP pressure matrix with an equal-order discretization without extra pressure stabilization [40, pp. 48–65]. Our tests (not shown here) also indicate that, for equal-order discretizations without pressure stabilization, the LDG pressure matrix is unstable for all reasonable time steps, whereas using the SIP matrix is feasible for a wide range of practical time step sizes, though it always becomes unstable in the limit $\delta t \to 0$.

Note that solving for the mass flux simplifies the pressure-correction scheme. If we had instead solved for the velocity, then the density would had to have been incorporated into the divergence operator D in Eq. (25), and into the mass matrix M in Eq. (24).

This extension of the pressure-correction method to compressible flows has sometimes proved unstable in finite difference schemes that were applied to mixing flows with large density ratios (of approximately more than a factor of 3), because the continuity equation was not satisfied in the inviscid limit; see Nicoud [30]. It is not certain whether the same instability would occur for the discontinuous Galerkin method presented here; our experience so far has not exposed instabilities with large density ratios. Nicoud suggested a different generalization of the pressure correction method to variable-density flows, where the density is incorporated into the pressure matrix, rather than on the right-hand side of Eq. (25). The large advantage of the approach presented here is that the pressure matrix is the same at all time steps. We can therefore assemble it once, and precompute the incomplete Cholesky preconditioner for the linear solver. Furthermore, the condition number of the diffusion matrix A worsens if it includes a variable coefficient that depends on the density. For these reasons, the pressure solves are much cheaper with a constant pressure matrix.

5. Linearizing the temporal derivative of the enthalpy

Solving for a primitive variable (h) with the enthalpy equation in conservative form slightly complicates the temporal derivative, because it is weighed by the temperature-dependent density. To study the stability and convergence of the time stepping scheme, we consider the space-independent enthalpy equation:

$$\frac{\mathrm{d}\,(\rho h)}{\mathrm{d}t} = -\lambda h + Q\ , \tag{33}$$

where λ is a constant, and $Q = Q\,(t)$. Using an implicit finite difference scheme, the enthalpy and the corresponding density can be estimated at a time step n by

$$\frac{\gamma_0}{\delta t}\,(\rho h)^n + \sum_{i=1}^{q} \frac{\gamma_i}{\delta t}\,(\rho h)^{n-i} = -\lambda h^n + Q^n\ . \tag{34}$$

Due to the variable density, this equation is not linear in the unknown h^n. We therefore consider two linearizations in h^n, which we term method #1 and method #2.

Both of these methods use a predictor h^* and a corresponding ρ^* that are close to h^n and ρ^n. This predictor can be obtained in several ways, such as by extrapolating from previous time steps. When solving the full system (1a)-(1c), a predictor for ρ^n can also be obtained by solving the continuity equation. The analyses in this section are for a general (h^*, ρ^*), though we will make the reasonable assumption that $(h^* - h^n)$ is at least $\mathcal{O}(\delta t)$.

The two linearization methods are as follows.

Method #1 is perhaps the most obvious approach: let $\rho^n \approx \rho^*$, resulting in an approximation $h^{[1]} \approx h^n$ that is given by

$$\frac{\gamma_0}{\delta t}\left(\rho^* h^{[1]}\right) + \sum_{i=1}^{q} \frac{\gamma_i}{\delta t}\,(\rho h)^{n-i} = -\lambda h^{[1]} + Q^n\ . \tag{35}$$

Method #2 is based on a Taylor expansion of $(\rho h)^n$ about the predictor:

$$(\rho h)^n \approx (\rho h)^* + \left(\frac{\partial(\rho h)}{\partial h}\right)^* \left(h^n - h^*\right) = \left(\frac{\partial(\rho h)}{\partial h}\right)^* h^n - \left(h^2 \frac{\partial \rho}{\partial h}\right)^*\ . \tag{36}$$

Substituting this into Eq. (34) yields an approximation $h^{[2]} \approx h^n$, given by

$$\frac{\gamma_0}{\delta t}\left(\frac{\partial(\rho h)}{\partial h}\right)^* h^{[2]} + \sum_{i=1}^{q} \frac{\gamma_i}{\delta t}(\rho h)^{n-i} = \frac{\gamma_0}{\delta t}\left(h^2 \frac{\partial \rho}{\partial h}\right)^* - \lambda h^{[2]} + Q^n \,. \tag{37}$$

Note that method #1 is effectively a single step in a fixed-point iteration, whereas method #2 is a single step in a Newton iteration.

5.1. Error estimates and stability

The errors and the stability of methods #1 and #2 can be analyzed by using a Taylor series for ρ^n about the predictor, that is,

$$\rho^n = \sum_{k=0}^{\infty} \frac{1}{k!}\left(\frac{\partial^k \rho}{\partial h^k}\right)^* \left(h^n - h^*\right)^k \,. \tag{38}$$

Define the following deviations from the non-linear finite difference equation (34):

$$\begin{aligned} &\text{error in the predictor:} && \epsilon^* := h^* - h^n \,, \\ &\text{linearization error in method \#1:} && \epsilon^{[1]} := h^{[1]} - h^n \,, \\ &\text{linearization error in method \#2:} && \epsilon^{[2]} := h^{[2]} - h^n \,. \end{aligned} \tag{39}$$

The derivations are tedious, and deferred to the Appendix. Here we summarize the main theoretical results.

The first result is an a priori error estimate. Appendix A shows that both Eq. (35) and Eq. (37) can be rewritten as

$$\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q} \frac{\gamma_i}{\delta t}(\rho h)^{n-i} = -\lambda_{\text{eff}}\, h^n + Q^n_{\text{eff}} \,, \tag{40}$$

where

$$\lambda_{\text{eff}} = \lambda + \mathcal{O}\left(\epsilon^{*2}/\delta t\right) \quad \text{and} \quad Q^n_{\text{eff}} = Q^n + \mathcal{O}\left(\epsilon^{*2}/\delta t\right) \quad \text{for method \#1,} \tag{41a}$$

and

$$\lambda_{\text{eff}} = \lambda + \mathcal{O}\left(\epsilon^{*3}/\delta t\right) \quad \text{and} \quad Q^n_{\text{eff}} = Q^n + \mathcal{O}\left(\epsilon^{*3}/\delta t\right) \quad \text{for method \#2.} \tag{41b}$$

That is, the approximations in method #1 and #2 are equivalent to the original Eq. (34), except that λ and Q^n are replaced by their effective values, which are related to the error in the predictor.

A second important result regards the stability of the linearization methods. In Appendix A.1 we show that the error for method #1 is related to the error in the predictor as

$$\left(\rho^* + \frac{\delta t}{\gamma_0}\lambda\right)\epsilon^{[1]} = \left(-\left(\frac{\partial \rho}{\partial h}\right)^* h^n\right)\epsilon^* + \mathcal{O}\left(\epsilon^{*2}\right) \,. \tag{42}$$

Note that $\epsilon^{[1]}$ vanishes up to first order in ϵ^* as $h^n \to 0$. Eq. (42) also shows that method #1 cannot always be made stable by iterating within a time step, that is, by calculating a new predictor ρ^* from the estimate $h^{[1]}$, and repeating Eq. (35). For the stability of the iteration, the error in the new approximation needs to be smaller than the error in the predictor, that is, $|\epsilon^{[1]}| < |\epsilon^*|$. Taking the limit $\delta t \to 0$ in Eq. (42), this condition is only met if

$$|h^n| < \left|\frac{\rho}{\partial \rho / \partial h}\right|^* = \left(\frac{c_p}{\beta}\right)^* \,, \tag{43}$$

where we have made the reasonable assumptions that ϵ^* is at least $\mathcal{O}(\delta t)$, and that $\partial \rho / \partial h < 0$. Note that Eq. (43) cannot be checked during a calculation, because it depends on the unknown h^n.

Similarly, iterating method #2 within a time step is stable if $|\epsilon^{[2]}| < |\epsilon^*|$. Appendix A.2 shows that

$$\left(\left(\frac{\partial(\rho h)}{\partial h}\right)^* + \frac{\delta t}{\gamma_0}\lambda\right)\epsilon^{[2]} = \left(\left(\frac{\partial \rho}{\partial h}\right)^* + \frac{1}{2}\left(\frac{\partial^2 \rho}{\partial h^2}\right)^* h^n\right)\epsilon^{*2} + \mathcal{O}\left(\epsilon^{*3}\right) \,. \tag{44}$$

Since we can reasonably expect that the error in the predictor (ϵ^*) is at least $\mathcal{O}(\delta t)$, we always have $|\epsilon^{[2]}| < |\epsilon^*|$ (i.e., the iteration is stable) in the limit $\delta t \to 0$, provided that

$$\left(\frac{\partial(\rho h)}{\partial h}\right)^{*} \neq 0\,. \tag{45}$$

In other words, the volumetric enthalpy (ρh) must be a strictly monotonic function of the specific enthalpy h.

This restriction for method #2 also follows more directly from Eq. (37) in the limit of small time steps, because the coefficient of $h^{[2]}$ cannot vanish. In practice one will want to satisfy the stronger relation

$$\frac{\partial(\rho h)}{\partial h} > 0 \tag{46}$$

to ensure that the enthalpy discretization is positive definite.

We conjecture that the stability requirements (Eq. (43) for method #1; Eq. (45) for method #2) must always be satisfied in the limit of small time steps, even when the linearization is not iterated within a time step. It seems reasonable to expect that a stable numerical method can be iterated without diverging. This is supported by the numerical tests in Section 6

5.2. Proper scaling of the enthalpy equation

Curiously, the analyses in the previous subsection have led to stability requirements (Eqs. (43), (45)) that depend on the fluid properties, and they are not satisfied for all fluids. For example, the volumetric enthalpy in supercritical fluids can either increase or decrease with the temperature due to the strong thermal expansion, thereby violating Eq. (45).

This problem can be addressed by solving for a different variable

$$\tilde{h} := h - h_0\,. \tag{47}$$

Eq. (1c) then becomes

$$h_0 R + \frac{\partial(\rho\tilde{h})}{\partial t} + \partial_l\left(m_l\tilde{h}\right) = \partial_l\left(\frac{k}{c_p}\partial_l\tilde{h}\right) + Q\,, \tag{48}$$

where $R := \partial\rho/\partial t + \partial_k m_k = 0$ is the residual of the continuity equation (1a). Thus $\tilde{h}$ satisfies the same transport equation as h, and it can therefore be discretized in the same way. This does not affect the convective or diffusive terms (since $\partial_i\tilde{h} = \partial_i h$), but it does make a difference for the temporal derivative.

In particular, the stability requirement (Eq. (43)) for method #1 becomes

$$\left|\tilde{h}^n\right| = \left|h^n - h_0\right| < \left(\frac{c_p}{\beta}\right)^{*}\,. \tag{49}$$

There is no a priori value for h_0 that guarantees stability, though it seems that h_0 is best chosen such that $h \approx h_0$ at the average temperature.

Conversely, we *can* find an a priori lower bound for h_0 when using method #2. The stability requirement (Eq. (46)) becomes

$$\frac{\partial(\rho\tilde{h})}{\partial\tilde{h}} = \frac{\partial(\rho(h-h_0))}{\partial h} = \rho + (h-h_0)\frac{1}{c_p}\frac{\partial\rho}{\partial T} > 0 \quad\Leftrightarrow\quad h_0 > h - \frac{c_p}{\beta}\,, \tag{50}$$

and so method #2 can be made unconditionally stable and SPD by choosing h_0 sufficiently large. In particular, if the temperature is known to lie in a range $[T^{\min}, T^{\max}]$, then the theoretical minimum value is

$$h_0^{\min} = \max_{T^{\min}\leq T\leq T^{\max}}\left(h - \frac{c_p}{\beta}\right)\,, \tag{51}$$

which can of course be negative. Fig. 1 shows an example of the rescaled volumetric enthalpy $(\rho(h-h_0))$ for various choices of h_0 for a real fluid. There is a practical limit on the magnitude of h_0, because we are solving a transport equation for $\rho(h-h_0)$, which becomes equivalent to the density ρ for very large values of $|h_0|$.

Of course a rescaling of the unknowns, such as in Eq. (47), is common in CFD literature, but it is usually presented as a mere numerical convenience (e.g., [33]). The above analyses show that the accuracy and stability of the numerical scheme depend critically on a proper choice of h_0. In practice this may require some trial and error, though these analyses offer useful guidelines.

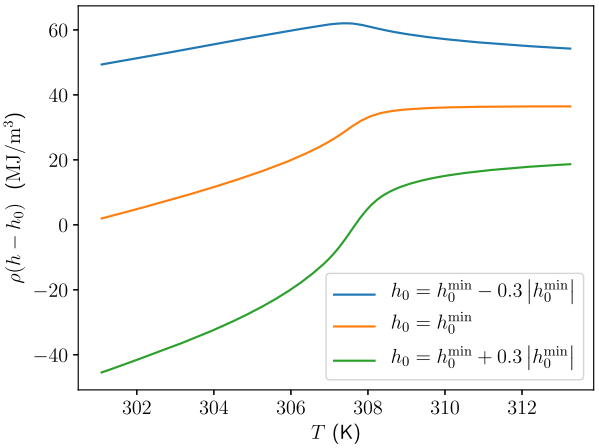


Fig. 1. Rescaled volumetric enthalpy ($\rho(h-h_0)$) of carbon dioxide at the supercritical pressure of 8 MPa, as a function of the temperature for various choices of h_0. The function is monotonic for $h_0 \geq h_0^{\min}$. The thermodynamic reference point is placed at (1 bar, 0 °C). The data are based on [45], accessed through the `CoolProp` software library [7]. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

6. Test case for the space-independent Enthalpy Equation[1]

Before solving the full system of transport equations, we clarify the theoretical results for the space-independent enthalpy equation in Section 5 with a numerical example that is based on a manufactured solution. Omitting the units of measurement, the exact temperature is

$$T^{\text{ex}}(t) = 0.5 + 0.1\sin(2\pi t) \tag{52}$$

with $0 \leq t \leq 1$. The equation of state is

$$\rho = \rho_0 T + \rho_1(1-T)\,, \tag{53}$$

and the specific heat capacity is kept constant, so that

$$h = c_p T - h_0\,. \tag{54}$$

The required source term $Q(t)$ follows from Eq. (33). For the numerical tests we let $\rho_0 = 0.5$, $\rho_1 = 2$, $c_p = 1$, and $\lambda = 0.1$. The results presented here were all obtained at $\delta t = 2^{-11}$ to investigate the limit of small time steps. We have checked that lowering the time step size to $\delta t = 2^{-14}$ does not affect whether the numerical method is stable. To make sure that rounding errors do not play a significant role with these tiny time steps, all calculations in this section were performed with 128-bit floating point precision.

The numerical schemes were tested with various orders of the BDF time stepping scheme. The predictor h^* is obtained with an s^{th}-order extrapolation from previous time steps (denoted by EXs), and the corresponding ρ^* is determined from the equation of state. Specifically,

$$h^* = \sum_{i=1}^{s} \alpha_i h^{n-i} = h^n + \mathcal{O}\left(\delta t^s\right)\,. \tag{55}$$

The weights are in Table 2.

There are two numerical errors in each time step: (i) the BDF error, which is inherent in the finite difference scheme, and (ii) the linearization error in going from Eq. (34) to either Eq. (35) or Eq. (37) when using method #1 or method #2. If the EXs coefficients are used to obtain a predictor, then the error in the predictor is $\epsilon^* := h^* - h^n = \mathcal{O}\left(\delta t^s\right)$. For method #1, Eq. (41a) then implies a linearization error of $\mathcal{O}\left(\epsilon^{*2}/\delta t\right) = \mathcal{O}\left(\delta t^{2s-1}\right)$. A BDF$q$ scheme makes an $\mathcal{O}\left(\delta t^{q+1}\right)$ error per time step, so that the overall order of accuracy is $\min(2s-1, q+1)$. Similarly, Eq. (41b) implies that the overall error per time step for method #2 is of order $\min(3s-1, q+1)$. The order of extrapolation should therefore satisfy

$$s \geq \begin{cases} (q+2)/2 & \text{for method \#1,} \\ (q+2)/3 & \text{for method \#2,} \end{cases} \tag{56}$$

[1] We have published the code for the finite difference method for the space-independent enthalpy equation on GitHub [19]. It can be used to reproduce the results in this section.

Table 2
Coefficients for extrapolation from previous time steps. (See Eq. (55)).

	α_1	α_2	α_3	α_4
EX1	1			
EX2	2	−1		
EX3	3	−3	1	
EX4	4	−6	4	−1

Table 3
Order of extrapolation for the enthalpy predictor (Eq. (55)) for the linearization methods described in Section 5. The minimum values satisfy Eq. (56); from the maximum value onward, Eq. (56) holds with strict inequality.

Finite difference coefficients	Method #1		Method #2	
	min	max	min	max
BDF1	EX2	EX2	EX1	EX2
BDF2	EX2	EX3	EX2	EX2
BDF3	EX3	EX3	EX2	EX2

or else the linearization error dominates, and the usual order of convergence of the BDF scheme cannot be achieved. If strict inequality in Eq. (56) is satisfied, then the linearization error is negligible, and increasing the order of extrapolation is pointless. Table 3 lists the range of reasonable extrapolation orders.

Fig. 2 shows the error in the numerical temperature as a function of h_0, by using method #1 (Eq. (35)). The stability requirement in Eq. (43) cannot be verified during the calculation, because it depends on the unknown h^n (i.e., the solution to the finite difference scheme without linearization error; see Eq. (34)). But h^n can be approximated by the numerical solution $h^{[1]}$, so that the stability requirement is approximately

$$\left|h^{[1]}\right| < \left(\frac{c_p}{\beta}\right)^* . \tag{57}$$

According to Table 3, the linearization error is negligible for extrapolation orders of at least 2, 3, and 3 for the BDF1, BDF2, and BDF3 schemes. For these cases the error $|h^{[1]} - h^n|$ is negligible, and the numerical scheme converges if and only if Eq. (57) is satisfied at all time steps. Note that the range of stable values for h_0 decreases with higher-order extrapolations, but all simulations converge with $h_0 = 0.5$, which is the value for which h is closest to zero.

Fig. 3 shows the equivalent error plots for method #2 (Eq. (37)). For the current equation of state, we have an explicit, a priori expression for the stability criterion in Eq. (45):

$$\frac{\partial(\rho h)}{\partial T} \neq 0 \quad \Leftrightarrow \quad h_0/c_p \neq 2T - \rho_1/(\rho_1 - \rho_0) . \tag{58}$$

The tests show that the numerical scheme is stable if and only if this criterion is satisfied everywhere in the domain, regardless of the order of the time-stepping scheme, or the order of extrapolation for the predictor. Furthermore, the results show that the minimal extrapolation orders in Table 3 need to be reached in order to achieve the lowest errors, but higher orders of extrapolation have no effect.

7. Test cases for the coupled transport equations

The numerical method for the coupled transport equations and our computer implementation are verified and validated with the following test cases. We first investigate a series of manufactured solutions for the system of transport equations, ranging from constant properties (Section 7.2), to variable transport properties (Section 7.3.1), and a variable density (Section 7.3.2). As in Section 6, we leave out the units of the variables for these manufactured solutions. Section 7.4 then shows numerical results for both constant-density and variable-density flows that can be compared to experiments.

7.1. Implementation

The simulations are performed with an in-house solver `DGFlows`. We use a hierarchical set of modal basis functions, so that there are $\binom{\mathcal{P}}{\mathcal{P}+d}$ degrees of freedom in a d-dimensional element with a polynomial order $\mathcal{P}$.

All integrals are evaluated with a quadrature set that is sufficiently accurate to negate the polynomial aliasing effect that has plagued other DG solvers. (See, e.g., [27].) The abscissa and the weights are taken from Solin et al. [44]. We store the values and derivatives of the basis functions on the quadrature set for a fast evaluation of integrals and numerical solutions. We verified that none of the results in this paper changed when the accuracy of the quadrature was increased.

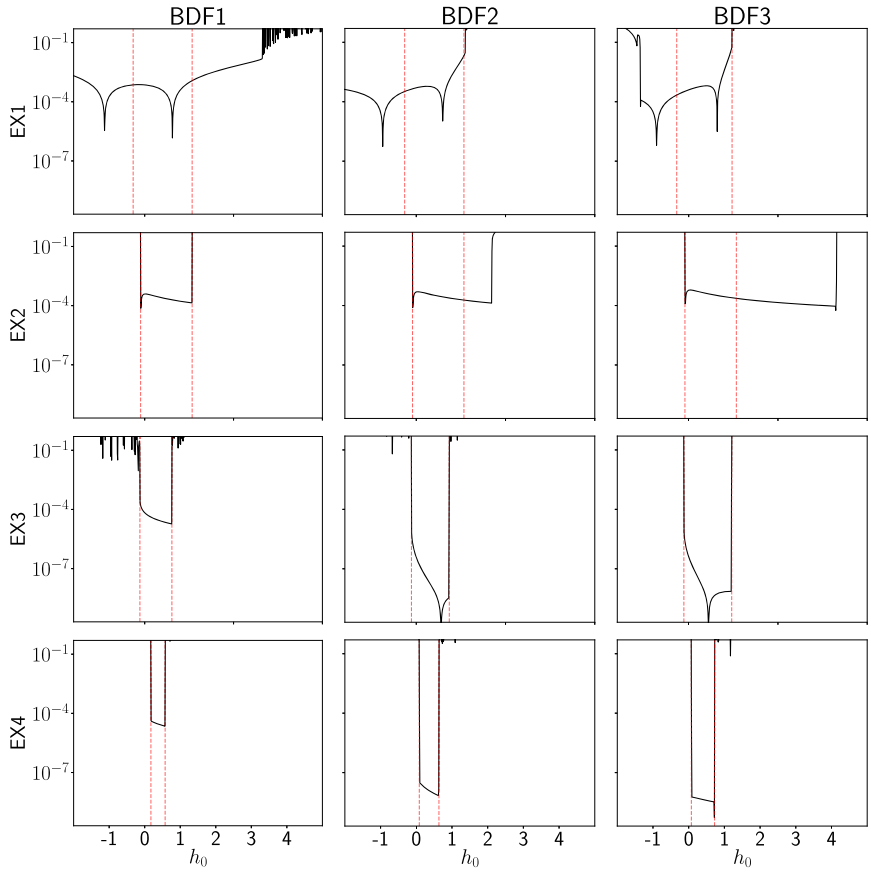


Fig. 2. Error ($|T - T^{\text{ex}}|/T^{\text{ex}}$) at $t = 1$ for the test case in Section 6 as a function of the enthalpy offset h_0, using method #1. The vertical dotted lines bound the values for which Eq. (57) held at all time steps.

All meshes were generated with the open-source software tool `Gmsh` [16]. The mesh is partitioned with METIS [22].

We use the MPI-based software library `PETSc` to solve the linear equations [3,2]. The pressure equation is solved with a conjugate gradient method, which is parallelized with an additive Schwarz preconditioner. The submatrix within each process is preconditioned with an incomplete Cholesky decomposition. The linear systems for the enthalpy and momentum equations are solved with a GMRES method, which is parallelized with an additive Schwarz preconditioner, and preconditioned with an incomplete LU decomposition within each process.

7.2. Taylor-Green Vortex

The first manufactured solution for the transport equations is the Taylor-Green Vortex, which is incompressible and has constant fluid properties. We generalize this well-known analytical solution to include a passive scalar temperature field. The enthalpy is $h = c_p T$. The exact solution is

$$\begin{aligned} u^{\text{ex}} &= \exp\left(-2\tilde{t}\right)\begin{bmatrix} -\cos\left(\tilde{x}\right)\sin\left(\tilde{y}\right) \\ +\sin\left(\tilde{x}\right)\cos\left(\tilde{y}\right) \end{bmatrix}, \\ p^{\text{ex}} &= -\frac{\rho}{4}\exp\left(-4\tilde{t}\right)\left(\cos\left(2\tilde{x}\right) + \cos\left(2\tilde{y}\right)\right), \\ T^{\text{ex}} &= \exp\left(-2\tilde{t}/\text{Pr}\right)\cos\left(\tilde{x}\right)\cos\left(\tilde{y}\right), \end{aligned} \tag{59}$$

on the domain $x, y \in [-L, L]$ with Dirichlet boundary conditions and $0 < t \leq 1$, where

$$\tilde{t} := \frac{\nu t}{\left(L/(n\pi)\right)^2}, \quad \tilde{x} := \frac{x}{L/(n\pi)}, \quad \tilde{y} := \frac{y}{L/(n\pi)}, \tag{60}$$

and n must be a positive integer in order for $\int_\Omega p = 0$. This solves the transport equations with $F = [0, 0]$ and $Q = 0$.

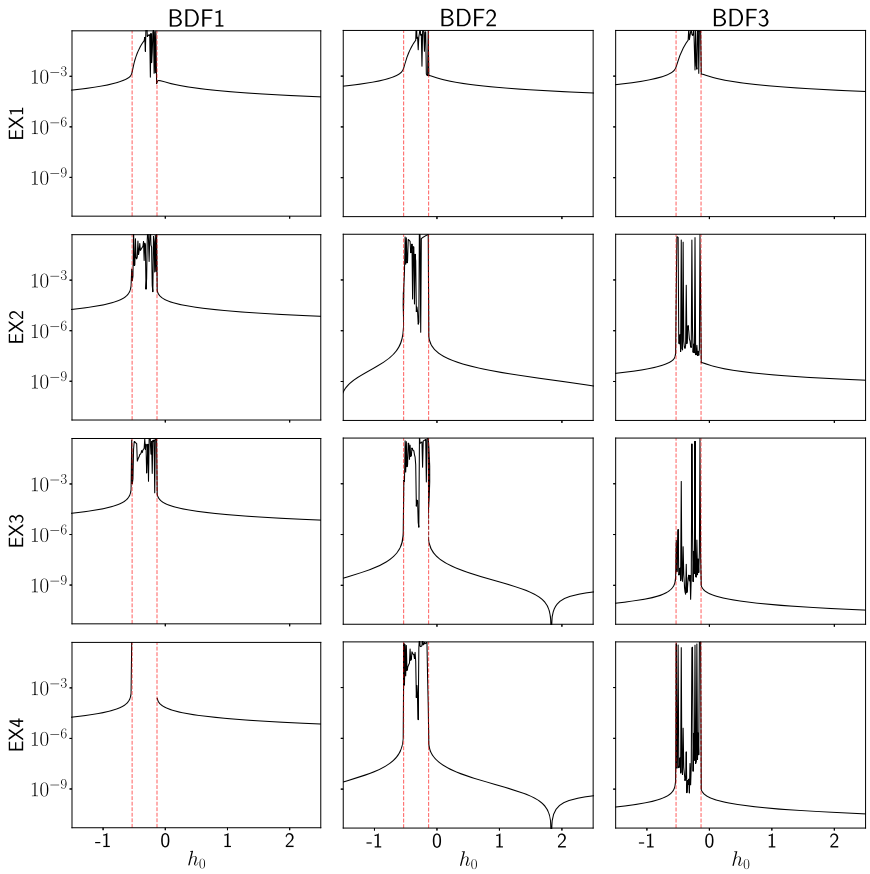


Fig. 3. Error ($|T - T^{\text{ex}}|/T^{\text{ex}}$) at $t = 1$ for the test case in Section 6 as a function of the enthalpy offset h_0, using method #2. The vertical dotted lines bound the values for which the stability criterion in Eq. (58) is violated at some time t.

For the manufactured solutions in this section and the next, the reported errors are relative in the L^2-norm, that is,

$$\frac{\|\phi - \phi^{\text{ex}}\|_2}{\|\phi^{\text{ex}}\|_2} = \sqrt{\frac{\sum_{T\in\mathcal{T}} \int_T (\phi - \phi^{\text{ex}})^2}{\int_\Omega (\phi^{\text{ex}})^2}} \tag{61}$$

for a quantity ϕ. Each integral in the numerator is evaluated with a numerical quadrature, resulting in a large sum over the squares of small numbers. We found that a naive implementation gives very large rounding errors that distort the obtained orders of convergence. We therefore perform the double summation over the elements and the quadrature points with the Kahan summation algorithm [48] and a 128-bit floating point number.

Fig. 4 shows the temporal convergence for $L = 1$, $n = 1$, $\mu = 0.01$, $\rho = 1$, $\text{Pr} = 100$, and a fourth-order polynomial space for the mass flux (i.e., $\mathcal{P}_m = 4$). We performed the same numerical experiments by independently varying the Prandtl number (to $\text{Pr} = 1$), and the polynomial order (to $\mathcal{P}_m = 2$), all of which yielded similar results. All errors saturate at small time steps, where the spatial discretization error dominates.

The Taylor-Green vortex is of course a strange test case, in that it does not feature any scale separation: its frequency spectrum is comprised of delta Dirac functions. The solution is an eigenfunction of the diffusive terms, so there is no interaction between the transport terms. Furthermore, it has Dirichlet boundary conditions where there is outflow, so that the continuity equation is over-constrained. This manifests itself in a stiff linear system for the pressure, though its ubiquity in the literature suggests that the Taylor-Green vortex is very easy to simulate. The next section features more challenging manufactured solutions.

7.3. Variable-property manufactured solutions

We present two more manufactured solutions with variable fluid properties for the system of equations (1a)-(1c): one with a constant density (in Section 7.3.1), and one with a variable density (in Section 7.3.2). The relation between T and h is the same as in Section 6, Eq. (54). The heat capacity is $c_p = 1$.

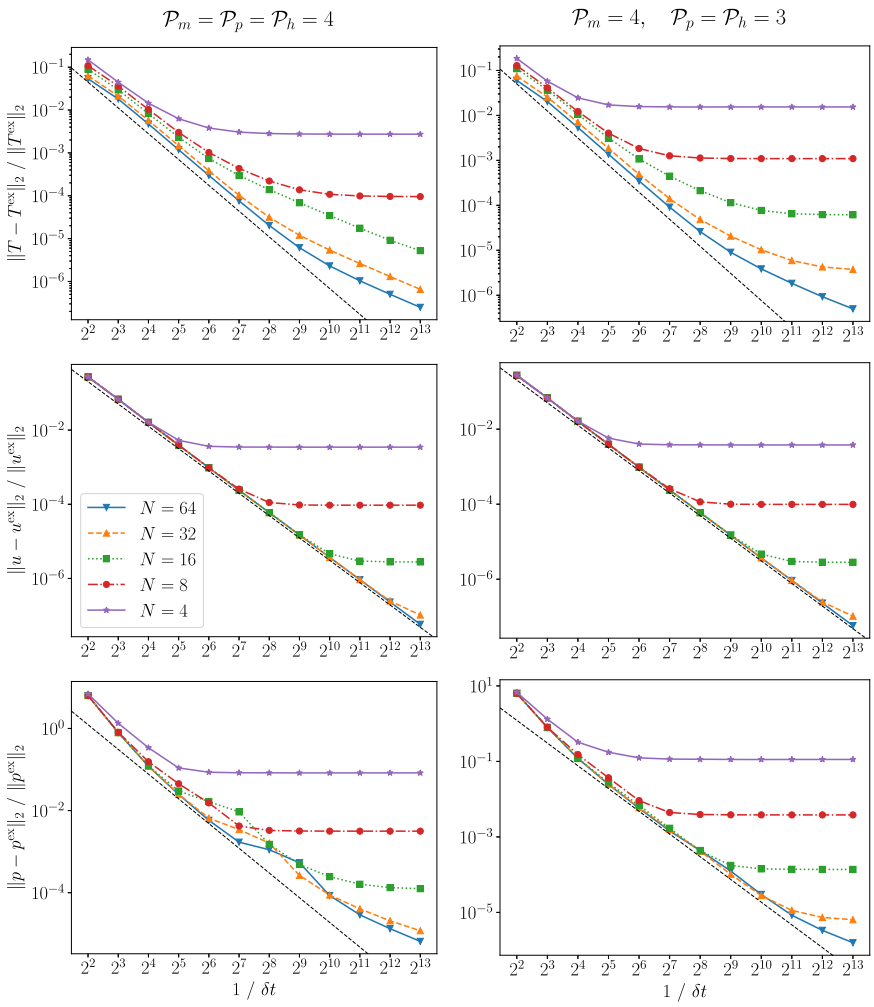


Fig. 4. Convergence toward the 2D Taylor vortex (Eq. (59)) at time $\tilde{t} = 1$ with temporal refinement for meshes with N^2 square elements. The black dashed lines indicate ideal second-order convergence in δt.

The manufactured solution is constructed by working backward from the exact mass flux and pressure, which can be chosen arbitrarily. The choice of the pressure is of little consequence, though we make sure that both m and p vary non-linearly in time, and that they do not lie in the numerical solution space. The density then follows by integrating the continuity equation over time. The temperature and the enthalpy are a function of the density. The external force and heat source can now be calculated from Eqs. (1b) and (1c), which we do symbolically with the Python SymPy library. We use polynomials for the manufactured solutions to keep these symbolic calculations feasible.

This approach differs slightly from previous literature (e.g., [43]), in that we make no attempt to find clever analytical solutions to the system (1) with a variable density, instead simply integrating the continuity equation to obtain a density that conforms to the mass flux. Some previous work (e.g., [39]) has included a source term in the continuity equation, but this appears less suitable for a time-splitting method, where the continuity plays a central role in the discretization (as in Section 5.2), and we do not want to adapt the numerical scheme to conform with the manufactured solution. Our approach permits arbitrary functions for $\mu = \mu(T)$ and $k = k(T)$. We are not aware of previous work with a manufactured solution that is (i) compressible, (ii) satisfies the unmodified continuity equation, and (iii) use temperature-dependent transport properties.

The domain is $(0, L) \times (-1, 1)$; see Fig. 5. We let $L = 10$ in all calculations, and use square elements. The inflow boundary at $x = 0$ has Dirichlet boundary conditions. The goal of the manufactured solutions is obviously not to model a particular physical phenomenon, but our configuration is reminiscent of a pipe flow with walls at $y = \pm 1$ that is heated asymmetrically, resulting in skewed density and velocity profiles.

All integrals are evaluated with a quadrature set with the usual polynomial accuracy of $(3\mathcal{P}_m - 1)$, and we verified that this is sufficient to integrate up to machine precision by comparing the results with a higher-order quadrature set of polynomial accuracy $(3\mathcal{P}_m + 10)$.

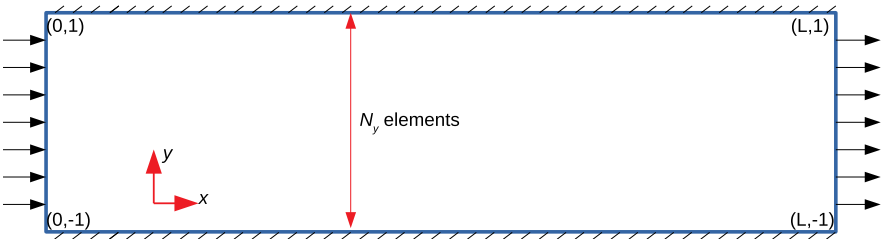
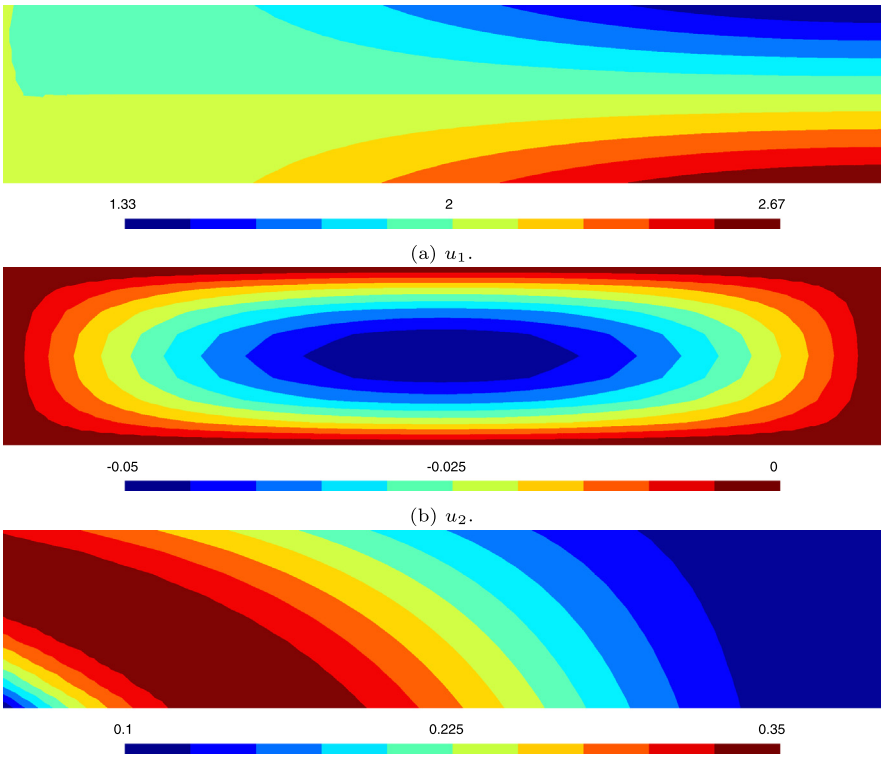


Fig. 5. Domain of the manufactured solutions in Section 7.3.



(a) u_1.

(b) u_2.

(c) Kinematic viscosity (ν) and thermal diffusivity (α). (Note that $\mathrm{Pr} = 1$, so $\alpha = \nu$.)

Fig. 6. Constant-density manufactured solution in Eq. (62) at $t = 1$.

We use a Neumann boundary condition for the temperature at the outlet, because this is the most common choice in practical applications. The imposed heat flux (q^{N}) follows from the known exact solution. We also tried imposing a Dirichlet boundary condition for the temperature, and found that it makes a negligible difference in the numerical errors.

7.3.1. Constant-density, variable-property manufactured solution

We first investigate the temporal convergence for a case where the transport properties (μ, k) depend on the temperature, but the density is constant. We choose the polynomial manufactured solution

$$\begin{aligned} m^{\mathrm{ex}} &= (1+t^3)\frac{1}{L^3}\begin{bmatrix} y\left(2x^3/3 - Lx^2\right) \\ (y-1)(y+1)x(L-x) \end{bmatrix} + \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \\ p^{\mathrm{ex}} &= (1+t^3)\,(L-x)^3\,, \\ T^{\mathrm{ex}} &= (1+t^3)\,(2-y)((x-L)/L)^2/6 \end{aligned} \tag{62}$$

with $0 < t \leq 1$, which satisfies $m_2 = 0$ on $y = \pm 1$, and $\partial_k m_k = 0$. The addition of the constant $[2, 0]$ to m^{ex} ensures that $m_1 > 0$ everywhere, so that there is no backflow at the outlet, and no outflow at any of the Dirichlet boundary conditions. The density is $\rho = 1$, but the transport properties are non-trivial: $\mu = 0.1 + T(1-T)$ and $k = \mu c_p/\mathrm{Pr}$, with $\mathrm{Pr} = 1$. The solution is depicted in Fig. 6.

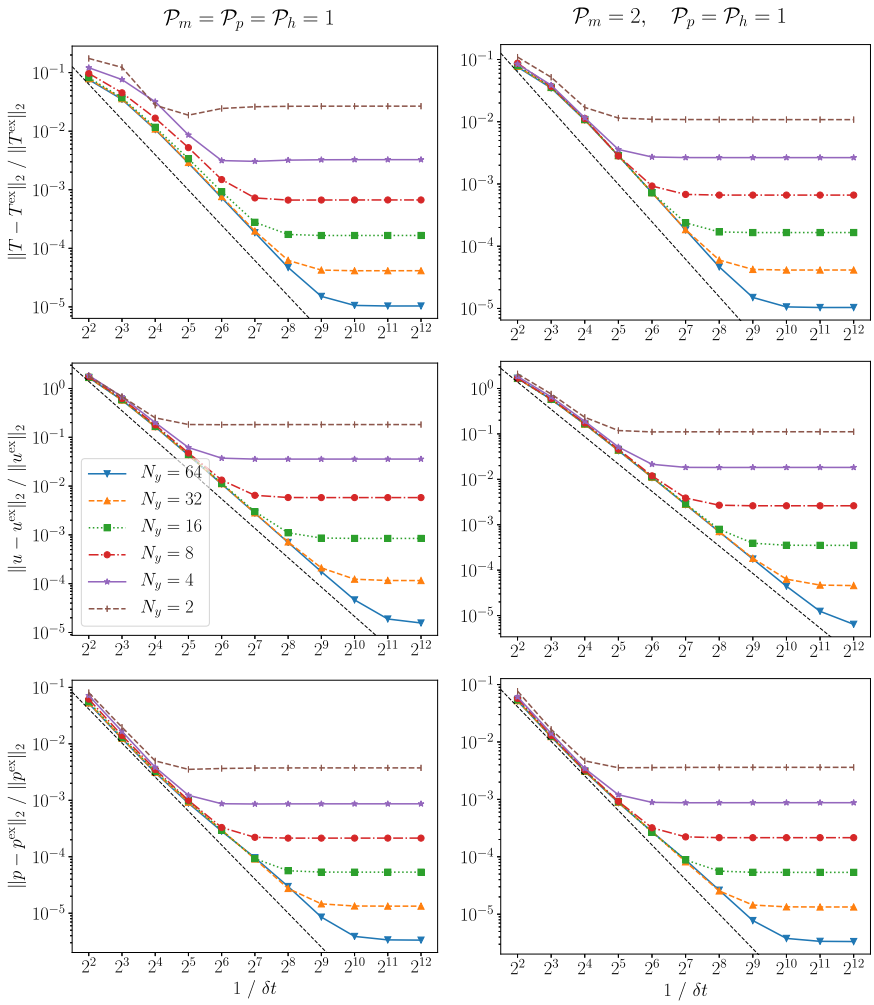


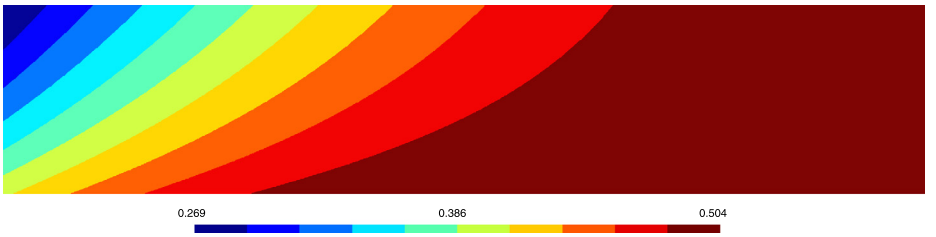
Fig. 7. Convergence of the numerical solution toward the constant-density manufactured solution (Eq. (62)) with temporal refinement. The characteristic element length is inversely proportional to N_y. The black dashed lines indicate ideal second-order convergence in δt.

Table 4
Convergence toward the constant-density manufactured solution in Eq. (62) (Fig. 6) with spatial refinement for the mixed-order case ($\mathcal{P}_m = 2$ and $\mathcal{P}_h = \mathcal{P}_p = 1$) with $\delta t = 2^{-12}$.

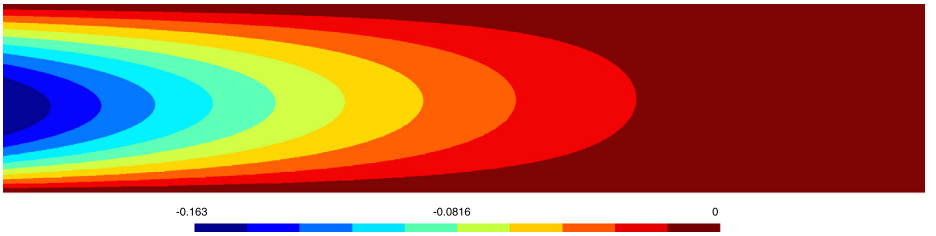
N_y	Error T	Conv. T	Error u	Conv. u	Error p	Conv. p
2^1	1.08e-2	-	1.11e-1	-	3.62e-3	-
2^2	2.65e-3	2.03	1.83e-2	2.61	8.75e-4	2.05
2^3	6.63e-4	2.00	2.62e-3	2.80	2.15e-4	2.02
2^4	1.66e-4	2.00	3.52e-4	2.90	5.35e-5	2.01
2^5	4.15e-5	2.00	4.57e-5	2.94	1.34e-5	2.00
2^6	1.04e-5	2.00	6.43e-6	2.83	3.34e-6	2.00

Fig. 7 displays the temporal convergence. We consider various meshes, varying the number of elements in the y-direction (N_y), and the spatial polynomial orders. The equal-order case does not appear to suffer from the inf-sup instability for small δt. The velocity and the temperature converge with second-order accuracy in δt until the error saturates when the spatial error starts to dominate the temporal error. The order of convergence for the pressure is slightly lower, in the range [1.5,2.0]. These orders of convergence for the velocity and pressure agree with what is found in previous literature on constant-property incompressible flows (e.g., [41], [15]).

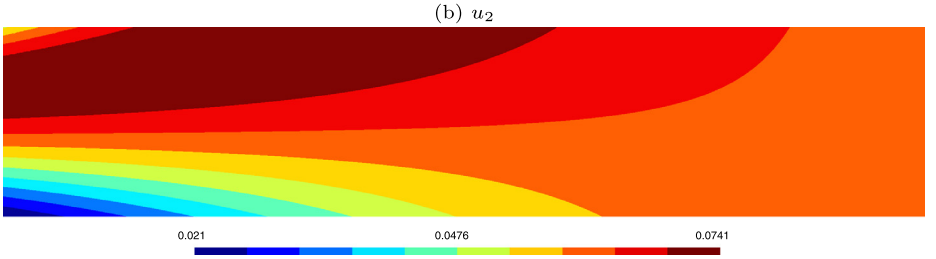
The spatial rates of convergence are in Table 4. As the mesh is refined, all quantities with polynomial order $\mathcal{P}$ converge with $\mathcal{O}\left(\ell^{\mathcal{P}+1}\right)$, where $\ell \propto 1/N_y$ is the characteristic mesh length. The convergence rate of the velocity saturates at high spatial refinement, as the temporal error starts to become significant.



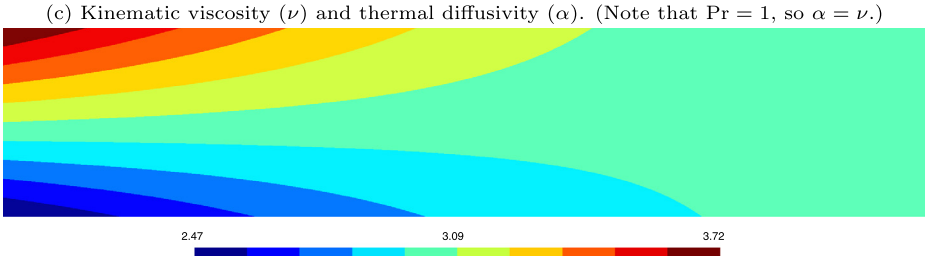
(a) u_1



(b) u_2



(c) Kinematic viscosity (ν) and thermal diffusivity (α). (Note that $\mathrm{Pr} = 1$, so $\alpha = \nu$.)



(d) ρ

Fig. 8. Variable-density manufactured solution in Eqs. (63)-(64) at $t = 1$.

7.3.2. Variable-density manufactured solution

The next manufactured solution features a transient density. The exact solution is

$$\begin{aligned} m^{\mathrm{ex}} &= \frac{1}{4}\left(1+t^3\right)\begin{bmatrix} (x/L-1)^3 \\ (x/L-1)^2\,(y-1)(y+1) \end{bmatrix} + \begin{bmatrix} 3/2 \\ 0 \end{bmatrix}, \\ p^{\mathrm{ex}} &= \left(1+t^3\right)(L-x)^3 \end{aligned} \tag{63}$$

with $0 < t \leq 1$. The density is determined by integrating $(-\partial_k m_k)$ over t, to find

$$\rho^{\mathrm{ex}} = -\frac{1}{4}\left(\frac{1}{4}t^4 + t\right)\,(x/L-1)^2(2y+3/L)\;+\;3\,, \tag{64}$$

where the addition of the constant 3 ensures that $\rho > 0$ everywhere. We use a non-affine equation of state: $T = ((\rho_1 - \rho)/(\rho_1 - \rho_0))^2$, where $\rho_0 = 2$ and $\rho_1 = 4$ are lower and upper bounds for ρ, so that the temperature is between 0 and 1. As in Section 7.3.1, the viscosity and conductivity are $\mu = 0.1 + T(1-T)$ and $k = \mu c_p/\mathrm{Pr}$ with $\mathrm{Pr} = 1$. The solution is depicted in Fig. 8.

We base the enthalpy time-stepping scheme on a linearization of $(\rho h)^n$ about a predictor for h^n, that is, method #2 in Section 5. We let $h_0 = 0.2$, so that Eq. (46) is satisfied everywhere. Increasing h_0 had no noticeable effect. The predictor h^* is obtained with a second-order extrapolation from previous time steps (using the EX2 weights in Table 2). We found that

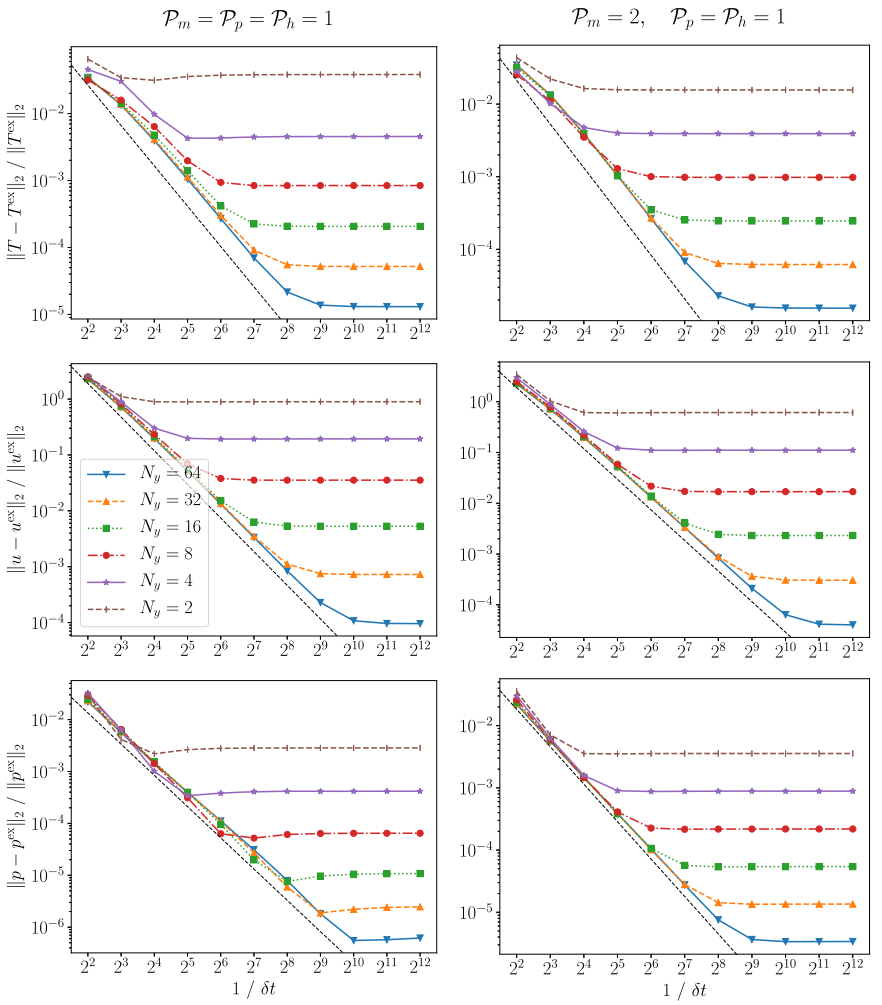


Fig. 9. Convergence of the numerical solution toward the variable-density manufactured solution (Eq. (63)) with temporal refinement, using method #2 with $h_0 = 0.2$. The characteristic element length is inversely proportional to N_y. The black dashed lines indicate ideal second-order convergence in δt.

increasing the extrapolation order for h^* had no noticeable effect on the stability or the errors, which is in line with the tests for the BDF2 scheme in Fig. 3.

Fig. 9 shows the convergence with temporal refinement. The velocity and the temperature converge with second order, just like for the constant-density results in Fig. 7, though in this case the pressure also shows $\mathcal{O}\left(\delta t^2\right)$ behavior. Note that the mixed-order cases remain fully stable, even for very small δt, despite the possible small-δt instability for variable-density flows that is discussed in Section 5.

Fig. 10 shows two other examples of temporal convergence with a mixed-order scheme, but with less effective enthalpy treatments. We found that method #1 was stable for all $h_0 \geq 0$, and that the precise value of h_0 is of little consequence to the L2 errors in the final answer. It is clear that the temperature does not converge with second order when method #1 is used for the enthalpy treatment. The right column in Fig. 10 shows that method #2 can also be unstable when the enthalpy offset is not sufficiently large (here it is 0.0 instead of 0.2): all quantities still converge with second-order accuracy, but the error diverges at small time steps. Some calculations failed due to numerical backflow at the outlet, which our simple outlet boundary condition cannot handle.

Table 5 collects the spatial convergence rates for mixed-order case with method #2 and $h_0 = 0.2$. As the mesh is refined, all quantities with polynomial order $\mathcal{P}$ converge with $\mathcal{O}\left(\ell^{\mathcal{P}+1}\right)$.

7.4. Flow past a circular cylinder

As a final test case, we computed laminar flow past a circular cylinder to validate our numerical method. This features a Von Kármán vortex street in the wake of the obstacle. The results of this well-known benchmark case can be compared to experiments and to other direct numerical simulations.

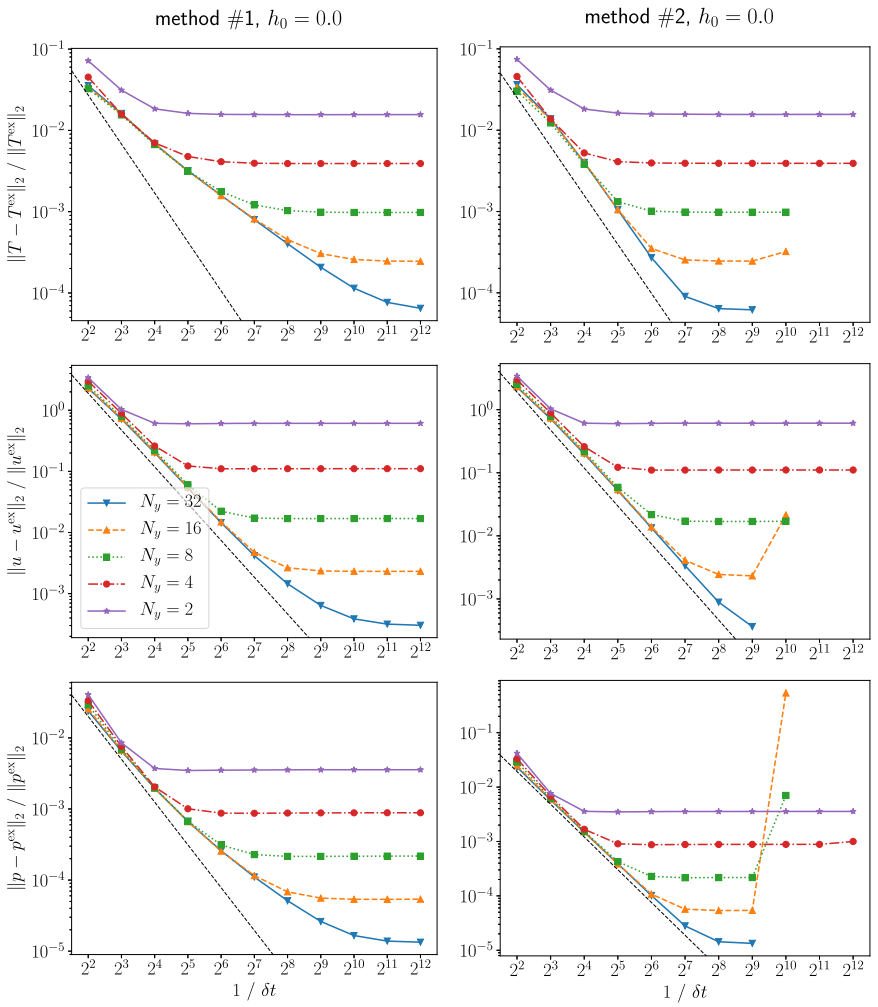


Fig. 10. Equivalent of the mixed-order case ($\mathcal{P}_m = 2$, $\mathcal{P}_p = \mathcal{P}_h = 1$) in Fig. 9, but with a different value of h_0, and comparing method #1 to method #2. The missing values in the figures in the right column indicate failed calculations.

Table 5
Convergence toward the variable-density manufactured solution in Eqs. (63)–(64) (Fig. 8) with spatial refinement for the mixed-order case ($\mathcal{P}_m = 2$ and $\mathcal{P}_h = \mathcal{P}_p = 1$) with $\delta t = 2^{-12}$.

N_y	Error T	Conv. T	Error u	Conv. u	Error p	Conv. p
2^1	1.56e-2	-	6.11e-1	-	3.57e-3	-
2^2	3.92e-3	2.00	1.11e-1	2.47	8.86e-4	2.01
2^3	9.80e-4	2.00	1.69e-2	2.71	2.18e-4	2.02
2^4	2.46e-4	2.00	2.32e-3	2.87	5.43e-5	2.01
2^5	6.17e-5	1.99	3.05e-4	2.92	1.36e-5	2.00
2^6	1.55e-5	1.99	4.03e-5	2.92	3.39e-6	2.00

Fig. 11 shows the computational domain. Denoting far-field values by a subscript ∞, the Dirichlet boundary values for the velocity and temperature are $[u_\infty, 0]$ and T_∞ on the left, top, and bottom parts of the domain. The right side (at $x = L$) is an outlet with homogeneous Neumann boundary conditions (i.e., $q^{\text{N}} = 0$ and $f_i^{\text{N}} = 0$). The cylinder has a no-slip boundary condition for the velocity, and a constant temperature T_{w}. We use the far-field values to define a Reynolds number $\text{Re} := D\,(\rho u/\mu)_\infty$ and Prandtl number $\text{Pr} := \left(\mu c_p/k\right)_\infty$.

The initial condition requires special care. First, it is not easy to find an initial velocity field that satisfies $n \cdot u = n \cdot u^{\text{D}}$ at the walls, inlet, and cylinder. We therefore initialize the velocity to $[u_\infty, 0]$, and leave out the convective term in the first 10 time steps, thus essentially simulating Stokes flow, which does not have the same requirement for the initial condition at the boundaries. Due to the instant smoothing property of the viscous operator, the velocity satisfies the Dirichlet boundary conditions when the convective term is 'activated' after the first 10 time steps. Second, we found that the combination of

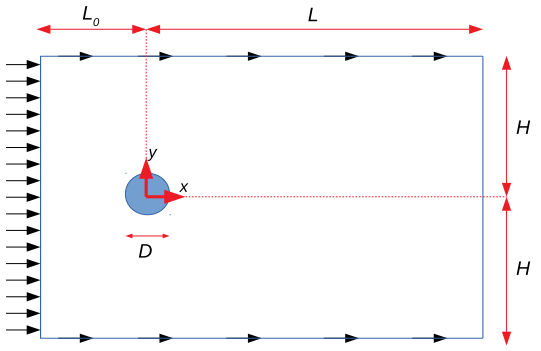


Fig. 11. Geometry of flow past a circular cylinder (not to scale). The cylindrical obstacle is centered at the origin.

a symmetrical mesh and a symmetrical initial condition does not induce vortex shedding. This is addressed by letting the cylinder rotate counter-clockwise for the first 100 time steps.

We are interested in the force on the cylinder S, which is given by

$$F_i^{(\mathrm{cyl})} = \int_{\partial S} (p\,\delta_{ik} - \tau_{ik})\, n_k \,, \tag{65}$$

where n is the outward normal of the fluid, pointing into the cylinder. The drag and the lift coefficients are

$$C_\mathrm{D} = \frac{2\,F_1^{(\mathrm{cyl})}}{D\left(\rho u^2\right)_\infty} \quad \text{and} \quad C_\mathrm{L} = \frac{2\,F_2^{(\mathrm{cyl})}}{D\left(\rho u^2\right)_\infty} \tag{66}$$

respectively. At our Reynolds number, the flow is laminar, and the force oscillates in time in a smooth, deterministic manner. This makes it easy to determine the frequency f of the lift coefficient, and the corresponding Strouhal number $\mathrm{St} := fD/u_\infty$.

Our experiments indicate that a small domain results in an overestimation of the Strouhal number. This likely explains the large discrepancy in the numerical predictions of the Strouhal number in previous literature; see Niroobakhsh et al. [32] for an overview. Collis [11] showed that a domain of $(H, L_0, L) = (30D, 15D, 30D)$ was sufficient for isothermal flow at $\mathrm{Re} = 100$ and a Mach number of 0.2. The results presented here were obtained on a domain of $(H, L_0, L) = (40D, 20D, 40D)$.

7.4.1. Isothermal vortex shedding

We first simulate isothermal flow (i.e., $T_\mathrm{w} = T_\infty$). Fig. 12 shows the mesh. It has approximately 27k elements with $\mathcal{P}_m = 2$ and $\mathcal{P}_p = \mathcal{P}_h = 1$, resulting in approximately 160k degrees of freedom per direction of the mass flux, and 64k degrees of freedom for p and h. There are 120 boundary elements at the cylinder, each with a width of $0.005D$. The time step is given by $D/(\delta t\, u_\infty) = 82$. We find $\mathrm{St} = 0.166$. This number remained unchanged when we decreased the number of elements to 19k, or when we doubled δt. It compares well with the experimental values of $\mathrm{St} = 0.165$ in [47], $\mathrm{St} = 0.165$ in [49], and $\mathrm{St} = 0.167$ in [38, p. 71].

7.4.2. Heated vortex shedding

To validate our numerical method with a variable-density flow, we replicate a numerical test case by Shi et al. [42], who used a specialized cylindrical finite volume scheme to handle the circular geometry. The temperatures are $T_\infty = 20\,^\circ\mathrm{C} = 293.15$ K and $T_\mathrm{w} = 1.5T_\infty = 166.575\,^\circ\mathrm{C}$, resulting in $\mathrm{Pr} = 0.7146$. They solved for the temperature, approximating the material properties as

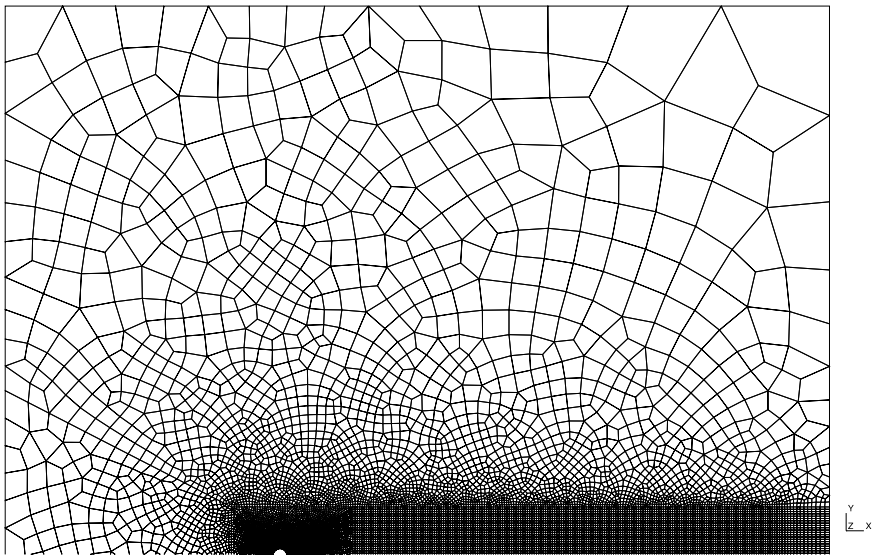
$$\phi = a_0 + a_1(T - T_F) + a_2(T - T_F)^2 \,, \tag{67}$$

where ϕ is one of (ρ, μ, k, c_p), $T_F = 0\,^\circ\mathrm{C}$, and the coefficients a_i are in Table 6. Since $c_p := \partial h/\partial T$ is a second-order polynomial in T, we need to find the root of the third-order polynomial $h = h(T)$ to map from h to a fluid property. This minor inconvenience permits a better comparison with the results in [42].

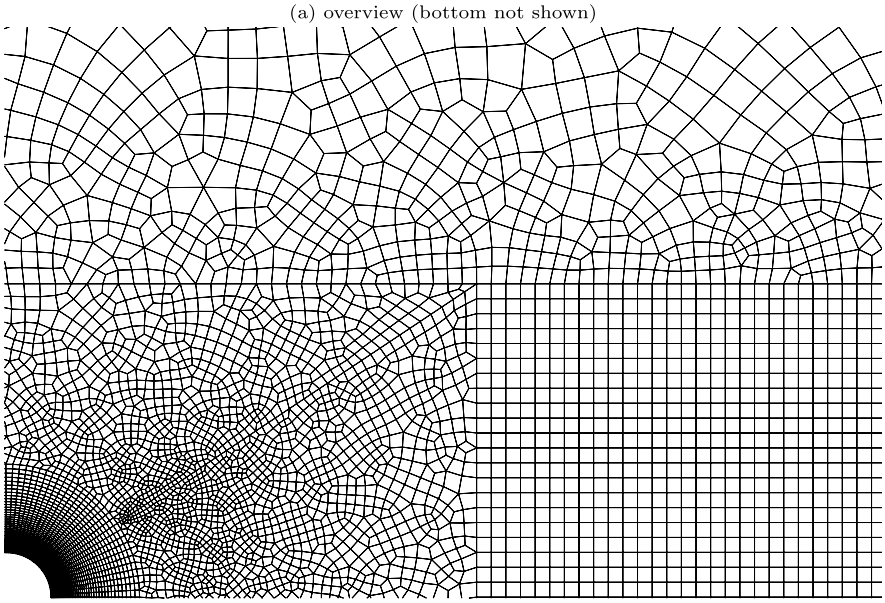
We obtain our results on the same mesh as for the isothermal case (Fig. 12). Fig. 13 shows an example of instantaneous flow fields. Fig. 14 shows the lift and drag coefficients and the Nusselt number, which is defined as

$$\mathrm{Nu} = \frac{D}{T_\mathrm{w} - T_\infty} \frac{1}{\|\partial S\|_\mathrm{leb}} \int_{\partial S} n_k \partial_k T \,, \tag{68}$$

where $\|\partial S\|_\mathrm{leb} = \pi D$ is the circumference of the circular obstacle. We find $\mathrm{St} = 0.1536$, which differs by 1% from the value of $\mathrm{St} = 0.152$ in Shi et al. [42] and the experimental value of $\mathrm{St} = 0.152$ in [47].



(a) overview (bottom not shown)



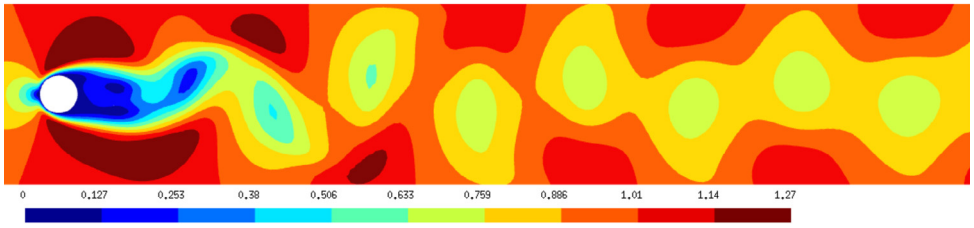
(b) Detail near the cylinder.

Fig. 12. Mesh for flow past an obstacle. It is structured near the cylinder and most of its wake. The rest of the mesh is unstructured to allow for large differences in the element size. It is symmetrical about the axis $y = 0$.

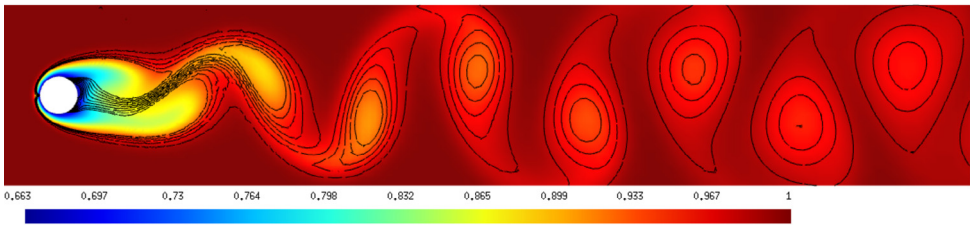
Table 6
Coefficients for the material properties in Eq. (67). Reproduced from [42].

	ρ (kg m^{-3})	μ (kg m^{-1} s^{-1})	k (W m^{-1} K^{-1})	c_p (m^2 s^{-2} K^{-1})
a_0	1.268672727	1.7254e-05	2.4195e-02	1.00620979e03
a_1 (K^{-1})	−4.08741e-03	4.95611e-08	7.5234e-05	1.4522145e-02
a_2 (K^{-2})	7.23864e-06	−2.7214e-11	−3.2588e-08	4.13753e-04

This result was obtained by linearizing $(\rho h)^n$ with method #2 for the temporal derivative of the enthalpy. We subtracted an offset h_0 from the enthalpy (as explained in Section 5.2), such that the maximum value of h was zero. Interestingly, there is no noticeable change in the results when we use an equal-order scheme, even when δt is decreased by a factor of 10 or 100. For the mixed-order calculations, we found no difference between using an SIP or an LDG pressure matrix. We repeated the calculation using method #1, setting the enthalpy offset to a value h_0^*, such that $h = 0$ at $T = (T_w + T_\infty)/2$. This resulted in almost exactly the same shedding frequency (St = 0.1537).



(a) $|m|/(\rho u)_\infty$



(b) Colored field of ρ/ρ_∞, overlayed with isolines of $\mathrm{curl}(u)D/u_\infty$.

Fig. 13. Instantaneous fields for flow past a heated circular cylinder.

For all test cases, we found that some values for the enthalpy offset result in unstable schemes, yielding oscillatory pressure fields. For method #1 this happens when h_0 is far from h_0^*; for method #2 this happens when h_0 is too small. When operating in the range of stable h_0 values, the exact enthalpy offset has no noticeable impact on the shedding frequency.

8. Discussion and conclusions

We have presented a discontinuous Galerkin method for solving the system of transport equations in their conservative form for low-Mach number flows, while taking the mass flux m, the pressure p, and the specific enthalpy h as the unknowns.

Since the density is a function of the specific enthalpy, the temporal finite difference scheme requires that the volumetric enthalpy (ρh) be linearized in h, and we have analyzed two methods for doing this, both of which need a predictor for the enthalpy at the new time step. This led to theoretical stability requirements in case the enthalpy equation is iterated within a time step, or, equivalently, in the limit of small time steps. The specific enthalpy is shifted with an offset h_0, so that solving for the new unknown $h - h_0$ satisfies these stability requirements. These results were verified with simple space-independent tests in Section 6.

Method #1 is basically what has been done in all previous literature we have seen (e.g., [29], [33], [24]), but for this approach we cannot determine the range of stable h_0 values a priori. Another disadvantage is that full accuracy with a second-order time-stepping scheme (BDF2) can only be achieved if the predictor is third-order accurate. This necessitates storing three instead of two previous time steps for the enthalpy, and the third-order extrapolation negatively affects the stability (see Fig. 2).

The other linearization of (ρh) (method #2) does provide full accuracy when the predictor is extrapolated from two previous time steps, in which case the error in the linearization is negligible in the limit of small time steps. Furthermore, the stability requirement is simply that the volumetric enthalpy be a monotonic function of the temperature (or, equivalently, the specific enthalpy), and this leads to a range of stable h_0 values that can be determined a priori. This results in a stable scheme, in which the error of the linearization of (ρh) becomes negligible in the limit of small time steps. The manufactured solutions demonstrate full second-order temporal accuracy, without any predictor steps.

In a real flow simulation the exact range of stable values for h_0 cannot be determined a priori due the coupling of the transport equations. Note that a discontinuous Galerkin discretizations does not guarantee that the extreme values of the numerical solution lie within physically acceptable bounds, so that the numerical temperature range is not known beforehand. Nevertheless, there are useful guidelines: for method #1, $h - h_0$ should be close to zero; for method #2, h_0 should be sufficiently large. Once a stable h_0 value is found, our numerical experiments in Section 7.3 and 7.4 suggest that the exact value of h_0 has little bearing on the overall accuracy.

The numerical method presented here can form the basis for simulations of non-laminar, three-dimensional flow with strongly varying properties. In future work the first author will explore this in combination with a large eddy simulation (LES) model.

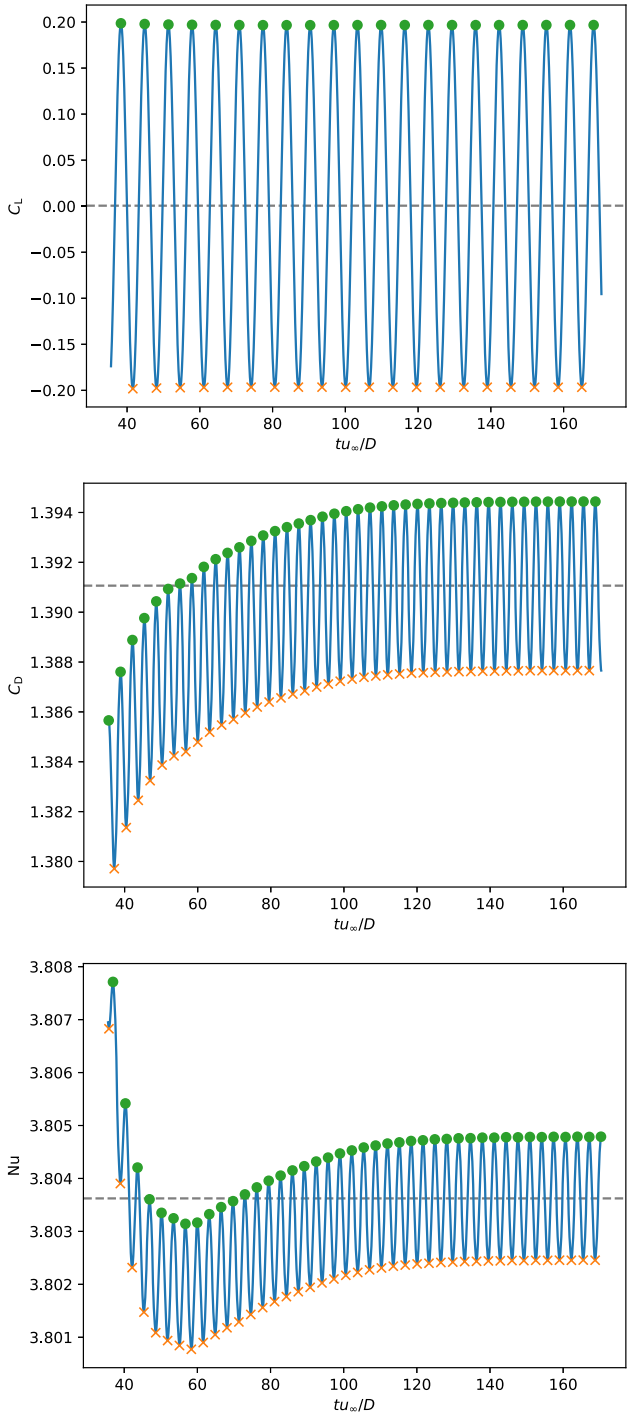


Fig. 14. Temporal behavior of the lift and drag coefficients (Eqs. (66)), and Nusselt number (Eq. (68)) for flow past a heated circular cylinder. The crosses and circles mark local minima and maxima. The dashed horizontal lines indicate the averages over the last five periods.

CRediT authorship contribution statement

Aldo Hennink: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing - original draft, Writing - review & editing. **Marco Tiberga:** Methodology, Software, Writing - review & editing. **Danny Lathouwers:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We are grateful to the sCO_2-HeRo project that has received funding from the European research and training program 2014-2018 (grant agreement ID 662116) for funding the first author.

Appendix A. Derivation of error estimates in Section 5.1

Here we derive the error estimates that were presented in Section 5.1. To ease the notation, let

$$K := \left(\frac{\partial \rho}{\partial h}\right)^* \quad \text{and} \quad B := \frac{1}{2}\left(\frac{\partial^2 \rho}{\partial h^2}\right)^*, \tag{69}$$

so that the Taylor expansion in Eq. (38) becomes

$$\rho^n = \rho^* - K\epsilon^* + B\epsilon^{*2} + \mathcal{O}\left(\epsilon^{*3}\right). \tag{70}$$

A.1. Derivations for Method #1

To derive Eq. (42), subtract Eq. (34) from Eq. (35) to get

$$\rho^* h^{[1]} - \rho^n h^n = -\frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[1]}. \tag{71}$$

Substituting Eq. (70) gives

$$\rho^*\epsilon^{[1]} + h^n\left(K\epsilon^* + \mathcal{O}\left(\epsilon^{*2}\right)\right) = -\frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[1]}, \tag{72}$$

which is indeed equivalent to Eq. (42).

This result can now be used to derive the error estimates in Eq. (41a). Using the fact that $h^{[1]} = h^n + \epsilon^{[1]}$, we can write

$$\rho^* h^{[1]} = \rho^n h^n - \left(\rho^n - \rho^*\right)h^n + \rho^*\epsilon^{[1]}, \tag{73}$$

which, upon substitution into Eq. (35), gives

$$\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} = -\left(\lambda - \frac{\gamma_0}{\delta t}\left(\rho^n - \rho^*\right)\right)h^n + Q^n - \frac{\gamma_0}{\delta t}\left(\rho^* + \frac{\delta t}{\gamma_0}\lambda\right)\epsilon^{[1]}. \tag{74}$$

Using the error estimate in Eq. (42),

$$\begin{aligned}\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} &= -\left(\lambda - \frac{\gamma_0}{\delta t}\left(\rho^n - \rho^* + \left(\frac{\partial \rho}{\partial h}\right)^*\epsilon^*\right)\right)h^n + Q^n + \mathcal{O}\left(\epsilon^{*2}/\delta t\right)\\ &= -\left(\lambda + \mathcal{O}\left(\epsilon^{*2}/\delta t\right)\right)h^n + Q^n + \mathcal{O}\left(\epsilon^{*2}/\delta t\right),\end{aligned} \tag{75}$$

where the second equality follows from the Taylor expansion in Eq. (38).

A.2. Derivations for Method #2

In analogy with the previous subsection, Eq. (44) is derived by subtracting Eq. (34) from Eq. (37) to get

$$\left(\frac{\partial(\rho h)}{\partial h}\right)^* h^{[2]} - \rho^n h^n = \left(h^2\frac{\partial \rho}{\partial h}\right)^* - \frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[2]}. \tag{76}$$

Upon substituting Eq. (70), this becomes

$$\left(\frac{\partial(\rho h)}{\partial h}\right)^* h^{[2]} - h^n\rho^* + Kh^n\epsilon^* - Bh^n\epsilon^{*2} + \mathcal{O}\left(\epsilon^{*3}\right) = \left(h^2\frac{\partial \rho}{\partial h}\right)^* - \frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[2]}. \tag{77}$$

Substituting $\partial(\rho h)/\partial h = \rho + Kh$ on the left-hand side, and $(h^2(\partial\rho/\partial h))^* = Kh^{*2} = Kh^*\left(h^n + \epsilon^*\right)$ on the right-hand side, gives

$$(\rho + Kh)^* h^{[2]} - h^n\rho^* + Kh^n\epsilon^* - Bh^n\epsilon^{*2} + \mathcal{O}\left(\epsilon^{*3}\right) = Kh^*\left(h^n + \epsilon^*\right) - \frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[2]}\,. \tag{78}$$

This can be rearranged to

$$\rho^*(h^{[2]} - h^n) + Kh^*(h^{[2]} - h^n) - K(h^* - h^n)\epsilon^* - Bh^n\epsilon^{*2} + \frac{\delta t}{\gamma_0}\lambda\,\epsilon^{[2]} = \mathcal{O}\left(\epsilon^{*3}\right)\,. \tag{79}$$

Using the definitions for $\epsilon^{[2]}$ and ϵ^*, this becomes

$$\left(\rho + Kh + \frac{\delta t}{\gamma_0}\lambda\right)^* \epsilon^{[2]} - \left(K + Bh^n\right)\epsilon^{*2} = \mathcal{O}\left(\epsilon^{*3}\right)\,, \tag{80}$$

which is indeed equivalent to Eq. (44).

This result can now be used to derive the error estimates in Eq. (41b). Upon substituting

$$\left(\frac{\partial(\rho h)}{\partial h}\right)^* h^{[2]} = \rho^* h^{[2]} + Kh^* h^{[2]} = \rho^n h^n - \left(\rho^n - \rho^*\right)h^n + \rho^*\epsilon^{[2]} + Kh^* h^{[2]}\,, \tag{81}$$

Eq. (37) becomes

$$\begin{aligned}
&\frac{\gamma_0}{\delta t}\left(\rho^n h^n - \left(\rho^n - \rho^*\right)h^n + \rho^*\epsilon^{[2]} + Kh^* h^{[2]}\right) + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} \\
&= \frac{\gamma_0}{\delta t}Kh^{*2} - \lambda h^{[2]} + Q^n \\
&= \frac{\gamma_0}{\delta t}Kh^*\left(h^{[2]} - \left(\epsilon^{[2]} - \epsilon^*\right)\right) - \lambda\left(h^n + \epsilon^{[2]}\right) + Q^n\,,
\end{aligned} \tag{82}$$

which can be rearranged to

$$\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} = \frac{\gamma_0}{\delta t}\left(\left(\rho^n - \rho^*\right)h^n - \rho^*\epsilon^{[2]} + Kh^*\left(\epsilon^* - \epsilon^{[2]}\right) - \frac{\delta t}{\gamma_0}\lambda\epsilon^{[2]}\right) - \lambda h^n + Q^n\,. \tag{83}$$

Eq. (80) can now be used to eliminate $\epsilon^{[2]}$:

$$\begin{aligned}
&\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} \\
&= \frac{\gamma_0}{\delta t}\left(\left(\rho^n - \rho^*\right)h^n + Kh^*\epsilon^* - \left(K + Bh^n\right)\epsilon^{*2} + \mathcal{O}\left(\epsilon^{*3}\right)\right) - \lambda h^n + Q^n \\
&= \frac{\gamma_0}{\delta t}\left(\left(\rho^n - \rho^* - B\epsilon^{*2}\right)h^n + K\epsilon^*\left(h^* - \epsilon^*\right) + \mathcal{O}\left(\epsilon^{*3}\right)\right) - \lambda h^n + Q^n \\
&= \frac{\gamma_0}{\delta t}\left(\left(\rho^n - \rho^* - B\epsilon^{*2} + K\epsilon^*\right)h^n + \mathcal{O}\left(\epsilon^{*3}\right)\right) - \lambda h^n + Q^n\,.
\end{aligned} \tag{84}$$

Finally, using the Taylor series in Eq. (70),

$$\frac{\gamma_0}{\delta t}(\rho h)^n + \sum_{i=1}^{q}\frac{\gamma_i}{\delta t}(\rho h)^{n-i} = \frac{\gamma_0}{\delta t}\left(\mathcal{O}\left(\epsilon^{*3}\right)h^n + \mathcal{O}\left(\epsilon^{*3}\right)\right) - \lambda h^n + Q^n\,, \tag{85}$$

which implies Eq. (40) with Eq. (41b).

References

[1] O. Axelsson, X. He, M. Neytcheva, Numerical solution of the time-dependent Navier-Stokes equation for variable density-variable viscosity. Part I, Math. Model. Anal. 20 (2) (2015) 232–260, https://doi.org/10.3846/13926292.2015.1021395.

[2] Satish Balay, William D. Gropp, Lois Curfman McInnes, Barry F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A.M. Bruaset, H.P. Langtangen (Eds.), Modern Software Tools in Scientific Computing, Birkhäuser Press, 1997, pp. 163–202.

[3] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Todd Munson, Karl Rupp, Patrick Sanan, Barry F. Smith, Stefano Zampini, Hong Zhang, Hong Zhang, PETSc users manual, Technical Report ANL-95/11 - Revision 3.9, Argonne National Laboratory, 2018.

[4] R. Barney, R. Nourgaliev, J.P. Delplanque, R. McCallen, Fully-implicit, high-order, reconstructed discontinuous Galerkin method for supercritical fluid flows, in: Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C), Portland, OR, USA, 2019, pp. 377–386.
[5] F. Bassi, A. Crivellini, D.A. Di Pietro, S. Rebay, An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier-Stokes equations, J. Comput. Phys. (ISSN 0021-9991) 218 (2) (2006) 794–815, https://doi.org/10.1016/j.jcp.2006.03.006.
[6] F. Bassi, L. Botti, A. Colombo, A. Ghidoni, F. Massa, Linearly implicit Rosenbrock-type Runge–Kutta schemes applied to the Discontinuous Galerkin solution of compressible and incompressible unsteady flows, Comput. Fluids 118 (2015) 305–320, https://doi.org/10.1016/j.compfluid.2015.06.007.
[7] Ian H. Bell, Jorrit Wronski, Sylvain Quoilin, Vincent Lemort, Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library CoolProp, Ind. Eng. Chem. Res. (ISSN 0888-5885) 53 (6) (Feb. 2014) 2498–2508, https://doi.org/10.1021/ie4033999.
[8] Lorenzo Botti, Daniele A. Di Pietro, A pressure-correction scheme for convection-dominated incompressible flows with discontinuous velocity and continuous pressure, J. Comput. Phys. 230 (3) (2011) 572–585, https://doi.org/10.1016/j.jcp.2010.10.004.
[9] K. Andrew Cliffe, Edward J.C. Hall, Paul Houston, Adaptive discontinuous Galerkin methods for eigenvalue problems arising in incompressible fluid flows, SIAM J. Sci. Comput. (ISSN 1064-8275) 31 (6) (Jan. 2010) 4607–4632, https://doi.org/10.1137/080731918.
[10] Bernardo Cockburn, Guido Kanschat, Dominik Schötzau, An equal-order DG method for the incompressible Navier-Stokes equations, J. Sci. Comput. (ISSN 1573-7691) 40 (1) (2009) 188–210, https://doi.org/10.1007/s10915-008-9261-1.
[11] S. Scott Collis, Discontinuous Galerkin methods for turbulence simulation, in: Summer Program 2002, Center for Turbulence Research, 2002, pp. 155–167, https://web.stanford.edu/group/ctr/ctrsp02/collis.pdf.
[12] Andrea Crivellini, Valerio D'Alessandro, Francesco Bassi, A Spalart-Allmaras turbulence model implementation in a discontinuous Galerkin solver for incompressible flows, J. Comput. Phys. (ISSN 0021-9991) 241 (2013) 388–415, https://doi.org/10.1016/j.jcp.2012.12.038.
[13] C. Ross Ethier, D.A. Steinman, Exact fully 3D Navier-Stokes Solutions for Benchmarking, Int. J. Numer. Methods Fluids 19 (March 1994) 369–375, https://doi.org/10.1002/fld.1650190502.
[14] Foll Fabian, Sandeep Pandey, Xu Chu, Claus-Dieter Munz, Eckart Laurien, Bernhard Weigand, High-fidelity direct numerical simulation of supercritical channel flow using discontinuous Galerkin spectral element method, in: High Performance Computing in Science and Engineering, Springer, Cham, 2019, pp. 275–289.
[15] E. Ferrer, D. Moxey, R.H.J. Willden, S.J. Sherwin, Stability of projection methods for incompressible flows using high order pressure-velocity pairs of same degree: continuous and discontinuous Galerkin formulations, Commun. Comput. Phys. (ISSN 1815-2406) 16 (3) (Sep. 2014) 817–840, https://doi.org/10.4208/cicp.290114.170414a.
[16] Christophe Geuzaine, J.F. Remacle, Gmsh: a 3-D finite element mesh generator with built-in pre- and post-processing facilities, Int. J. Numer. Methods Eng. (ISSN 1097-0207) 79 (11) (2009) 1309–1331, https://doi.org/10.1002/nme.
[17] J.-L. Guermond, Abner Salgado, A splitting method for incompressible flows with variable density based on a pressure Poisson equation, J. Comput. Phys. 228 (8) (may 2009) 2834–2846, https://doi.org/10.1016/j.jcp.2008.12.036.
[18] Ralf Hartmann, Numerical analysis of higher order discontinuous Galerkin finite element methods, 2008, http://elib.dlr.de/57074/1/Har08b.pdf.
[19] Aldo Hennink, Finite difference methods for the non-linear enthalpy equation, 2019, https://github.com/hennink/finite-difference-enthalpy-eq.
[20] Koen Hillewaert, Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries, PhD thesis, Université catholique de Louvain, 2013, http://hdl.handle.net/2078.1/128254.
[21] Guido Kanschat, Discontinuous Galerkin Methods for Viscous Incompressible Flow, 1st edition, Teubner Research: Deutscher Universitäts-Verlag, Wiesbaden, ISBN 978-3-8350-4001-4, 2007, http://site.ebrary.com/id/10231899.
[22] George Karypis, Vipin Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, SIAM J. Sci. Comput. 20 (1) (1998) 359–392, https://doi.org/10.1137/S1064827595287997.
[23] C.M. Klaij, J.J.W. van der Vegt, H. van der Ven, Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations, J. Comput. Phys. (ISSN 0021-9991) 217 (2) (2006) 589–611, https://doi.org/10.1016/j.jcp.2006.01.018.
[24] B. Klein, B. Müller, F. Kummer, M. Oberlack, A high-order discontinuous Galerkin solver for low Mach number flows, Int. J. Numer. Methods Fluids 81 (2016) 489–520, https://doi.org/10.1002/fld.4193.
[25] Benedikt Klein, A high-order Discontinuous Galerkin solver for incompressible and low-Mach number flows, PhD thesis, Technische Universität, Darmstadt, 2015, http://tuprints.ulb.tu-darmstadt.de/5143/.
[26] Benjamin Krank, Niklas Fehn, Wolfgang A. Wall, Martin Kronbichler, A high-order semi-explicit discontinuous Galerkin solver for 3d incompressible flow with application to dns and les of turbulent channel flow, J. Comput. Phys. (ISSN 0021-9991) 348 (2017) 634–659, https://doi.org/10.1016/j.jcp.2017.07.039.
[27] G. Mengaldo, D. De Grazia, D. Moxey, P.E. Vincent, S.J. Sherwin, Dealiasing techniques for high-order spectral element methods on regular and irregular grids, J. Comput. Phys. (ISSN 0021-9991) 299 (Oct. 2015) 56–81, https://doi.org/10.1016/J.JCP.2015.06.032.
[28] Habib N. Najm, Peter S. Wyckoff, Omar M. Knio, A semi-implicit numerical scheme for reacting flow, I: stiff chemistry, J. Comput. Phys. (ISSN 0021-9991) 143 (2) (Jul. 1998) 381–402, https://doi.org/10.1006/jcph.1997.5856.
[29] H. Nemati, Direct numerical simulation of turbulent heat transfer to fluids at supercritical pressures, doctoral thesis, TU Delft, 2016.
[30] F. Nicoud, Conservative high-order finite-difference schemes for low-Mach number flows, J. Comput. Phys. (ISSN 0021-9991) 158 (1) (2000) 71–97, https://doi.org/10.1006/jcph.1999.6408.
[31] A. Nigro, C. De Bartolo, R. Hartmann, F. Bassi, Discontinuous Galerkin solution of preconditioned Euler equations for very low Mach number flows, Int. J. Numer. Methods Fluids 63 (4) (2010) 449–467, https://doi.org/10.1002/fld.2083.
[32] Zahra Niroobakhsh, Nehzat Emamy, Roozbeh Mousavi, Florian Kummer, Martin Oberlack, Numerical investigation of laminar vortex shedding applying a discontinuous Galerkin finite element method, Prog. Comput. Fluid Dyn. (ISSN 1468-4349) 17 (3) (2017) 131, https://doi.org/10.1504/PCFD.2017.084346.
[33] J.W.R. Peeters, Turbulence and turbulent heat transfer at supercritical pressure, doctoral thesis, Delft University of Technology, 2016.
[34] L. Pesch, J.J.W. van der Vegt, A discontinuous Galerkin finite element discretization of the Euler equations for compressible and incompressible fluids, J. Comput. Phys. (ISSN 0021-9991) 227 (11) (2008) 5426–5446, https://doi.org/10.1016/j.jcp.2008.01.046.
[35] Marian Piatkowski, Steffen Müthing, Peter Bastian, A stable and high-order accurate discontinuous Galerkin based splitting method for the incompressible Navier-Stokes equations, J. Comput. Phys. (ISSN 0021-9991) 356 (2018) 220–239, https://doi.org/10.1016/J.JCP.2017.11.035.
[36] Daniele Antonio Di Pietro, Alexandre Ern, Mathematical Aspects of Discontinuous Galerkin Methods, Springer, ISBN 9783642229800, 2012.
[37] Sander Rhebergen, Bernardo Cockburn, Jaap J.W. van der Vegt, A space-time discontinuous Galerkin method for the incompressible Navier-Stokes equations, J. Comput. Phys. 233 (2013) 339–358, https://doi.org/10.1016/j.jcp.2012.08.052.
[38] Anatol Roshko, On the development of turbulent wakes from vortex streets, PhD thesis, California Institute of Technology, 1952, http://resolver.caltech.edu/CaltechETD:etd-10302003-144124.
[39] Kambiz Salari, Patrick Knupp, Code Verification by the Method of Manufactured Solutions, Technical report, Sandia National Labs., Albuquerque, NM (US), Livermore, CA (US), 2000, https://www.osti.gov/biblio/759450/.
[40] Khosro Shahbazi, A Parallel High-Order Discontinuous Galerkin Solver for the Unsteady Incompressible Navier-Stokes Equations in Complex Geometries, PhD thesis, University of Toronto, 2007.
[41] Khosro Shahbazi, Paul F. Fischer, C. Ross Ethier, A high-order discontinuous Galerkin method for the unsteady incompressible Navier-Stokes equations, J. Comput. Phys. 222 (1) (2007) 391–407, https://doi.org/10.1016/j.jcp.2006.07.029.

[42] J.-M. Shi, D. Gerlach, M. Breuer, G. Biswas, F. Durst, Heating effect on steady and unsteady horizontal laminar flow of air past a circular cylinder, Phys. Fluids 16 (12) (2004) 4331–4345, https://doi.org/10.1063/1.1804547.
[43] Lee Shunn, Frank Ham, Parviz Moin, Verification of variable-density flow solvers using manufactured solutions, J. Comput. Phys. (ISSN 0021-9991) 231 (9) (2012) 3801–3827, https://doi.org/10.1016/j.jcp.2012.01.027.
[44] Pavel Solin, Karel Segeth, Ivo Dolezel, Higher-Order Finite Element Methods, 1st edition, Chapman and Hall/CRC, New York, ISBN 9781584884385, 2003.
[45] Roland Span, Wolfgang Wagner, A new equation of state for carbon dioxide covering the fluid region from the triple-point temperature to 1100 K at pressures up to 800 MPa, J. Phys. Chem. Ref. Data (ISSN 0047-2689) 25 (6) (Nov. 1996) 1509–1596, https://doi.org/10.1063/1.555991.
[46] G. Volpe, Performance of compressible flow codes at low Mach numbers, AIAA J. 31 (1) (1993) 49–56, https://doi.org/10.2514/3.11317.
[47] An-Bang Wang, Zdenek Trávníček, Kai-Chien Chia, On the relationship of effective Reynolds number and Strouhal number for the laminar vortex shedding of a heated circular cylinder, Phys. Fluids 12 (6) (2000) 1401–1410, https://doi.org/10.1063/1.870391.
[48] Wikipedia contributors. Kahan summation algorithm — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Kahan_summation_algorithm&oldid=970524976, 2020. (Accessed 14 September 2020) [Online].
[49] C.H.K. Williamson, Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers, J. Fluid Mech. 206 (1989) 579–627, https://doi.org/10.1017/S0022112089002429.