



HAL
open science

A high-order numerical algorithm for DNS of low-Mach-number reactive flows with detailed chemistry and quasi-spectral accuracy

Emmanuel Motheau, John Abraham

► **To cite this version:**

Emmanuel Motheau, John Abraham. A high-order numerical algorithm for DNS of low-Mach-number reactive flows with detailed chemistry and quasi-spectral accuracy. *Journal of Computational Physics*, 2016, 313, pp.430-454. 10.1016/j.jcp.2016.02.059 . hal-01297033

HAL Id: hal-01297033

<https://hal.science/hal-01297033>

Submitted on 1 Apr 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A high-order numerical algorithm for DNS of low-Mach-number reactive flows with detailed chemistry and quasi-spectral accuracy

E. Motheau^{a,*}, J. Abraham^{a,b}

^a*School of Mechanical Engineering, University of Adelaide, Adelaide, SA 5005, Australia*

^b*School of Mechanical Engineering, Purdue University, 585 Purdue Mall, West Lafayette, IN 47907-2088, USA*

Abstract

A novel and efficient algorithm is presented in this paper to deal with DNS of turbulent reacting flows under the low-Mach-number assumption, with detailed chemistry and a quasi-spectral accuracy. The temporal integration of the equations relies on an operating-split strategy, where chemical reactions are solved implicitly with a stiff solver and the convection-diffusion operators are solved with a Runge-Kutta-Chebyshev method. The spatial discretisation is performed with high-order compact schemes, and a FFT based constant-coefficient spectral solver is employed to solve a variable-coefficient Poisson equation. The numerical implementation takes advantage of the **2DECOMP&FFT** libraries developed by Li and Laizet [1], which are based on a pencil decomposition method of the domain and are proven to be computationally very efficient. An enhanced pressure-correction method is proposed to speed-up the achievement of machine precision accuracy. It is demonstrated that a second-order accuracy is reached in time, while the spatial accuracy ranges from fourth-order to sixth-order depending on the set of imposed boundary conditions. The software developed to implement the present algorithm is called **HOLOMAC**, and its numerical efficiency opens the way to deal with DNS of reacting flows to understand complex turbulent and chemical phenomena in flames.

Keywords: DNS, Low-Mach-number, Detailed chemistry, Turbulent reacting flow, High-order methods, Spectral accuracy, Operator splitting

1. Introduction

The rapid growth of computational capabilities in the last decades has allowed the application of high-fidelity numerical methods to unsteady reactive turbulent flows. For example, Large-Eddy Simulations (LES) are now commonly employed as a predictive tool in realistic complex configurations of interest to industry (see Pitsch [2], Gicquel et al. [3], Motheau et al. [4], among others). However, such methods rely on several models to take into account the unresolved physics. The development of turbulent combustion models that can be employed in practical applications requires high-fidelity data with which model results can be compared. In some cases, experimental results are available and very useful, but they often lack detailed

*Corresponding author. Present address: Lawrence Berkeley National Laboratory, MS 50A-1148, 1 Cyclotron Rd, Berkeley, CA 94720, USA.

Email address: emotheau@lbl.gov (E. Motheau)

Preprint submitted to *Journal of Computational Physics*

February 25, 2016

10 information about the many species and temperature that are involved in chemical kinetics and
11 about flow parameters like strain rate and turbulent statistics. Hence, Direct Numerical Simula-
12 tion (DNS) is a powerful tool to generate the data required to develop models. Note also that in
13 many applications such as gas turbines and reciprocating engines, ignition and extinction are im-
14 portant limit phenomena that effect system performance. The accurate prediction of these limit
15 phenomena requires detailed kinetics. Moreover, quasi-spectral accuracy is needed to generate
16 the high-fidelity data.

17 In principle, whether LES or DNS approaches are employed, the most accurate representa-
18 tion of the physics would be achieved through the solution of the full set of compressible, reactive
19 Navier-Stokes equations with detailed chemistry models. The computational cost of solving the
20 compressible equations can be however prohibitive. One reason is the complexity of the chem-
21 istry. The reacting mixture generally includes a large number of chemical species with many
22 hundreds of chemical reactions. The solution of the reaction equations leads to complex, stiff
23 systems of equations that must be solved with dedicated numerical algorithms. In practice, sim-
24 ple models with reduced mechanisms are often employed to represent the chemistry with a very
25 few number of species and chemical reactions (see Franzelli et al. [5] for an example). Detailed
26 chemistry is generally limited to DNS of fundamental problems in canonical configurations (see
27 Mukhopadhyay and Abraham [6] for an example).

28 Furthermore, from a numerical point of view, it is well known that the fully compressible ap-
29 proach presents several issues when dealing with flows featuring regions where the Mach number
30 M is small, which is often the case in combustion process [7]. When $M \rightarrow 0$ the equations tends
31 to a singularity that break the stability and accuracy of numerical schemes employed to solve the
32 set of fully compressible Navier-Stokes equations. As pointed out by Volpe [8], either with an
33 explicit or an implicit approach for the time discretisation, when the Mach number is decreased
34 to very small values the numerical errors increase as well as the convergence rate deteriorates.
35 Furthermore, the critical time-step scales with the Mach number through a CFL condition so as to
36 ensure the numerical stability due to the presence of acoustic waves. Hence, low-Mach-number
37 regions in the flow will constrain the time-step to impracticable small values, which is a waste of
38 time and computational resources in practice.

39 Several methods have been proposed to handle this particular issue of low-Mach-number
40 regions in flows. Basically two approaches can be distinguished:

- 41 • The preconditioning methods, in which the time-derivatives in the Navier-Stokes equations
42 are modified so as to optimise the eigenvalues of the compressible system by reducing the
43 disparities between hydrodynamic and acoustic wave speeds. However, because the time-
44 derivatives are modified, these methods are preferably applied to steady-state problems [9].
45 Otherwise, they can be embedded through dual time-stepping algorithms [10, 11]. Most
46 restricting, these methods cannot be generalised and must be properly designed for each
47 particular problem investigated [12].
- 48 • The projection methods, where an equation for the pressure is solved to enforce a diver-
49 gence constraint on the velocity field. Originally developed by Chorin [13] as a fractional-
50 step method to solve incompressible flows, two different approaches were derived by re-
51 formulating the compressible Navier-Stokes equations:
 - 52 – under the low-Mach-number assumption [14, 15], which remove the acoustic waves
53 from the equations and only keeps hydrodynamic and entropic fluctuations;

- 54 – with a scaling with respect to the pressure, leading to the so-called *Mach-uniform* or
55 *all-speed* methods [16, 17, 18].

56 The domain of validity of the first approach is of course restricted to low-Mach-number
57 flows and no acoustic phenomena can be captured, while the second approach aims to
58 provide a more general method that can handle flows featuring a wide range of Mach
59 numbers, from virtually zero to supersonic shockwaves, and where all compressibility
60 effects are taken into account.

61 For DNS of combustion applications, the implementation of a detailed chemistry solver and
62 high-order discretisation methods is not straightforward. Whereas the Mach-uniform method
63 has received little attention [19, 20] in the past few years, the low-Mach-number approach has
64 received more attention [21, 22, 23]. Most algorithms published in the past are based on a
65 fractional-step approach. Different ways to arrange the conservation equations are possible, but
66 as recalled by Knikker [15] in his review paper, it is not possible to solve all of them in a conser-
67 vative form unless an implicit approach is employed. This is, however, impractical in the context
68 of DNS with detailed chemistry. For example Najm et al. [24], followed by Knio et al. [25],
69 proposed to sacrifice the energy equation, solving it in its non-conservative form in the whole
70 algorithm. Taking a different approach, Day and Bell [26] proposed a complex algorithm to
71 solve the energy equation in its conservative form, but by compromising on the equation of state.
72 This then required the implementation of a damping source term to control the deviation of the
73 computed solution away from one which satisfies the equation of state. The developments that
74 followed these seminal works were focused on the implementation of adaptative mesh refinement
75 [27] or the design of more efficient and stable algorithms [28], especially for the treatment of the
76 diffusion terms as well as the pressure equation appearing in the projection-correction procedure.

77 The aim of the present paper is to introduce an efficient algorithm, with optimised numerical
78 methods, to perform high-fidelity DNS of reacting flows under the low-Mach-number assump-
79 tion, with detailed chemistry and quasi-spectral accuracy. The first innovation of the present
80 work is to compose a novel algorithm by selecting appropriate strategies and numerical methods
81 from the prior works discussed above. The second innovation is that high-order discretisation
82 schemes with spectral-like resolution are employed for all the variables solved. The pressure
83 is solved with an FFT solver by means of the so-called spectral equivalence principle. To the
84 authors' knowledge such accuracy for the pressure has not been reached before in the context
85 of simulations of low-Mach-number combustion with detailed chemistry. Finally, the third sig-
86 nificant innovation of the present paper is to propose an enhanced efficient method to solve the
87 variable-coefficient pressure equation by employing a constant-coefficient spectral solver, allow-
88 ing speed-up in achieving machine precision accuracy.

89 The present paper is organised as follows. First, the governing reactive Navier-Stokes equa-
90 tions under the low-Mach-number assumption are presented in Sec. 2. Next, the particular issues
91 related to the treatment of the detailed chemistry, the diffusion terms, and the solution of the
92 pressure equation are presented in Sec. 3. A description of the whole algorithm and its imple-
93 mentation in the **HOLOMAC** (High-Order LOw-MACH number Combustion) software is then
94 given. In Sec. 4, the performance and accuracy of the algorithm are assessed with the help of
95 several test problems. It is demonstrated that a second-order accuracy is achieved in time, while
96 the spatial accuracy ranges from fourth-order to sixth-order, depending on the set of imposed
97 boundary conditions.

98 2. Governing equations

99 According to Giovangigli [29], a dimensional analysis of the momentum equation that ap-
 100 pears in the set of fully compressible reactive Navier-Stokes equations (see Poinso and Veynante
 101 [30] for further developments) reveals that the spatial gradient of the pressure p is of order $O(M^2)$
 102 so that

$$p(\mathbf{x}, t) = p_0(t) + \underbrace{p_1(\mathbf{x}, t)}_{O(M^2)}, \quad (1)$$

103 where $p_0(t)$ is spatially uniform and $p_1(\mathbf{x}, t)$ is the fluid dynamic perturbation, while M is the
 104 Mach number. Introducing the pressure split expressed in Eq. (1) in the state equation and taking
 105 the asymptotic limit $M \rightarrow 0$ leads to the following simplified state law:

$$p_0 = \rho T \frac{\mathcal{R}}{\bar{W}}, \quad (2)$$

106 where ρ is the density, T the temperature, \mathcal{R} the universal gas constant and \bar{W} the mean molecular
 107 weight defined as

$$\bar{W} = 1 / \sum_{s=1}^{N_s} \frac{Y_s}{W_s}. \quad (3)$$

108 Here, W_s and Y_s are the molecular weight and the mass fraction of the species s , respectively, N_s
 109 being the total number of species present in the mixture.

110 The meaning of Eq. (2) is that the thermodynamic pressure p_0 is constant in space (but may
 111 vary in time) and it is decoupled from the fluctuating part p_1 . As the Mach number is considered
 112 small, acoustic fluctuations are neglected and $p_1(\mathbf{x}, t)$ only embeds hydrodynamic perturbations.
 113 A rigorous mathematical derivation may be found in the seminal work of Majda and Sethian
 114 [21]. Conservation equations of continuity, momentum, energy and species transport can then be
 115 recast in the following low-Mach-number formulation, respectively:

$$\frac{\partial \rho}{\partial t} = -\frac{\partial \rho u_i}{\partial x_i}, \quad (4)$$

$$\frac{\partial \rho u_i}{\partial t} = C_{u_i} + D_{u_i} - \frac{\partial p_1}{\partial x_i}, \quad (5)$$

$$\frac{\partial T}{\partial t} = C_T + D_T + R_T + \frac{1}{\rho C_p} \frac{dp_0}{dt}, \quad (6)$$

$$\frac{\partial Y_s}{\partial t} = C_{Y_s} + D_{Y_s} + R_{Y_s} \quad s = 1, 2, \dots, N_s, \quad (7)$$

116 where u_i and x_i are the velocity and spatial coordinate along the i -th direction, respectively.

117 The convection C and diffusion D terms in Eqs. (5-7) are defined as

$$C_{u_i} = -\frac{\partial \rho u_j u_i}{\partial x_j}, \quad D_{u_i} = \frac{\partial}{\partial x_j} \left(\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_k}{\partial x_k} \delta_{ij} \right), \quad (8)$$

$$C_T = -u_i \frac{\partial T}{\partial x_i}, \quad D_T = \frac{1}{\rho C_p} \left(\frac{\partial}{\partial x_i} \left(\lambda \frac{\partial T}{\partial x_i} \right) \right) + \frac{1}{\rho C_p} \left(\rho \sum_{s=1}^{N_s} C_{p,s} V_{s,i} Y_s \right) \frac{\partial T}{\partial x_i}, \quad (9)$$

$$C_{Y_s} = -u_i \frac{\partial Y_s}{\partial x_i}, \quad D_{Y_s} = \frac{1}{\rho} \frac{\partial \rho V_{s,i} Y_s}{\partial x_i}, \quad (10)$$

118 with

$$V_{s,i} Y_s = \mathcal{D}_s \frac{\partial Y_s}{\partial x_i} - Y_s \sum_{n=1}^{N_s} \mathcal{D}_n \frac{\partial Y_n}{\partial x_i}. \quad (11)$$

119 The source terms from chemical reactions are given by:

$$R_T = -\frac{1}{\rho C_p} \sum_{s=1}^{N_s} h_s \dot{\omega}_s, \quad R_{Y_s} = \frac{1}{\rho} \dot{\omega}_s. \quad (12)$$

120 In the above equations, μ and λ are the dynamic viscosity and heat conductivity of the mix-
121 ture, respectively, while $C_{p,s}$ and C_p are the specific heat capacity at constant pressure for the
122 species s and the total mixture, respectively, and are related by the following expression:

$$C_p = \sum_{s=1}^{N_s} Y_s C_{p,s}. \quad (13)$$

123 The terms $\dot{\omega}_s$ and h_s represent the mass production rate and enthalpy, respectively, of the
124 species s . The enthalpy term h_s is expressed as:

$$h_s = \int_{T_0}^T C_{p,s} dT + \Delta h_{f,s}^0, \quad (14)$$

125 where $\Delta h_{f,s}^0$ is the enthalpy of formation of the species s at $T_0 = 298.15$ K. Finally, \mathcal{D}_s is
126 the mass diffusivity of the species s . Rigorous evaluation of \mathcal{D}_s is very expensive and several
127 numerical strategies have been proposed to significantly reduce the computational burden (see
128 Ern and Giovangigli [31], Magin and Degrez [32]). These methods have generally been applied
129 to high-speed flows. In the present work, for ease of implementation, diffusion is approximated
130 to first-order by using an effective-diffusivity multicomponent diffusion model formulated under
131 the Hirschfelder-Curtiss [33] assumption:

$$\mathcal{D}_s = \frac{1 - Y_s}{\sum_{j \neq s} X_j / \mathcal{D}_{js}}. \quad (15)$$

132 Here \mathcal{D}_{js} is the binary diffusion coefficient between species j and s , and X_s is the mole fraction
133 of species s . A well-known issue with this latter assumption is that the mass is not conserved,
134 but the global mass can be maintained by adding a correction velocity term [30]. In the present
135 mathematical formulation, this term appears in the right hand side of Eq. (11). Note also that

136 Soret (molecular species diffusion due to temperature gradients) and Dufour (heat flux due to
137 species mass fraction gradients) effects have been neglected.

138 The set of Eqs. (5-7) is subject to a constraint on the velocity field, which will be used in
139 the pressure-projection part of the fractional-step algorithm. This constraint can be derived by
140 reformulating the continuity Eq. (4) as

$$\frac{\partial u_i}{\partial x_i} = -\frac{1}{\rho} \frac{D\rho}{Dt}, \quad (16)$$

141 where the RHS of Eq. (16) is obtained by differentiating the equation of state along particle paths,
142 leading to:

$$\frac{\partial u_i}{\partial x_i} = \frac{1}{T} \frac{DT}{Dt} + \sum_{s=1}^{N_s} \frac{\bar{W}}{W_s} \frac{DY_s}{Dt} - \frac{1}{p_0} \frac{dp_0}{dt}. \quad (17)$$

143 Replacing the material derivatives that appear in Eq. (17) by their expressions in Eqs. (6) and (7)
144 leads to the following expression of the velocity constraint:

$$\frac{\partial u_i}{\partial x_i} = \frac{dp_0}{dt} \tilde{\varphi} + \tilde{\mathcal{D}}, \quad (18)$$

where

$$\tilde{\varphi} = \frac{1}{\rho C_p T} - \frac{1}{p_0}, \quad (19)$$

$$\tilde{\mathcal{D}} = \frac{1}{T} (D_T + R_T) + \sum_{s=1}^{N_s} \frac{\bar{W}}{W_s} (D_{Y_s} + R_{Y_s}). \quad (20)$$

145 In the context of a simulation with open boundaries, typically inflow/outflow conditions, the
146 thermodynamic pressure p_0 is static in time and set by atmospheric conditions. Consequently,
147 the term dp_0/dt vanishes in Eq. (18). However if the computational domain is closed, the ther-
148 modynamic pressure p_0 may change in time. As the total mass remains constant through the
149 domain and is equal to the volume integral of the density, p_0 can be expressed with the help of
150 Eq. (2) and reads:

$$p_0 = \frac{M_0 \mathcal{R}}{\int_V \left(T \sum_{s=1}^{N_s} \frac{Y_s}{W_s} \right)^{-1} dV}, \quad (21)$$

151 with $M_0 = \int_V \rho dV$. As pointed out by Nicoud [14], the time derivative of p_0 may be expressed
152 by integrating Eq. (18) over a domain V to give the following equation:

$$\frac{dp_0}{dt} = \left(\int_V \partial u_i / \partial x_i dV - \int_V \tilde{\mathcal{D}} dV \right) / \int_V \tilde{\varphi} dV. \quad (22)$$

153 Since $\int_V \partial u_i / \partial x_i dV = \int_S u_i^n dS$, S being the surface boundary, the velocities along normal n sum
154 up to zero if hard walls or periodic boundary conditions are imposed everywhere. Hence, the
155 term $\int_V \partial u_i / \partial x_i dV$ in Eq. (22) vanishes.

156 3. Numerical methods

157 3.1. Overall presentation of the algorithm

158 The algorithm developed in the present paper is based on a fractional-step, segregated method.
159 Basically the procedure can be summarised by the following two parts:

- 160 1. The thermochemical system composed of the energy Eq. (6) and the species transport
161 Eq. (7) is solved first. During this step, temperature and species mass fractions are ad-
162 vanced in time, making it possible to compute a new density through the equation of state
163 (2).
- 164 2. With this new density, the momentum Eq. (5) can be integrated to advanced in time the ve-
165 locity fields. This step relies on a projection/correction method, where a Poisson equation
166 for the pressure is solved to enforce the divergence condition imposed by the continuity
167 Eq. (16).

168 3.1.1. Operator-split techniques, stiff integration and treatment of the diffusion

169 A particular issue encountered in Part 1 is with the wide range of time scales involved in the
170 different operators comprising the thermochemical system. Indeed, due to the detailed chemistry,
171 the evaluation of the chemical reactions is very sensitive to the state variables and it forms a stiff
172 system of ODEs to solve, which requires dedicated numerical methods that belong to the class
173 of *stiff solver* (see Hairer and Wanner [34] for a review). However, the set of ODEs has con-
174 vection and diffusion operators, and including them into a stiff integrator would be particularly
175 inefficient. The popular strategy employed to cope with this numerical challenge is to solve each
176 term separately through an operator-split scheme. Knio et al. [25] and Day and Bell [26] imple-
177 mented the so-called *Strang operator-split* (see Strang [35]), which is second order in time, and
178 demonstrated its efficiency to solve the thermochemical system while keeping large time-steps
179 and an acceptable computational cost. As recalled by Duarte et al. [36], a particular attraction of
180 this strategy is that each operator can be solved with its own specific numerical method, opening
181 the way to tailor an overall algorithm.

182 In the algorithm developed in the present paper, the following operator-split scheme is re-
183 tained:

$$\mathbf{H}(t^n + \Delta t^n) = \mathcal{H}_{dt/2}^{C-D} \mathcal{H}_{dt}^R \mathcal{H}_{dt/2}^{C-D} \mathbf{H}(t^n), \quad (23)$$

184 where \mathbf{H} is the solution vector while \mathcal{H} refers to the operators of convection (C), diffusion (D)
185 and reaction (R). Another combination $\mathcal{H}_{dt/2}^R \mathcal{H}_{dt}^{C-D} \mathcal{H}_{dt/2}^R$ was tested and compared to Eq. (23)
186 for the freely propagating methane/air flames cases presented at §4.1, but virtually no difference
187 was noticed in the solutions. It is emphasised that within the order of time-steps employed for
188 the whole algorithm, errors occurring in the operator-splitting procedure are negligible. Con-
189 sequently, the operator-split scheme described at Eq. (23) is selected because as the evaluation
190 of the reaction operators requires a significant computational effort, it is practically more time-
191 efficient to solve it only once.

192 A second issue arising in Part 1 is the choice of a numerical method for the integration of
193 the diffusion operators. It is well known that the evaluation of diffusion imposes a limit on the
194 time-step to ensure the stability of the algorithm. This limitation can be reduced by treating the
195 diffusion terms with an implicit method. However, as recalled by Najm and Knio [37], the dif-
196 fusion terms exhibit a non-linear dependence on the temperature and species mass fractions, and

197 are strongly coupled to the reaction operators. Hence, relying on an implicit method is particu-
198 larly unattractive within the context of solving detailed reacting flows. To overcome this issue,
199 Day and Bell [26] proposed to solve a temperature equation prior to an enthalpy equation in or-
200 der to provide fluid properties at the new time level. Another possibility proposed by Knio et al.
201 [25] is to solve explicitly the diffusion operators by decomposing the time integration in smaller
202 fractional steps. However the major drawback of this approach is the large number of iterations
203 required to ensure stability. For example, Yu et al. [28] reported the need to use more than one
204 hundred fractional steps for the simulation of a hydrogen/air flame. Such poor performance moti-
205 vated Najm and Knio [37] to adopt an explicit Runge-Kutta-Chebyshev (RKC) method [38]
206 instead of their initial choice based on the fractional-steps approach. They reported a number of
207 iterations of approximately 32, which is a significant gain of computational time. However in this
208 latter work, the authors reformulated the temperature equation into an evolution equation for the
209 density, requiring evaluation of additional operators, which is not computationally efficient (see
210 the discussion in the review of Knikker [15]). In the present paper, the temperature and species
211 mass fraction equations are kept in their original form, and the RKC method is selected to inte-
212 grate the \mathcal{H}^{C-D} operators. Results presented in Sec. 4 show that only a dozen (approximately)
213 iterations for the RKC method are enough to ensure stability, even for relatively high CFL num-
214 bers of approximately 0.85. This value is close to the stability limit for explicit time integration.
215 Moreover, for all cases studied, an estimation based on the von Neumann stability criterion for
216 the diffusion has been conducted. It shows that, compared to the RKC method, the maximum
217 critical time-step would be about four orders of magnitude smaller with a forward explicit time-
218 integration. Thus, the results demonstrate that the RKC method is very efficient to deal with
219 relatively larger time-steps. Note that the fluid properties and chemical reactions are computed
220 with the **CHEMKIN** libraries [39]. As pointed out by Najm and Knio [37], the repeated eval-
221 uation of fluid properties through these libraries is computationally costly and they proposed to
222 rely on extrapolation/interpolation techniques instead. Such an approach, particularly efficient
223 and introducing negligible errors, is adopted in the present paper.

224 Particular attention must be drawn to the convection operators. Indeed, they all involve the
225 velocity vector, which is however not known at the next time-step. Thus, repeated evaluations
226 of the convection operators at each RKC stage would require an extrapolation of the velocities.
227 Instead, the time integration of the convection operators is performed once with a linear multi-
228 step method, namely a second-order Adams-Bashforth (AB2) scheme. Hence, convection terms
229 are imposed during the integration of diffusion operators as constant source terms. Yu et al. [28]
230 pointed out that merging the term appearing in the right hand side of D_T (see Eq. (9)) into the
231 diffusion integration procedure would increase the computational burden. Instead, they merged
232 this term with the convection operator to impose it as a constant source term. It was found in the
233 development of the present algorithm that such a choice leads locally to a lack of conservation
234 of the mass, reducing strongly the accuracy of the algorithm. Hence, despite the additional
235 computational cost, this term is repeatedly evaluated during the integration procedure to ensure
236 mass conservation.

237 3.1.2. Projection methods and treatment of the pressure

238 The pressure-projection step employed in Part 2 is widely pointed out in the literature as
239 posing a particular difficulty, playing a crucial role in determining the numerical stability [14, 15].
240 The origin of this method goes back to the fractional-step, projection method developed for
241 incompressible flows by Chorin [13]. Basically, the velocity and the pressure are decoupled in
242 the momentum Eq. (5); after the time advancement of the velocity, a Poisson equation for the

243 pressure is solved to project the intermediate velocity onto a space that enforce a divergence-
244 free constraint. In the context of low-Mach-number flows, the variable density appears in the
245 Poisson equation. In the fractional-step method, the constraint on the velocity field is imposed
246 from already known values computed at the new time level, and can be formulated in two ways:

- 247 • A first approach is to impose the divergence of the momentum (ρu), which is known
248 through the continuity Eq. (4). This amounts to computing and imposing the time deriva-
249 tive of the density at the new time level. It leads to a *constant-coefficient* Poisson equation
250 requiring the evaluation of the operator ∇^2 .
- 251 • A second approach is to impose the divergence of the velocity u , which is known by taking
252 the material derivative of the equation of state, as detailed in Eqs. (16-17). It leads to
253 a *variable-coefficient* Poisson equation requiring the evaluation of the operator $\nabla(1/\rho\nabla)$,
254 where ρ varies spatially.

255 The numerical solution of the *constant-coefficient* Poisson equation is the most straightfor-
256 ward and can be done by means of direct solvers. However this approach presents two ma-
257 jor drawbacks: first, it has been demonstrated by Nicoud [14] that in the inviscid limit, un-
258 like the *variable-coefficient* Poisson equation approach, this formulation does not recover the
259 divergence-free velocity constraint. Second, it has been widely reported in the literature that the
260 time-derivative of density is a source of instability in the Poisson equation, limiting the method to
261 small amplitude variations of the density (the ratio value of 3 is commonly reported). Najm et al.
262 [24] proposed a predictor-corrector scheme that extended the stability to density ratios of 10.
263 During the development of the present paper, this predictor-corrector approach has been tested
264 and it has been observed that the stability also depends on the spatial steepness and time evolution
265 of the density ratio. In the context of autoignition of an heptane/air mixture presented in §4.3,
266 this approach has been found to be always unstable for density ratios greater than approximately
267 7.

268 Such disappointing results have motivated the adoption of an approach based on the *variable-*
269 *coefficient* Poisson equation. Although not shown in the present paper, numerical tests have
270 validated the strongly stable behaviour of this approach for large and steep density ratios. How-
271 ever, the major drawback reported in the literature is that this equation is challenging to solve
272 numerically. In the context of low-Mach-number reactive flows, many of the algorithms pub-
273 lished rely on iterative solvers. Yu et al. [28] and Safta et al. [27] used multigrid methods with a
274 finite-difference discretisation, reaching second and fourth-order accuracy, respectively. On the
275 other hand, Knikker [15] employed a second-order Krylov-based solver but reached fourth-order
276 accuracy by using an iterative residual correction method, the residuals being computed with
277 compact schemes. Moreover, Desjardins et al. [40] reached similar high-order accuracy with a
278 combination of spectral and Krylov-based methods. Unlike the previous works cited above, a
279 constant-coefficient spectral solver based on fast Fourier transforms and pencil domain decom-
280 position is employed in this work as a novel contribution to the literature. It is shown in Sec. 4
281 that sixth-order accuracy is reached for the pressure.

282 The numerical methods presented in this paper are implemented in the **HOLOMAC** soft-
283 ware, which is based on the **Incompact3D** framework [41, 42] and the **2DECOMP&FFT** li-
284 braries [1]. Details about the spectral solver, its implementation and the parallelisation tech-
285 niques can be found in the aforementioned references. Note that the **FFTW3** libraries [43] have
286 been implemented to perform the Fast Fourier Transforms (FFT) employed by the spectral solver

287 for the pressure. The mathematical description of the algorithm for the time integration, as well
 288 as the numerical methods for the spatial discretisation, are now presented below.

289 3.2. Temporal integration

290 3.2.1. Step 1: Computation of explicit terms

291 The first step of the present algorithm is to compute the explicit terms that will be imposed as
 292 source terms during the time integration of the diffusion operators with the RKC method. These
 293 source terms embed the explicit time integration of the convection terms C_{u_i} , C_T and C_{Y_s} from the
 294 time-step n to $n+1$ with an AB2 scheme, as well as the spatial gradient of hydrodynamic pressure
 295 p_1 and the time derivative of the thermodynamic pressure p_0 that appear in the momentum Eq. (5)
 296 and the energy Eq. (6), respectively. The source terms S_{u_i} , S_T and S_{Y_s} for the momentum, energy
 297 and species transport equations are expressed as follows:

$$S_{u_i} = \left[(1 + \zeta_t) C_{u_i}^n - \zeta_t C_{u_i}^{n-1} \right] - \frac{\partial p_1^n}{\partial x_i}, \quad (24)$$

$$S_T = \left[(1 + \zeta_t) C_T^n - \zeta_t C_T^{n-1} \right] + \frac{1}{(1 + \zeta_t) \rho^n C_p^n - \zeta_t \rho^{n-1} C_p^{n-1}} \frac{\widetilde{dp}_0}{dt}, \quad (25)$$

$$S_{Y_s} = (1 + \zeta_t) C_{Y_s}^n - \zeta_t C_{Y_s}^{n-1}, \quad (26)$$

298 where

$$\frac{\widetilde{dp}_0}{dt} = \frac{\Delta t^{n-2} \left(1 + \frac{\Delta t^n}{\Delta t^{n-1}} \right) p_0^n - \left(\Delta t^{n-2} + \Delta t^n + \frac{\Delta t^n \Delta t^{n-2}}{\Delta t^{n-1}} \right) p_0^{n-1} + \Delta t^n p_0^{n-2}}{\Delta t^{n-1} \Delta t^{n-2}} \quad (27)$$

is the extrapolation of the time derivative of the thermodynamic pressure computed over the new
 time-step, and with

$$\zeta_t = \frac{1}{2} \frac{\Delta t^n}{\Delta t^{n-1}}, \quad (28)$$

$$\Delta t^n = t^{n+1} - t^n, \quad (29)$$

$$\Delta t^{n-1} = t^n - t^{n-1}, \quad (30)$$

$$\Delta t^{n-2} = t^{n-1} - t^{n-2}. \quad (31)$$

299 Note that the evaluation of the time derivative of p_0 through Eq. (22) would require all the
 300 diffusion and reaction operators to be computed twice, as well as two volume integrations over
 301 the full domain to be performed. In order to save computational time, this procedure is only per-
 302 formed at the first iteration of a simulation, the remaining one being computed with the discrete
 303 formulation expressed in Eq. (27). Of course in the case of an open domain with inflow/outflow
 304 boundary conditions, $\frac{\widetilde{dp}_0}{dt} = 0$.

305 3.2.2. Step 2: RKC integration of scalars diffusion terms over the first half time-step

306 The diffusion terms D_{u_i} , D_T and D_{Y_s} are integrated from the time-step n to $n + 1/2$ with
 307 the explicit Runge-Kutta-Chebyshev (RKC) method presented by Verwer et al. [38]. The source
 308 terms computed in §3.2.1 are imposed during each iteration of the RKC method. The total

309 number of iterations K is a free parameter chosen by the user, but the RKC scheme requires at
 310 least two iterations, viz. $K \geq 2$.

311 The starting values, denoted by superscript 0, are values at time-step n :

$$T^0 = T^n, \quad (32)$$

$$Y_s^0 = Y_s^n, \quad (33)$$

$$\rho^0 = \rho^n. \quad (34)$$

312 During the first stage of integration, denoted by superscript 1, values are updated using the
 313 following relations:

$$T^1 = T^0 + \tilde{\eta}_1 \frac{\Delta t^n}{2} (D_T^0 + S_T), \quad (35)$$

$$Y_s^1 = Y_s^0 + \tilde{\eta}_1 \frac{\Delta t^n}{2} (D_{Y_s}^0 + S_{Y_s}), \quad (36)$$

$$\rho^1 = \frac{P_0^n}{\mathcal{R}T^1 \sum_{s=1}^{N_s} \frac{Y_s^1}{W_s^1}}. \quad (37)$$

314 Integration over the remaining stages of integration, denoted k with $k = 2, \dots, K$, is per-
 315 formed by employing the following relations:

$$T^k = (1 - \eta_k - \nu_k) T^0 + \eta_k T^{k-1} + \nu_k T^{k-2} + \tilde{\eta}_k \frac{\Delta t^n}{2} (D_T^{k-1} + S_T) + \tilde{\gamma}_k \frac{\Delta t^n}{2} (D_T^0 + S_T), \quad (38)$$

$$Y_s^k = (1 - \eta_k - \nu_k) Y_s^0 + \eta_k Y_s^{k-1} + \nu_k Y_s^{k-2} + \tilde{\eta}_k \frac{\Delta t^n}{2} (D_{Y_s}^{k-1} + S_{Y_s}) + \tilde{\gamma}_k \frac{\Delta t^n}{2} (D_{Y_s}^0 + S_{Y_s}), \quad (39)$$

$$\rho^k = \frac{P_0^n}{\mathcal{R}T^k \sum_{s=1}^{N_s} \frac{Y_s^k}{W_s^k}}. \quad (40)$$

316 At the end of the integration, values at time-step $n + 1/2$ are given by

$$T^{n+1/2} = T^K, \quad (41)$$

$$Y_s^{n+1/2} = Y_s^K, \quad (42)$$

$$\rho^{n+1/2} = \rho^K. \quad (43)$$

317 Note that in order to save a significant amount of computational time, the thermodynamic pa-
 318 rameters \mathcal{D}_s , C_p and λ are not computed at each stage through the **CHEMKIN** routines, but
 319 extrapolated from previous values (see §3.4 for more details).

320 The coefficients appearing in all stages of integration are given by

$$\tilde{\eta}_1 = b_1 \omega_1, \quad (44)$$

$$\eta_k = \frac{2b_k \omega_0}{b_{k-1}}, \quad \nu_k = \frac{-b_k}{b_{k-2}}, \quad \tilde{\eta}_k = \frac{2b_k \omega_1}{b_{k-1}}, \quad \tilde{\gamma}_k = -a_{k-1} \tilde{\eta}_k, \quad (45)$$

321 where a_k, b_k, ω_0 and ω_1 are given by

$$a_k = 1 - b_k \mathcal{T}_k(\omega_0), \quad (46)$$

$$b_0 = b_2, \quad b_1 = \frac{1}{\omega_0}, \quad b_k = \frac{\mathcal{T}'_k(\omega_0)}{(\mathcal{T}'_k(\omega_0))^2}, \quad (47)$$

with

$$\omega_0 = 1 + \frac{\epsilon}{K^2}, \quad \omega_1 = \frac{\mathcal{T}'_k(\omega_0)}{\mathcal{T}''_k(\omega_0)}. \quad (48)$$

Note that in the remainder of the present paper, the value of $\epsilon = 10$ is chosen (see discussion at §4.4). Moreover, $\mathcal{T}_i(x)$ is the first kind Chebyshev polynomials:

$$\mathcal{T}_k(x) = 2x\mathcal{T}_{k-1}(x) - \mathcal{T}_{k-2}(x), \quad (49)$$

$$\mathcal{T}_0(x) = 1, \quad (50)$$

$$\mathcal{T}_1(x) = x. \quad (51)$$

322 3.2.3. Step 3: Stiff integration of reactive terms over a full time-step

323 During this step, the reactive terms R_T and R_Y are integrated over a full time-step Δt^n . These
324 terms involve the computation of chemical reaction rates that are very sensitive to temperature
325 and species mass fractions. Thus, it forms the following stiff system of ODEs:

$$\begin{aligned} T^{**} - T^* &= \int_{t^n}^{t^{n+1}} R_T dt, \\ Y_s^{**} - Y_s^* &= \int_{t^n}^{t^{n+1}} R_{Y_s} dt. \end{aligned} \quad (52)$$

326 The starting values are denoted by the superscript $*$ and are the ones computed at the end
327 of §3.2.2. Note that as the chemical reactions do not involve a variation of mass, the density is
328 kept constant during the whole integration stage, and is only updated at the end of the process,
329 denoted by the superscript $**$:

$$\rho^{**} = \frac{p_0^n}{\mathcal{R}T^{**} \sum_{s=1}^{N_s} \frac{Y_s^{**}}{W_s^{**}}} \quad (53)$$

330 Many stiff solvers are available in the literature to integrate Eq. (52). DVODE [44], RADAU5
331 [34] and SEULEX [34] have been implemented in the framework of **HOLOMAC**. All of these
332 solvers rely on relative and absolute tolerance values to reach convergence. Assessment of per-
333 formance and accuracy of these solvers is out of the scope of the present paper, and DVODE
334 is retained for the remainder of the present paper, with both relative and absolute tolerance val-
335 ues set to 10^{-14} , respectively. With such low values, it has been observed that all solvers give
336 virtually the same results.

337 *3.2.4. Step 4: RKC integration of scalars diffusion terms over the last half time-step*

338 This step is very similar to §3.2.2, using T^{**} , Y_s^{**} and ρ^{**} as initial values and the same
 339 number of iterations K . At the end of this step, T and Y_s are available at the new time t^{n+1} and
 340 boundary conditions for temperature and species mass fractions can be applied. Note that in the
 341 case of an outflow boundary condition, an Orlanski-type boundary condition [45] is employed to
 342 convect the physical variables out of the domain.

343 In the case of a closed domain, the new thermodynamic pressure p_0^{n+1} can be evaluated by
 344 applying Eq. (21). Hence, by knowing T^{n+1} , Y_s^{n+1} and p_0^{n+1} , the new density ρ^{n+1} , diffusion
 345 coefficients \mathcal{D}_s^{n+1} , λ^{n+1} and μ^{n+1} , as well as C_p^{n+1} can now be updated via the **CHEM**
 346 **KIN** routines.

347 *3.2.5. Step 5: Computation of the divergence velocity constraint*

348 Prior to the computation of the momentum equation, the divergence velocity constraint that
 349 will be applied during the pressure projection step can be computed with the final values at the
 350 end of §3.2.4. Similarly to §3.2.1, this velocity constraint could be discretised directly by apply-
 351 ing Eqs. (18) and (22), but in order to save computational time and avoid redundant evaluation
 352 of diffusion and reaction operators, the semi-discrete approach proposed by Yu et al. [28] is
 353 adopted:

$$\begin{aligned} \frac{\partial u_i^{n+1}}{\partial x_i} = & -\frac{1}{p_0^{n+1}} \left. \frac{dp_0}{dt} \right|^{n+1} + \frac{1}{T^{n+1}} \left((1 + \zeta_t^*) \frac{DT^{n+\frac{1}{2}}}{Dt} - \zeta_t^* \frac{DT^{n-\frac{1}{2}}}{Dt} \right) \\ & + \sum_{s=1}^{N_s} \frac{\bar{W}^{n+1}}{W_s} \left((1 + \zeta_t^*) \frac{DY_s^{n+\frac{1}{2}}}{Dt} - \zeta_t^* \frac{DY_s^{n-\frac{1}{2}}}{Dt} \right), \end{aligned} \quad (54)$$

354 with

$$\left. \frac{dp_0}{dt} \right|^{n+1} = \frac{\left[(\Delta t^n + \Delta t^{n-1})^2 - (\Delta t^n)^2 \right] p_0^{n+1} - (\Delta t^n + \Delta t^{n-1})^2 p_0^n + (\Delta t^n)^2 p_0^{n-1}}{\Delta t^n \Delta t^{n-1} (\Delta t^n + \Delta t^{n-1})}, \quad (55)$$

$$\frac{DT^{n+\frac{1}{2}}}{Dt} = \frac{T^{n+1} - T^n}{\Delta t^n} - \left[(1 + \zeta_t) C_T^n - \zeta_t C_T^{n-1} \right], \quad (56)$$

$$\frac{DY_s^{n+\frac{1}{2}}}{Dt} = \frac{Y_s^{n+1} - Y_s^n}{\Delta t^n} - \left[(1 + \zeta_t) C_{Y_s}^n - \zeta_t C_{Y_s}^{n-1} \right]. \quad (57)$$

355 Note that unlike the formulation of Yu et al. [28], Eqs. (54-57) include the possibility of
 356 time-step variations. In the above expressions, ζ_t is defined by Eq. (28) and ζ_t^* is expressed as:

$$\zeta_t^* = \frac{\Delta t^n}{\Delta t^n + \Delta t^{n-1}}. \quad (58)$$

357 *3.2.6. Step 6: RKC integration of momentum diffusion terms over a full time-step*

358 During this step, the momentum equation is advanced in time through the RKC integration
 359 of diffusion terms over a full time-step. Note that the source terms include the gradients of
 360 hydrodynamic pressure from the previous time-step. According to Guermond et al. [46], this
 361 technique leads to a correction in pressure of the velocity fields and increases the accuracy of

362 the projection method to second-order. Note also that the stages of integration are denoted $l =$
 363 $2, \dots, L$. The following relations are then employed:

$$(\rho u_i)^0 = (\rho u_i)^n, \quad (59)$$

$$(\rho u_i)^1 = (\rho u_i)^0 + \tilde{\eta}_1 \Delta t^n (D_{u_i}^0 + S_{u_i}), \quad (60)$$

$$(\rho u_i)^l = (1 - \eta_l - \nu_l) (\rho u_i)^0 + \eta_l (\rho u_i)^{l-1} + \nu_l (\rho u_i)^{l-2} + \tilde{\eta}_l \Delta t^n (D_{u_i}^{l-1} + S_{u_i}) + \tilde{\gamma}_l \Delta t^n (D_{u_i}^0 + S_{u_i}). \quad (61)$$

364 Similarly to the RKC integration of diffusion terms in the scalars equations described in
 365 §3.2.2, the density ρ and viscosity μ are interpolated from t^n and newly known t^{n+1} values (see
 366 §3.4 for more details).

367 At the end of this step, the velocity field does not satisfy the divergence constraint formulated
 368 in Eq. (16), and is at a provisional state denoted by the superscript *:

$$(\rho u_i)^* = (\rho u_i)^L. \quad (62)$$

369 The boundary conditions for momentum are applied at the end of this step. In the case of an
 370 outflow boundary condition, an Orlanski-type boundary condition [45] is employed to convect
 371 the momentum (ρu_i) out of the domain. Note that according to Gresho [47] the conservation of
 372 mass is enforced to machine precision accuracy by correcting the mass flux leaving the domain
 373 to exactly match the entering mass flux imposed at the inlet boundary condition.

374 3.2.7. Step 7: Pressure projection

375 In the fractional-step method, the final velocity field u^{n+1} is obtained by correcting the pro-
 376 visional velocity field u_i^* with the gradients of the hydrodynamic pressure, which is obtained by
 377 solving the following variable-coefficient Poisson equation:

$$\frac{\partial}{\partial x_i} \left(\frac{1}{\rho^{n+1}} \frac{\partial p'}{\partial x_i} \right) = \frac{1}{\Delta t^n} \left(\frac{\partial u_i^*}{\partial x_i} - \frac{\partial u_i^{n+1}}{\partial x_i} \right), \quad (63)$$

378 where $p' = p_1^{n+1} - p_1^n$. Note that the last term of the RHS has been computed in §3.2.5 using
 379 Eq. (54).

380 In the present paper, three methodologies based on a FFT spectral solver are investigated to
 381 efficiently solve the non-linear Poisson Eq. (63):

- 382 • **Method I: Fully implicit - iterative**

383 The basic brute-force approach suggested by Nicoud [48] is to solve Eq. (63) with the
 384 following iterative procedure:

$$\frac{\partial^2 p'_q}{\partial x_i^2} = \frac{\partial}{\partial x_i} \left[\left(1 - \frac{\rho_0^{n+1}}{\rho^{n+1}} \right) \frac{\partial p'_{q-1}}{\partial x_i} \right] + \frac{\rho_0^{n+1}}{\Delta t^n} \left(\frac{\partial u_i^*}{\partial x_i} - \frac{\partial u_i^{n+1}}{\partial x_i} \right). \quad (64)$$

385 Nicoud [48] recommends for the value of ρ_0^{n+1} a plane averaging of the density. However in
 386 the present algorithm, it was found that such a choice prevents the convergence of Eq. (64),
 387 which only converge for $\rho_0^{n+1} = \min(\rho^{n+1})$. Each sub-iteration q in Eq. (64) is solved

388 exactly using a spectral solver based on Fast Fourier Transforms. Eq. (64) is considered
389 converged when

$$\|p'_q - p'_{q-1}\| \leq \xi, \quad (65)$$

390 where $\|\cdot\|$ is the \mathcal{L}^2 -norm, and ξ is a tolerance parameter set by the user.

391 The provisional velocity field is then corrected by applying the following relation:

$$u_i^{n+1} = u_i^* - \frac{\Delta t^n}{\rho^{n+1}} \frac{\partial p'_q}{\partial x_i}, \quad (66)$$

392 where p'_q is the solution of the last iteration of Eq. (64).

393 As shown in §4.3, inaccuracy in the evaluation of the pressure leads to errors in the ve-
394 locity field when applying Eq. (66). Resolution of the projection/correction step up to the
395 machine precision then requires very low values for ξ , typically of the order of 10^{-12} , lead-
396 ing to a significant total number of iterations q , typically of the order of several hundreds.
397 Note that in order to accelerate the convergence, an initial starting value p'_0 is provided
398 with the help of the extrapolation $p^{n+1} = 2p^n - p^{n-1}$, so that:

$$p'_0 = (2p^n - p^{n-1}) - p^n. \quad (67)$$

399 • **Method II: Semi implicit - direct**

400 In order to save computational time, an interesting technique proposed by Dodd and Fer-
401 rante [49] is to split Eq. (63) into a variable part p'_0 provided explicitly by an extrapolation
402 of the pressure (see Eq. (67)), while the other term p' is solved within a constant-coefficient
403 Poisson equation requiring only one evaluation of the FFT spectral solver, i.e.

$$\frac{\partial^2 p'}{\partial x_i^2} = \frac{\partial}{\partial x_i} \left[\left(1 - \frac{\rho_0^{n+1}}{\rho^{n+1}} \right) \frac{\partial p'_0}{\partial x_i} \right] + \frac{\rho_0^{n+1}}{\Delta t^n} \left(\frac{\partial u_i^*}{\partial x_i} - \frac{\partial u_i^{n+1}}{\partial x_i} \right). \quad (68)$$

404 Of course Eq. (68) is very similar to Eq. (64) in method **I**. The difference in method **II** is
405 that the correction of the provisional velocity field now includes the contributions of both
406 the implicit and explicit pressure terms, i.e.

$$u_i^{n+1} = u_i^* - \Delta t^n \left[\frac{1}{\rho_0^{n+1}} \frac{\partial p'}{\partial x_i} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0^{n+1}} \right) \frac{\partial p'_0}{\partial x_i} \right]. \quad (69)$$

407 As demonstrated by Dodd and Ferrante [49], this method is efficient when the gradients of
408 pressure are smooth in time, which is not always the case in combustion applications. As
409 shown in §4.3 with a test case of autoignition of a heptane/air mixture, this method, while
410 the fastest, leads to errors of order 10^{-3} during the pressure correction step.

411 • **Method III: Semi implicit - iterative**

412 Method **III** is a mix of methods **I** and **II**. The pressure p' is computed with Eq. (64), while
413 the correction of the velocity field is obtained with the following variant of Eq. (69):

$$u_i^{n+1} = u_i^* - \Delta t^n \left[\frac{1}{\rho_0^{n+1}} \frac{\partial p'_q}{\partial x_i} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0^{n+1}} \right) \frac{\partial p'_{q-1}}{\partial x_i} \right]. \quad (70)$$

414 As shown in §4.3, this method is very efficient. Based on the two test cases investigated,
 415 the machine precision accuracy in the projection/correction step is achieved for values of
 416 the tolerance parameter ξ ranging from 10^{-8} to 10^{-6} , which represents a significant gain in
 417 the required total number of iterations and thus, computational time.

418 Once the final velocity field u_i^{n+1} is updated, the new hydrodynamic pressure p_1^{n+1} can be
 419 updated with the following relation:

$$p_1^{n+1} = p_1^n + p'_q. \quad (71)$$

420 At this step, the velocities u_i^{n+1} , temperature T^{n+1} and species mass fractions Y_s^{n+1} are spa-
 421 tially filtered (see §3.3) in order to remove small oscillations that could destabilise the algorithm.
 422 Hence, the computation of the next time-step can begin with §3.2.1.

423 3.3. Spatial discretisation and filtering

424 The domain of length l_x is discretised along the x axis by a uniform distribution of N_x nodes
 425 x_i , with $x_i = (i - 1)\Delta x$ for $1 \leq i \leq N_x$. In the present algorithm, the spatial derivatives are
 426 computed implicitly with high-order finite difference compact schemes. Given a generic function
 427 $f(x)$, the first derivative $f'(x)$ is computed with the following expression:

$$\alpha f'_{i-1} + f'_i + \alpha f'_{i+1} = a \frac{f_{i+1} - f_{i-1}}{2\Delta x} + b \frac{f_{i+2} - f_{i-2}}{4\Delta x}. \quad (72)$$

428 As shown by Lele [50], the coefficients $\alpha = 1/3$, $a = 14/9$ and $b = 1/9$ give a quasi-spectral
 429 sixth-order accurate approximation of the derivatives. Of course this procedure is the same along
 430 the y and z directions if the domain is 2D or 3D.

431 A particular issue concerns the computation with compact schemes of the diffusion operators,
 432 which can be defined in a generic way with the following expression:

$$\frac{\partial}{\partial x_i} \left(\mathcal{D} \frac{\partial \vartheta}{\partial x_i} \right), \quad (73)$$

433 where ϑ is the state variable and \mathcal{D} the associated diffusion coefficient. Two discretisation tech-
 434 niques can be employed to evaluate Eq. (73):

- 435 1. by taking the first derivative of ϑ , multiplying by \mathcal{D} and then taking again the first deriva-
 436 tive of the whole product;
- 437 2. by expanding Eq. (73) with the chain rule, leading to:

$$\frac{\partial \mathcal{D}}{\partial x_i} \frac{\partial \vartheta}{\partial x_i} + \mathcal{D} \frac{\partial^2 \vartheta}{\partial x_i^2}. \quad (74)$$

438 Under the compact scheme formulation, the second derivative that appears in Eq. (74) is
 439 approximated by the following relation:

$$\alpha f''_{i-1} + f''_i + \alpha f''_{i+1} = a \frac{f_{i+1} - 2f_i + f_{i-1}}{2\Delta x^2} + b \frac{f_{i+2} - 2f_i + f_{i-2}}{4\Delta x^2}, \quad (75)$$

440 where f_i'' represents the second derivative of a function $f(x)$ at a point x_i , and where the
 441 coefficients $\alpha = 2/11$, $a = 12/11$ and $b = 3/11$ have been chosen to ensure sixth-order
 442 accuracy with the similar spectral resolution as for the first derivative.

443 As pointed out by Cook and Riley [23], the second discretisation technique is free of grid-
 444 to-grid oscillations but the conservative form of the diffusion operator is not maintained. On the
 445 other hand, the first discretisation technique is conservative but develops grid-to-grid oscillations,
 446 which can however be removed by spatial filtering.

447 In the present algorithm, care must be taken in the choice of a discretisation technique. It
 448 has been found that the second discretisation technique leads to significant errors in mass con-
 449 servation when applied to the computation of the diffusion operators D_T and D_{Y_s} in the energy
 450 Eq. (6) and species mass fraction transport Eq. (7). However the first discretisation technique is
 451 problematic when applied to the diffusion operator D_{u_i} in the momentum equation Eq. (5), be-
 452 cause it introduces spatial oscillations in the velocity field prior to the pressure-projection step.
 453 Even with a spatial filtering at this step, numerical inconsistencies have been observed between
 454 $\partial u_i^* / \partial x_i$ and $\partial u_i^{n+1} / \partial x_i$ during the evaluation of the RHS of Eq. (63), leading to errors in the final
 455 velocity field. In summary, D_T and D_{Y_s} are evaluated with the discretisation technique 1, while
 456 D_{u_i} is evaluated with the discretisation technique 2.

457 Periodically at the end of a time-step, a spatial filtering is applied along all directions to the
 458 state variables T , Y_s and u_i . According to Lele [50], the spatial filtering operator can be done
 459 with the help of compact filters by applying the following relation:

$$\alpha \bar{f}_{i-1} + \bar{f}_i + \alpha \bar{f}_{i+1} = a f_i + \frac{b}{2} (f_{i+1} + f_{i-1}) + \frac{c}{2} (f_{i+2} + f_{i-2}) + \frac{d}{2} (f_{i+3} + f_{i-3}), \quad (76)$$

460 where \bar{f} is the filtered variable while the following coefficients represent a sixth-order low-pass
 461 filter:

$$a = \frac{1}{16} (11 + 10\alpha), \quad (77)$$

$$b = \frac{1}{32} (15 + 34\alpha), \quad (78)$$

$$c = \frac{1}{16} (-3 + 6\alpha), \quad (79)$$

$$d = \frac{1}{32} (1 - 2\alpha). \quad (80)$$

462 Note that $-0.5 < \alpha < 0.5$ is a parameter freely set by the user to control the spectrum of the
 463 filter. A value of $\alpha = 0.5$ implies no filtering, while reducing α moves the cut-off frequency to
 464 low-band values. Note that a value of $\alpha = 0.48$ has been chosen for all the simulations conducted
 465 in the present paper.

466 When non-periodic boundary conditions are imposed, single sided formulations are used for
 467 the approximation of the first and second derivatives. For the boundary nodes, the derivatives
 468 have the form

$$f'_1 + 2f'_2 = \frac{1}{2\Delta x} (-5f_1 + 4f_2 + f_3), \quad (81)$$

$$f''_1 + 11f''_2 = \frac{1}{\Delta x^2} (13f_1 - 27f_2 + 15f_3 - f_4), \quad (82)$$

469 that are third-order accurate, while at adjacent nodes the following fourth-order accurate formu-
470 lations are used:

$$\frac{1}{4}f_1' + f_2' + \frac{1}{4}f_3' = \frac{3}{2} \frac{f_3 - f_1}{2\Delta x}, \quad (83)$$

$$\frac{1}{10}f_1'' + f_2'' + \frac{1}{10}f_3'' = \frac{6}{5} \frac{f_3 - 2f_2 + f_1}{\Delta x^2}. \quad (84)$$

471 For spatial filtering with compact schemes, several formulations for points near boundaries can
472 be found in the review paper of Gaitonde and Visbal [51].

473 Note that the pressure is discretised on a staggered grid and interpolations and derivatives
474 are performed at mid-points via sixth-order compact schemes. Concerning the FFT solver, the
475 transposition between the physical space and the spectral space is performed under the so-called
476 *spectral equivalence principle*. Basically, this means that derivations/interpolations in the phys-
477 ical space lead strictly to the same results in the spectral space. Recall that the present **HOLO-**
478 **MAC** code is built upon the **Incompact3D** framework and all of these procedures are discussed
479 in detail in the reference paper [41].

480 3.4. Extrapolation/Interpolation of thermodynamic properties

481 The temporal integration of diffusion operators D_u , D_T , D_Y through the RKC stages requires
482 the knowledge of thermodynamic properties between two computational time-steps, say t^n and
483 t^{n+1} . Computing these properties with the **CHEMKIN** libraries is computationally costly. Re-
484 peated evaluation within the RKC stages would drastically increase the computational burden.
485 Najm and Knio [37] proposed to estimate the intermediate values via extrapolation and interpo-
486 lation procedures. It was also demonstrated that such approaches recover an overall second-order
487 accuracy in time. Thus, the direct evaluation of the properties with **CHEMKIN** libraries is only
488 performed once for each time-step at the end of step §3.2.4.

489 Denoting t^{RKC} as the intermediate time, i.e. $t^n \leq t^{RKC} < t^{n+1}$, the generic thermodynamic
490 parameter ψ is evaluated as follows:

- 491 • In steps §3.2.2 and §3.2.4, values at time t^{RKC} are extrapolated with known values at t^n and
492 t^{n-1} , i.e.

$$\psi^{RKC} = \psi^n + \frac{t^{RKC} - t^n}{\Delta t^n} (\psi^n - \psi^{n-1}), \quad (85)$$

493 where ψ refers to \mathcal{D}_s , C_p and λ .

- 494 • In step §3.2.6, values at time t^{RKC} are interpolated from known values at t^n and t^{n+1} , i.e.

$$\psi^{RKC} = \psi^n + \frac{t^{RKC} - t^n}{\Delta t^n} (\psi^{n+1} - \psi^n), \quad (86)$$

495 where ψ refers to μ and ρ .

496 **4. Results**

497 In this section the performance of the numerical methods developed in Sec. 3 is assessed.
 498 First, temporal and spatial convergence tests are performed for a freely propagating 1D methane/air
 499 flame. Two configurations are investigated:

- 500 • an open domain, where unburnt fuel/air mixture enters the domain at the intake boundary
 501 with the flame speed;
- 502 • a periodic closed domain where two flames propagate in opposite directions into the un-
 503 burnt mixture from a hot spot.

504 Both cases exhibit an overall second-order accuracy in time, validating the implementation of
 505 a Strang operator-split strategy and the pressure-projection procedure into a fractional-step al-
 506 gorithm. As expected, the algorithm in a periodic closed domain shows sixth-order accuracy in
 507 space, while the use of inflow/outflow boundary conditions in a open domain gives an overall
 508 spatial accuracy of order 4.5, due to the degradation of the precision of compact schemes near
 509 the boundaries. Following on, a second test problem concerning a 2D vortex interacting with a
 510 flame is reported in §4.2.

511 Another test problems are employed in §4.3 to evaluate the three methods presented in §3.2.7
 512 to solve the variable-coefficient Poisson equation. In a first test problem, a heptane/air mixing
 513 layer is established and the autoignition of the mixtures is investigated in a 1D closed domain.
 514 In this problem, time-evolving strong gradients of the physical variables are generated, leading
 515 to difficulties in the convergence of the pressure equation. It is shown that method **III** efficiently
 516 achieves machine precision accuracy while maintaining a low number of iterations when solv-
 517 ing the variable-coefficient Poisson equation for the pressure. Following on, the same study is
 518 extended to the 2D vortex/flame interaction test case.

519 In §4.4, numerical experiments are performed with the RKC method to study the influence
 520 of the number of integration stages on the stability behaviour. Finally, in §4.5, 3D simulations of
 521 a premixed turbulent flame in a lean methane/air mixture are reported, demonstrating the ability
 522 of the numerical algorithm to handle complex turbulent reacting flows.

523 In the remainder of the paper, results computed with the numerical methods detailed in Sec. 3
 524 are referred by **HOLOMAC**, which is the name of the software developed during the present
 525 study to perform the numerical simulations.

526 *4.1. Temporal and spatial convergence tests*

527 *4.1.1. One-dimensional freely propagating premixed methane flame in an open domain*

528 The physical problem consists of a freely propagating premixed methane (CH_4)/air flame
 529 in a one-dimensional domain. All flames are computed with the skeletal mechanism for lean
 530 methane-air developed by Sankaran et al. [52], composed of 17 species and 73 reactions. More-
 531 over, the fresh gas is a premixed mixture of methane and air, with a temperature set to 810 K
 532 and an equivalence ratio of $\Phi = 0.6$. Hence, $Y_{\text{CH}_4}^0 = 0.0338$, $Y_{\text{O}_2}^0 = 0.2252$ and $Y_{\text{N}_2}^0 = 0.741$.
 533 In the present simulation, the profile of a 1D methane/air premixed flame is computed with the
 534 reference code **CHEMKIN** [39], and imposed as an initial condition for the simulation with
 535 **HOLOMAC**. The relative and absolute tolerances for the **CHEMKIN** computation are set to
 536 10^{-12} and 10^{-8} , respectively. The total length of the computational domain is 0.04 m, and the
 537 mesh grid is progressively refined with an adaptative technique. At convergence, the solution is
 538 discretised on approximately 9000 points and the computed flame speed is $u_f = 0.3202 \text{ m}\cdot\text{s}^{-1}$.

539 For this configuration, the thermodynamic pressure p_0 is set to 2×10^6 Pa. The flame is kept
 540 stationary in the domain by setting the inflow velocity u_f to be the flame speed.

541 The premixed flame profile is interpolated from **CHEMKIN** into **HOLOMAC** on a domain
 542 of 2 mm, the flame front being located at 1 mm. In order to ensure a consistent initial condition to
 543 perform convergence test, a preliminary computation is performed to let the flame profile adapt
 544 to the mesh grid of the new solver. As the influence of numerical errors must be as small as
 545 possible, the domain is discretised with 2001 points, which represents a size between each grid
 546 points of $\Delta x \approx 1 \mu\text{m}$. Moreover, a very small time-step of $\Delta t = 1 \times 10^{-8}$ s, which corresponds
 547 to a convective CFL number of 0.008, is set to minimise splitting errors between convection-
 548 diffusion-reaction operators. The absolute and relative tolerances for the DVODE solver are
 549 both set to 10^{-14} , while method **III** is used to solve the variable-coefficient Poisson equation
 550 with a tolerance parameter ξ set to 10^{-10} . The number of stages for the RKC method is set to
 551 $K = L = 10$ for the integration of diffusion terms. Note that for an explicit integration of the
 552 diffusion terms without the RKC method, the von Neumann stability criterion for the diffusion
 553 would impose a theoretical maximum time-step of approximately 3.5×10^{-12} s, which is four
 554 orders below the selected time-step. At 1 ms of the physical simulation time, a solution S_{init} is
 555 stored.

556 The typical structure of the computed premixed methane flame is depicted in Fig. 1. The
 557 solid line represents the initial solution interpolated from the **CHEMKIN** simulation, while the
 558 dot symbols represent the solution S_{init} computed with **HOLOMAC**. Differences are very small.
 559 Note that the temperature, while not depicted, follows a trend similar to the velocity profile. The
 560 temperature of unburnt and burnt mixtures are 810 K and 2073 K, respectively. As shown in
 561 the bottom of Fig. 1, the influence of numerical errors is visible in the computation of radical
 562 species, for example typically $\mathcal{O}(10^{-6})$ for Y_{HCO} and Y_{H} . These results validate the ability of the
 563 algorithm and the numerical methods developed in the present paper to accurately simulate a
 564 reacting flow with detailed chemistry.

565 Convergence tests are evaluated with the \mathcal{L}^2 -norm of the difference between the computed
 566 and the reference solutions, which is expressed as follows:

$$\mathcal{L}^2(S_{sol} - S_{ref}) = \sqrt{\frac{(\phi_{sol} - \phi_{ref})^2}{N_x}}, \quad (87)$$

567 where subscripts *sol* and *ref* identify the computed and reference solutions, ϕ is the variable
 568 investigated, and N_x is the number of points of the mesh grid. The convergence rate is then
 569 computed by best-fitting the curve formed by successive \mathcal{L}^2 -norms.

570 Recall that S_{init} is the solution computed on the 2001 points grids ($\Delta x \approx 1 \mu\text{m}$) with a time-
 571 step of 1×10^{-8} s at 1 ms after the **CHEMKIN** profile is fitted on the grid. The procedures to
 572 perform convergence tests are as follows:

- 573 • for the spatial accuracy, the solution S_{init} is taken as the reference S_{ref} . The time-step is
 574 kept at 1×10^{-8} s and simulations are performed over a physical time of 0.5 ms on succes-
 575 sive mesh grids of 1001, 501, 251 and 151 points, which represents $\Delta x \approx 2, 4, 8, 13.25 \mu\text{m}$,
 576 respectively;
- 577 • for the temporal accuracy, the reference solution S_{ref} is formed by interpolating S_{init} on a
 578 mesh grid of 1001 points ($\Delta x \approx 2 \mu\text{m}$). Simulations are performed over a physical time of
 579 0.1 ms with successive time-steps of $\Delta t = 2, 3.125, 4, 6.25, 8 \times 10^{-8}$ s.

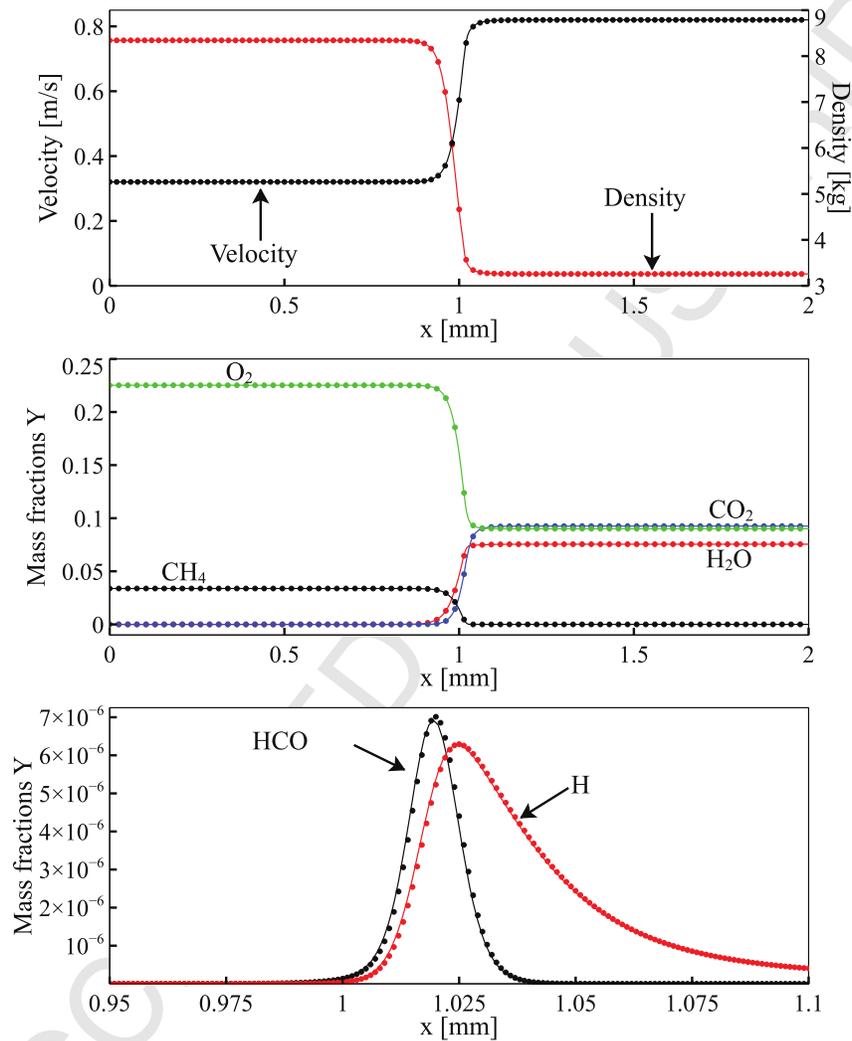


Figure 1: Structure of a steady one-dimensional premixed methane/air flame at 2×10^6 Pa and equivalence ratio of $\Phi = 0.6$. Solid line (—): initial solution interpolated from a **CHEMKIN** simulation. Symbols (\bullet): solution S_{mit} computed at 1 ms with **HOLOMAC** on a fine mesh ($\Delta x = 1 \mu\text{m}$) and a very small time-step ($\Delta t = 1 \times 10^{-8}$ s).

Variable	Global rates of convergence	
	Spatial	Temporal
T	4.6	2.27
u	4.57	2.27
p_1	4.52	2.18
ρ	4.78	2.27
Y_{CH_4}	4.45	2.30
Y_{O_2}	4.54	2.27
Y_{CO_2}	4.63	2.22
Y_{CO}	3.84	2.33
$Y_{\text{H}_2\text{O}}$	4.55	2.29
Y_{OH}	3.63	2.17
Y_{CH_3}	3.25	2.23
Y_{HCO}	2.81	1.60
Y_{H}	3.30	1.99

Table 1: Spatial and temporal global rates of convergence for the premixed 1D methane/air flame in an open domain.

Δt range (s)	T	u	p_1	ρ	Y_{CH_4}	Y_{HCO}
$(4 \rightarrow 6.25 \rightarrow 8) \times 10^{-8}$	1.66	1.66	1.65	1.66	1.66	1.31
$(3.125 \rightarrow 4 \rightarrow 6.25) \times 10^{-8}$	2.06	2.06	2.04	2.06	2.08	1.44
$(2 \rightarrow 3.125 \rightarrow 4) \times 10^{-8}$	2.40	2.40	2.33	2.40	2.43	1.63
$(1 \rightarrow 2 \rightarrow 3.125) \times 10^{-8}$	2.76	2.76	2.52	2.78	2.83	1.97

Table 2: Partial temporal convergence rates over a selected range of time-steps for the premixed 1D methane/air flame in an open domain.

580 Results are gathered in Table 1 for a selection of variables. The global spatial rate of conver-
581 gence is approximately $\mathcal{O}(4.5)$. This result was expected, because as described at §3.3 the order
582 of accuracy of the spatial discretisation schemes is degraded to third-order at boundary points and
583 fourth-order accuracy for adjacent points. Due to the implicit nature of compact schemes, despite
584 the sixth-order accuracy for the remaining interior nodes, this degradation at boundaries intro-
585 duces then an error $\mathcal{O}(3)$ in the whole domain that impede the global spatial rate of convergence.
586 With a balance between sixth-order and third-order accuracy for the discretisation schemes, an
587 overall accuracy $\mathcal{O}(4.5)$ can be expected, which is confirmed by results presented in Table 1. As
588 explained above, species with very small mass fractions are more sensitive to numerical errors
589 and it is observed that the global spatial convergence rate for such variables can be affected and
590 reduced to approximately $\mathcal{O}(3)$.

591 Furthermore, results reported in Table 1 show a temporal global rate of convergence of order
592 2. This result was expected and it validates that an implementation of a Strang operator-split
593 strategy and a pressure-projection procedure into a fractional-step algorithm ensure a second-
594 order accuracy in time. One can notice that the global temporal convergence rate of Y_{HCO} , an

intermediate species of low concentration, is only 1.60. Table 2 presents the temporal convergence rates for a selected ranges of time-steps. These results reveal that the order of accuracy increases as Δt is reduced. Moreover, the temporal convergence rate of Y_{HCO} reach second-order accuracy slower than other selected variables. This behaviour may be explained by the fact that HCO is a species sensitive to the evaluation of chemical reactions, and is more prone to be affected by errors introduced by the operator-split procedure. Of course such statement depends on the mechanism employed for chemistry and the configuration of the flow. In the context of methane/air flame, Safta et al. [27] also observed, with a similar chemistry mechanism (16 species and 46 reactions), the same behaviour for the temporal convergence of Y_{HCO} .

4.1.2. One-dimensional freely propagating premixed methane flame in a periodic closed domain

This test case is devoted to assess the spatial and temporal rate of convergence of the numerical methods when the domain is fully closed by periodic boundary conditions. Moreover, unlike the test case investigated previously in §4.1.1, the time derivative of the thermodynamic pressure (see Eqs. (27) and (55)) is now involved in the energy Eq. (6) and the divergence constraint expressed by Eq. (54). Thus, p_0 is expected to evolve. Although not detailed in the present paper, the algorithm with time evolving p_0 has been validated against **CHEMKIN** with a test case of autoignition of a homogeneous heptane/air mixture in a periodic, closed domain.

The initial solution S_{init} is obtained for the fresh premixed mixture of methane and air of equivalence ratio of $\Phi = 0.6$ as described at the beginning of Sec. 4.1. The imposed temperature follows a Gaussian profile:

$$T = T_{\text{fresh}} + T_{\text{amp}} \exp\left(-\left(\frac{1 - l_x/2}{\delta}\right)^2\right) \quad (88)$$

where $T_{\text{fresh}} = 810$ K is the temperature of the unburnt mixture and $T_{\text{amp}} = 1500$ K is the maximum amplitude of the Gaussian, $l_x = 20.48$ mm being the total length of the computational domain and $\delta = 0.5$ mm the width of the Gaussian. The initial temperature profile is depicted at the top of Fig. 2. Moreover, a velocity of $1 \text{ m}\cdot\text{s}^{-1}$ is uniformly imposed on the flow.

Figure 2 presents the flame structure for three physical times: $t = 0.50$ ms, $t = 1.50$ ms and $t = 2.50$ ms. Very shortly after the beginning of the simulation, the deposition of a hot temperature spot ignites the mixture, creating then two flame fronts propagating in opposite directions. Although not shown, the two flames merge at approximately $t = 3$ ms: the fresh premixed mixture is then totally consumed and only burnt gas remains in the domain.

The procedures to perform convergence tests are described as follow:

- for the spatial accuracy, the time-step is set to 1×10^{-7} s and simulations are performed over a physical time of 2.50 ms on successive mesh grids of 2048, 1536, 1024, 768, 512 and 384 points, which represents $\Delta x \approx 10, 13.3, 20, 26.6, 40, 53.3 \mu\text{m}$, respectively;
- for the temporal accuracy, a mesh grid of 1024 points ($\Delta x = 20 \mu\text{m}$) is used. Simulations are performed over a physical time of 2.50 ms with successive time-steps of $\Delta t = 1, 2, 4, 8, 12.5, 20 \times 10^{-7}$ s.

Note that for all computations performed to assess the convergence rates, the absolute and relative tolerances for the DVODE solver are both set to 10^{-14} , while method **III** is used to solve the variable-coefficient Poisson equation with a tolerance parameter ξ set to 10^{-10} . The number of stages for the RKC method is set to $K = L = 12$ for the integration of diffusion terms. This value

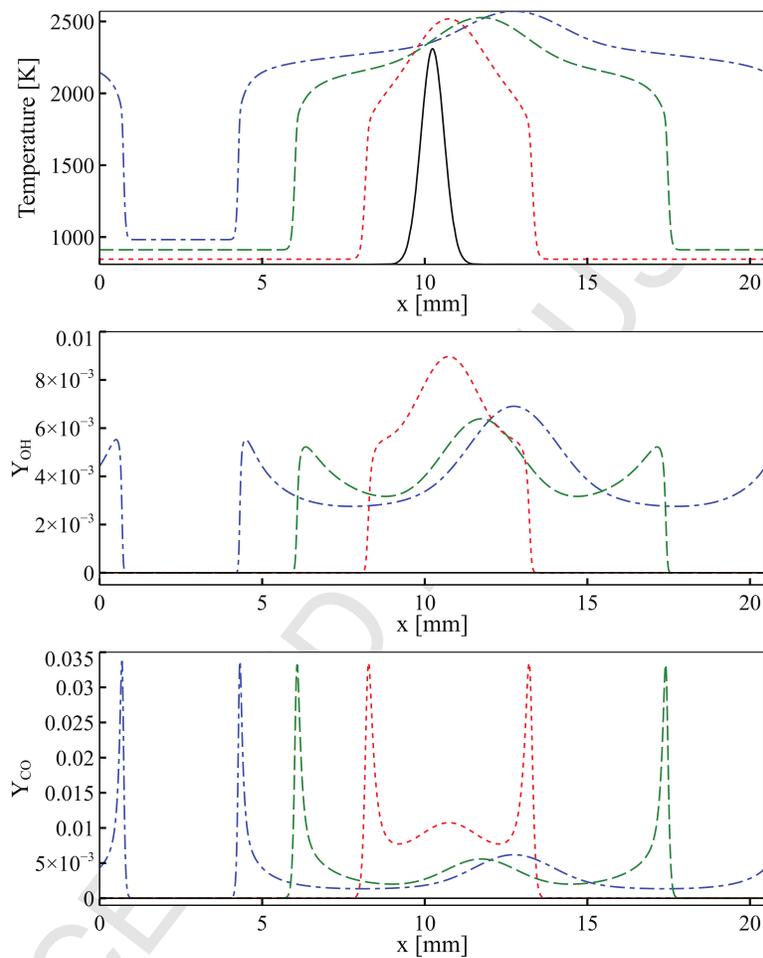


Figure 2: Structure of a freely propagating one-dimensional premixed methane/air flame in a closed domain at 101325 Pa and equivalence ratio of $\Phi = 0.6$. Top: temperature. Middle: Y_{OH} . Bottom: Y_{CO} . Solid line (—) initial solution at $t = 0$ ms. Dotted line (⋯): $t = 0.50$ ms. Dashed line (---): $t = 1.50$ ms. Dashed-dotted line (-.-.-): $t = 2.50$ ms.

635 ensures stability for the computation, which otherwise would become unstable for the largest
 636 time-steps (with $\Delta t = 20 \times 10^{-7}$ s and $\Delta x = 20 \mu\text{m}$). Indeed, for this configuration the von
 637 Neumann stability criterion for the diffusion would impose a theoretical maximum time-step of
 638 approximately 1.2×10^{-9} s. Moreover the CFL number reach 0.7 during the autoignition of the
 639 flame when the gases expand rapidly in either direction from the reacting zone. The CFL number
 640 of 0.7 is near the theoretical stability limit for explicit time integration.

Variable	Global rates of convergence	
	Spatial	Temporal
T	5.91	2.11
u	5.97	1.86
p_1	7.48	1.93
ρ	6.24	2.10
Y_{CH_4}	5.98	2.12
Y_{O_2}	5.89	2.11
Y_{CO_2}	5.99	1.98
Y_{CO}	5.81	2.07
$Y_{\text{H}_2\text{O}}$	5.98	2.11
Y_{OH}	5.73	1.97
Y_{CH_3}	5.34	1.99
Y_{HCO}	5.23	1.70
Y_{H}	5.61	1.95

Table 3: Spatial and temporal global rates of convergence for the premixed 1D methane/air flame in a periodic, closed domain.

641 Results are gathered in Table 3 for a selection of variables. The global spatial and temporal
 642 rates of convergence are approximately sixth-order and second-order, respectively. As the spatial
 643 discretisation is performed with sixth-order compact schemes everywhere in the computational
 644 domain, this result was expected. Moreover the addition of the time derivative of thermodynamic
 645 pressure into the energy equation and the divergence constraint does not affect the temporal
 646 accuracy. Note that like the test case investigated in §4.1.1, the same issues for mass fractions of
 647 small amplitudes such as Y_{HCO} are observed in terms of convergence rates.

648 All convergence tests performed within this section used method **III** with a tolerance pa-
 649 rameter ξ set to 10^{-10} to solve the variable-coefficient Poisson Eq. (63) up to machine precision
 650 accuracy. The properties of the three methods proposed at §3.2.7 as well as the influence of the
 651 tolerance parameter ξ are now investigated in the following section.

652 4.2. 2D Vortex-Flame interaction

653 This test case aims to simulate the interaction between a pair of vortices and a flame front.
 654 Similar cases have been studied in the literature but with different fuel, flow configuration and
 655 numerical methods [53, 26, 37, 54, 28]. As recalled by Lessani and Papalexandris [54], quan-
 656 titative comparisons are difficult to establish because the aforementioned works do not provide

657 sufficient information to exactly reproduce the same flame. However, this case represents a good
 658 numerical test to assess the implementation in 2D, as well as the stability behaviour of the code
 659 when vortices are leaving the domain through the outflow boundary.

660 The simulations are carried out in a rectangular domain of length $l_x = 4$ mm and height
 661 $l_y = 2$ mm. The freely propagating 1D methane/air flame, which has been previously computed
 662 in §4.1.1, is imported as an initial condition, and flow parameters as well as boundary conditions
 663 are unchanged. The position of the flame front is located at $x = 1$ mm, and the flame profile is
 664 homogeneously recopied along the y axis.

665 Two vortices are superimposed on the flow. One vortex can be described by the following
 666 expressions:

$$u_x = u_f + C \frac{y - y_0}{R^2} \exp(-r^2/2), \quad u_y = -C \frac{x - x_0}{R^2} \exp(-r^2/2), \quad (89)$$

where u_f is the initial velocity profile of the flame along the x axis, while x_0 and y_0 are the
 coordinates of the center of a vortex and C and R denote the strength and radius, respectively.
 Finally, $r = \sqrt{((x - x_0)^2 + (y - y_0)^2)}/R$, and in the present simulation $R = l_y/48$. If the subscripts
 1 and 2 refer to the clockwise and counterclockwise rotating vortices, respectively, their initial
 locations and parameter C are defined by

$$x_{01} = l_x/4, \quad y_{01} = l_y/2 - l_y/6, \quad C_1 = +0.3 \times 10^{-3}, \quad (90)$$

$$x_{02} = l_x/4, \quad y_{02} = l_y/2 + l_y/6, \quad C_2 = -0.3 \times 10^{-3}. \quad (91)$$

667 The inflow/outflow configuration is problematic, especially for the treatment of the hydrody-
 668 namic pressure. Indeed, as a zero pressure gradient is imposed at inflow and outflow boundaries,
 669 fluctuations leaving the domain through the outflow boundary may trigger non-physical oscil-
 670 lations of the hydrodynamic pressure in the whole domain, leading to a destabilisation of the
 671 simulation. It is well known that the design of a non-reflecting radiation condition at the outlet of
 672 a computational domain is difficult, especially in 2D/3D. There is an extensive literature on this
 673 subject but there is no consensus on resolving the challenges, and the proposition of a general
 674 form for open conditions is still under investigation [55]. In the present code, the 2D extension
 675 of the Orlanski outflow boundary condition proposed by Raymond and Kuo [56] was tested, but
 676 no real improvement in the stability was observed.

677 Furthermore, this stability problem can be efficiently overcome by imposing a sponge layer
 678 zone near the outlet, so as to damp fluctuations and virtually keep the flow variables constant
 679 at the boundary. The method proposed by Billson et al. [57] was implemented and successfully
 680 tested. Basically a source term is added to the governing equations:

$$\frac{\partial \mathbf{Q}}{\partial t} = \dots - \frac{\sigma(x)}{\Delta x} (\mathbf{Q}^n - \bar{\mathbf{Q}}^n), \quad (92)$$

681 where

$$\sigma(x) = \sigma_{\max} \left(\frac{x - x_0}{x_{\max} - x_0} \right)^2, \quad (93)$$

682 and

$$\bar{\mathbf{Q}}^n = \varpi \bar{\mathbf{Q}}^{n-1} + (1 - \varpi) \mathbf{Q}^n. \quad (94)$$

683 In the above equations, \mathbf{Q} represents the solution vector and $\sigma_{\max} = 1$ in the present sim-
 684 ulation, while the beginning and the end of the sponge region is bounded by $x_0 = 3$ mm and

685 $x_{\max} = 4$ mm, respectively. The term $\bar{\mathbf{Q}}^n$ represents a target field where the computed solution
686 should approach. A value can be provided, but it is however difficult to guess. In the present sim-
687 ulation, this term is constructed by time-averaging of the solutions over the first 50 time-steps.
688 During this process, $\varpi = 0.99$. For the remaining of the simulation, $\varpi = 1$ so as to deactivate the
689 averaging process. Note that in the present simulation, the damping term described at Eq. (92) is
690 only applied to the momentum and temperature Eqs. (5-6), the species mass fractions being not
691 directly constrained but following the density and temperature evolution.

692 The present simulation was performed on a regular, cartesian mesh grid of 1024×512 points
693 in the x and y directions, respectively. A constant $\Delta t = 20 \mu\text{s}$ was selected, whereas the number
694 of RKC stages were set to $K = L = 5$. Both absolute and relative tolerances were set to 1×10^{-8}
695 for the implicit integration of the chemistry with DVODE, while $\xi = 1 \times 10^{-6}$ was set for the
696 resolution of the Poisson equation. The simulation was run over a physical time of 3 ms on 192
697 cpus, which represent a wall-clock time of 120 hours.

698 Results are shown in Fig. 3 at 4 instants after start of the computation. The mass fraction
699 of the species OH is employed to distinguish the flame front. The vorticity is depicted by the
700 contour lines, the white ones and black ones representing the clockwise and counterclockwise
701 rotations, respectively. At the beginning of the simulation, the vortices are naturally convected by
702 the flow inside the fresh mixture. As they come closer to the reaction zone, the reaction front is
703 stretched and flame-generated vorticity appears along the flame front. When crossing the flame,
704 the initial vortices are strongly dissipated, and then convected away from the flame by the burnt
705 mixture. This physical behaviour is qualitatively similar to other simulations reported in the
706 aforementioned references. Moreover, as evident in the shapes of the flame front, the symmetric
707 features of the flow with respect to the central axis at $l_y/2$ are perfectly reproduced, verifying
708 then the 2D implementation.

709 Note that the present case was also run with a coarser mesh of 512×256 points in the x
710 and y directions, respectively. The simulation was advanced in time over 7 ms so as to assess
711 the stability of the simulation when the vortices leave the computational domain. Although
712 not shown in the present paper, results show that computations with the sponge layer region is
713 efficient and that no instabilities arise either from the outflow boundary, or from the interface
714 between the beginning of the damping zone and the physical computational domain.

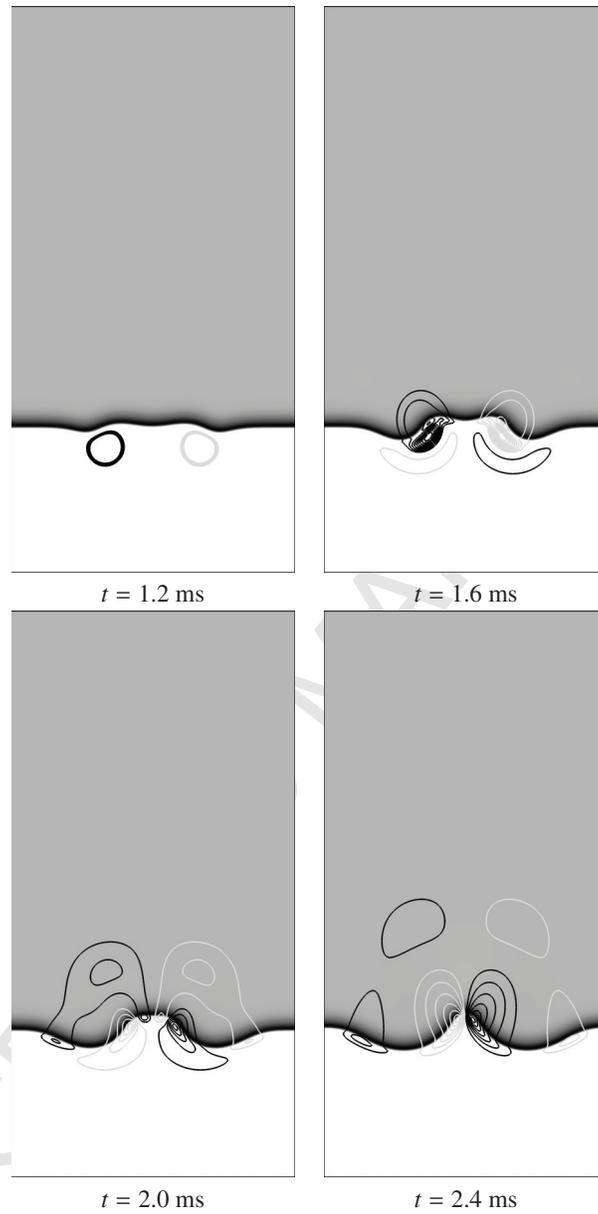


Figure 3: Simulation of the interaction of a vortices pair with a methane/air flame at $t = 1.2$ ms, $t = 1.6$ ms, $t = 2.0$ ms and $t = 2.4$ ms. The mass fraction of the species OH is represented to distinguish the flame front. The vorticity is depicted by the contour lines, the white ones and black ones representing the clockwise and counterclockwise rotations, respectively.

715 *4.3. Variable-coefficient Poisson equation: Assessment of resolution strategies*

716 The present section is devoted to the analysis of the three methods proposed in §3.2.7 to solve
 717 the variable-coefficient Poisson Eq. (63), as well as the influence of the tolerance parameter ξ .
 718 This study relies first on a test case of autoignition of mixing layers of heptane (C_7H_{16}) and air in
 719 a domain closed by solid walls. Then the study is extended to the 2D Vortex-Flame configuration
 720 previously reported in §4.2.

721 *4.3.1. One-dimensional autoignition of a heptane-air mixture in a closed domain.*

722 The configuration consists of a one-dimensional domain of length $l = 5$ mm. Walls are
 723 imposed at both ends, and the temperature and species mass fractions evolve freely, i.e. $u = 0$,
 724 $\partial T/\partial x = 0$ and $\partial Y_s/\partial x = 0$. The initial thermodynamic pressure p_0^0 is set to 4×10^6 Pa, while the
 725 mixing layers have an initial uniform temperature of $T^0 = 1000$ K and follow a species profile
 726 given by

$$Y_{C_7H_{16}}^0 = \frac{1}{2} \left(1 - \tanh \left((x - x_f) / \delta_f \right) \right), \quad (95)$$

$$Y_{O_2}^0 = 0.233 \left(1 - \frac{1}{2} \left(1 - \tanh \left((x - x_f) / \delta_f \right) \right) \right), \quad (96)$$

$$Y_{N_2}^0 = 1 - Y_{C_7H_{16}}^0 - Y_{O_2}^0, \quad (97)$$

727 where x is the coordinate of a mesh point, while $x_f = l/2$ and $\delta_f = 120 \mu\text{m}$ are the location and
 728 the thickness of the interface between heptane and air, respectively. Moreover, no initial velocity
 729 is imposed on the flow, hence $u^0 = 0$ everywhere.

730 The mesh grid is composed of 1001 points, which represents a space grid of $\Delta x \approx 5 \mu\text{m}$.
 731 The time-step is set to $\Delta t = 1 \times 10^{-7}$ s and the simulations are performed over a physical time
 732 of 0.3 ms. The number of stages for the RKC integration of diffusions terms are set to $K =$
 733 $L = 5$. Note that the von Neumann stability criterion for the diffusion would impose a theoretical
 734 maximum time-step of approximately 7×10^{-11} s. The absolute and relative tolerance parameters
 735 for DVODE are both set to 10^{-14} . Moreover in the present numerical set-up, the convective CFL
 736 reaches a maximum value of 0.02 at 0.2 ms when the velocity is maximum (see below). Such a
 737 low value is selected to ensure that no stability issues will arise during the study of the methods
 738 proposed to solve the variable-coefficient Poisson equation. Finally, the chemistry is described
 739 by a 37 species, 56 reactions mechanism proposed by Peters et al. [58].

740 The temporal evolution of the flow is depicted in Fig. 4 for a selection of variables and
 741 for three times during the autoignition process. From the beginning of the simulation until
 742 $t = 0.1$ ms, the fuel (heptane) and the air are mixing through laminar diffusion. As the ini-
 743 tial pressure and temperature are high, typical of diesel combustion engines, the mixture then
 744 auto-ignites and the temperature rises rapidly to values higher than 2000 K. Hence, as shown in
 745 Fig. (5) for $t = 0.2$ ms, a strong gradient of velocity is generated due to expansion. By about
 746 0.3 ms the temperature has stabilised at approximately 2800 K, and the peak velocity decreases
 747 to approximately $0.15 \text{ m}\cdot\text{s}^{-1}$.

748 The methods proposed in §3.2.7 are now investigated. The simulation of autoignition of
 749 heptane is repeatedly performed with methods **I**, **II** and **III**. For methods **I** and **III**, the tolerance
 750 parameter is varied with values $\xi = 1, 2, 4, 6, 8, 10, 12 \times 10^{-12}$, method **II** being without iterations.
 751 The solutions are the velocity fields taken at $t = 3$ ms and Eq. (87) is used to compute to the \mathcal{L}^2 -
 752 norm of the errors formed with the reference solution taken to be the solution obtained with

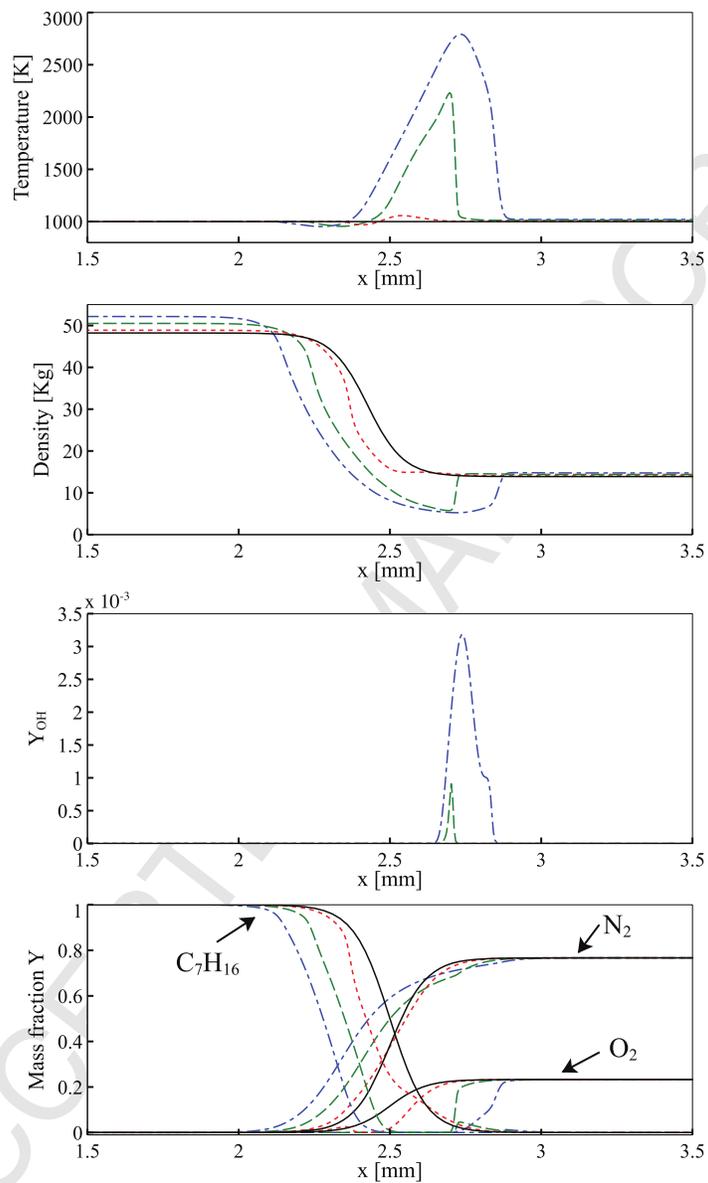


Figure 4: Structure of an autoignition sequence of a one-dimensional flame into mixing layers of heptane and air, at 4×10^6 Pa. Solid line (—) initial solution at $t = 0$ ms. Dotted line (⋯): $t = 0.1$ ms. Dashed line (---): $t = 0.2$ ms. Dashed-dotted line (-.-.-): $t = 0.3$ ms.

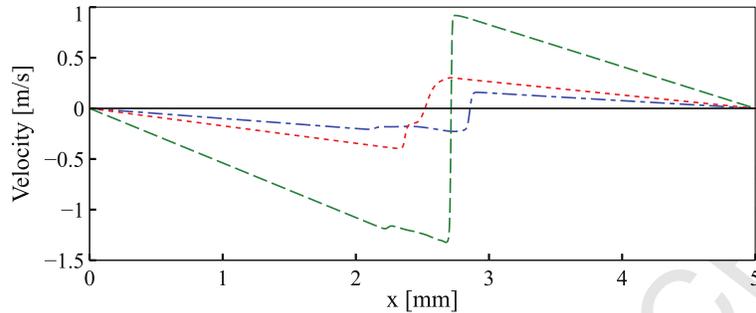


Figure 5: Velocity of the flow during an autoignition sequence of a one-dimensional flame into mixing layers of heptane and air, at 4×10^6 Pa. Solid line (—) initial solution at $t = 0$ ms. Dotted line (⋯): $t = 0.1$ ms. Dashed line (---): $t = 0.2$ ms. Dashed-dotted line (-.-.-): $t = 0.3$ ms.

753 method **I** and with $\xi = 10^{-12}$. This choice is justified by the fact that for $\xi = 10^{-12}$, both methods
 754 **I** and **III** give virtually the same solution.

755 Results are shown in Fig. 6. The methods **I**, **II** and **III** are represented by circle, cross and
 756 square symbols, respectively. Obviously method **II** is the least accurate as the error is $\mathcal{O}(10^{-3})$,
 757 which is far from the machine precision limit. Both methods **I** and **III** convergence to the machine
 758 precision accuracy for very low ξ values. However an interesting feature of method **III** is
 759 that the accuracy of $\mathcal{O}(10^{-12})$ is reached for values of ξ of $\mathcal{O}(10^{-6})$, while with method **I** such
 760 order of accuracy is only reached for ξ of $\mathcal{O}(10^{-12})$. This suggests that method **III** significantly
 761 accelerates the convergence when the pressure equation is solved iteratively.

762 As visible in Fig. (4) for the density and Fig. (5) for the velocity, there are strong gradients
 763 in the flow that evolve quickly in time, and with a density ratio of about 11. Hence, due to
 764 the dependance of the non-linear Poisson equation on the density, its resolution up to machine
 765 accuracy is difficult and require a significant computational effort. Indeed, as shown in Fig. 7,
 766 both methods **I** and **III** require the same number of approximately 250 iterations to solve the
 767 pressure Poisson equation up to machine accuracy. However choosing $\xi = 10^{-6}$ with method **III**
 768 requires about 100 iterations to reach the accuracy of $\mathcal{O}(10^{-12})$ during the projection-correction
 769 step. Note that the numbers of iterations presented in Fig. 7 are mean values computed over a
 770 whole simulation. It is also worth noting that this test case has been chosen to represents a flow
 771 under severe conditions, i.e. with transient ignition. In the context of the freely propagating
 772 methane/air flames presented at §4.1, solving the pressure Poisson equation with method **III** and
 773 $\xi = 10^{-10}$ requires only approximately 30 iterations.

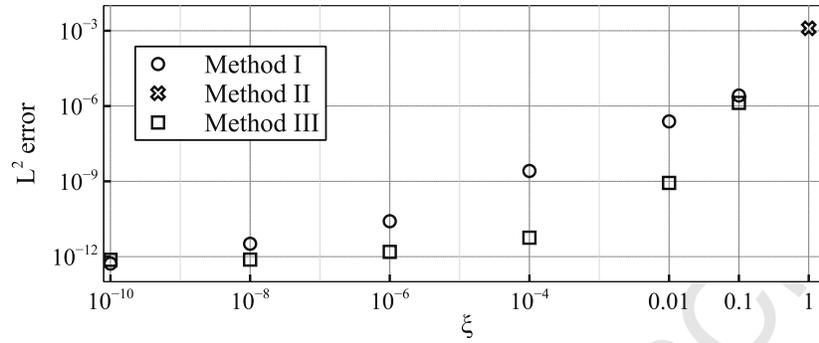


Figure 6: \mathcal{L}^2 norm errors of the velocity field at $t = 3$ ms formed by methods **I**, **II** and **III** respect to the tolerance parameter ξ for the heptane/air autoignition test case.

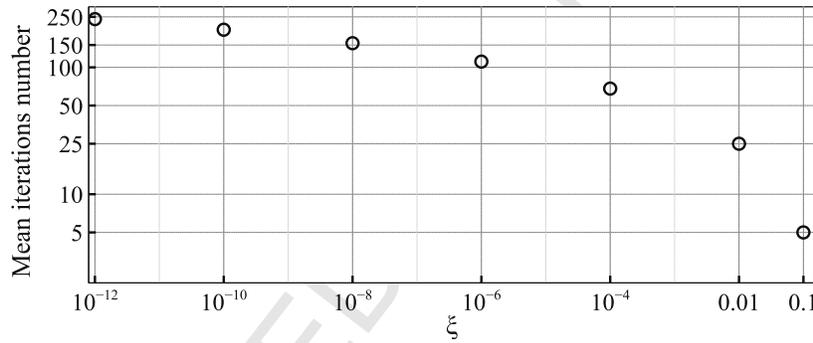


Figure 7: Mean number of iterations performed to reach the specified tolerance ξ for the heptane/air autoignition test case when methods **I** and **III** are chosen to solve the variable-coefficient Poisson Eq. (63).

774 4.3.2. Extension to the 2D Vortex-Flame test case

775 The simulation of the interaction between a pair of vortices and a premixed methane flame,
 776 previously reported in §4.2, is repeatedly performed with methods **I**, **II** and **III**. Similarly to
 777 the 1D study presented in §4.3.1, for methods **I** and **III**, the tolerance parameter is varied with
 778 values $\xi = 1, 2, 4, 6, 8, 10, 12 \times 10^{-12}$ and there are no iterations for method **II**. The solutions
 779 are the velocity fields taken at $t = 1$ ms when the pair of vortices interacts with the flame front.
 780 Equation (87) is used to compute to the \mathcal{L}^2 -norm of the errors formed with the reference solution
 781 taken to be the solution obtained with method **I** and with $\xi = 10^{-12}$.

782 Results are shown in Fig. 8. The methods **I**, **II** and **III** are represented by circle, cross and
 783 square symbols, respectively. Similarly to the 1D study (see §4.3.1), method **III** provides \mathcal{L}^2
 784 norm errors reduced by several orders of magnitude compared to method **I**. In the present case,
 785 the accuracy of $\mathcal{O}(10^{-12})$ is reached for values of ξ of $\mathcal{O}(10^{-8})$. However, contrary to the 1D study,
 786 the method **II** without iterations provides a better error compared to method **I** for $\xi > 0.01$. Of
 787 course, the required value of ξ to reach machine precision accuracy depends on the configuration

788 simulated and the numerical set-up. However it is emphasised that this value lies in the range of
 789 $O(10^{-8})$ to $O(10^{-6})$.

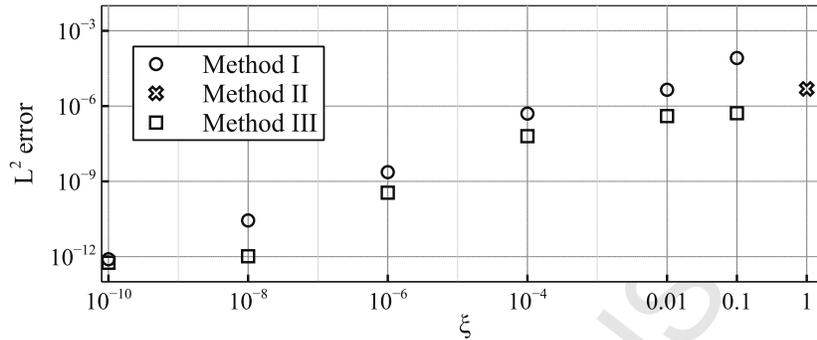


Figure 8: L^2 norm errors of the velocity field at $t = 1$ ms formed by methods **I**, **II** and **III** respect to the tolerance parameter ξ for the 2D Vortex-Flame test case.

790 4.4. Investigation of the RKC method

791 The damping parameter ϵ in Eq. (48) can be freely set by the user, but is critical to control the
 792 stability region of the method. As explained by Verwer et al. [38], if ϵ is very small, the stability
 793 region is extended along the negative real axis of the stability map but has a narrow strip with a
 794 sinusoidal shape. On the other hand, if ϵ is increased, the stability region becomes wider with
 795 an oval shape, but is shortened along the negative real axis. According to Verwer et al. [38], the
 796 choice of ϵ is crucial because the stability region must include the eigenvalues of both convection
 797 and diffusion; smaller values result in large imaginary parts whereas larger values result in large
 798 negative real part. Verwer et al. [38] recommend the following choices, depending on the kind
 799 of problem solved: $\epsilon = 2/13$ for strongly diffusion-dominated problems, $\epsilon = 5$ for convection-
 800 dominated problems, and $\epsilon = 10$ for mixed convection-diffusion problems. The attraction of the
 801 RKC method is that it has a stability interval that increases, respect to the number of iterations,
 802 with a quadratic behaviour. For very large values of ϵ , this quadratic behaviour is lost. However,
 803 for $\epsilon = 10$ the stability boundary $\beta(K)$ follows the relation $\beta(K) \approx 0.34(K^2 - 1)$. This means that
 804 even for a convection-diffusion problem, the quadratic behaviour of the RKC method is ensured.

805 As the algorithm developed in the present paper employs the RKC method to solve problems
 806 involving convection and diffusion operators of equivalent importance, a value of $\epsilon = 10$ is
 807 chosen. Hence, the stability depends on the number of stages K and L chosen for the RKC
 808 integration procedures. It is interesting to study the evolution of the minimum number of stages
 809 K and L required to ensure stability. The freely propagating 1D premixed methane/air flame
 810 in a periodic closed domain, detailed above in §4.1.2, is selected as a test case. Recall that
 811 $\Delta x = 20\mu$ m. Figure 9 presents the results for different values of Δt ranging from 6.25×10^{-8} s to
 812 2.75×10^{-6} s. Note that this latter value corresponds to a convective CFL of approximately 0.85
 813 and represents an upper-bound limit on the overall stability of the algorithm. Beyond this value
 814 the algorithm is found unstable whatever the choice of K and L .

815 A simple calculation based on the von Neumann stability criterion shows that a maximum
 816 time-step of approximately 2.75×10^{-8} s would be required to ensure stability. Recall that the
 817 minimum number of stages required for the RKC method is $K = L = 2$. This minimum value

818 ensures stability for time-steps below $\Delta t = 6.25 \times 10^{-8}$ s, which is the limit estimated from the
 819 von Neumann stability criterion. As depicted in Fig. 9, employing larger time-steps requires the
 820 number of stages of the RKC method to be increased. For a large time-step $\Delta t = 2.75 \times 10^{-6}$ s,
 821 $K = 14$ and $L = 12$ are needed to keep the stability of the algorithm.

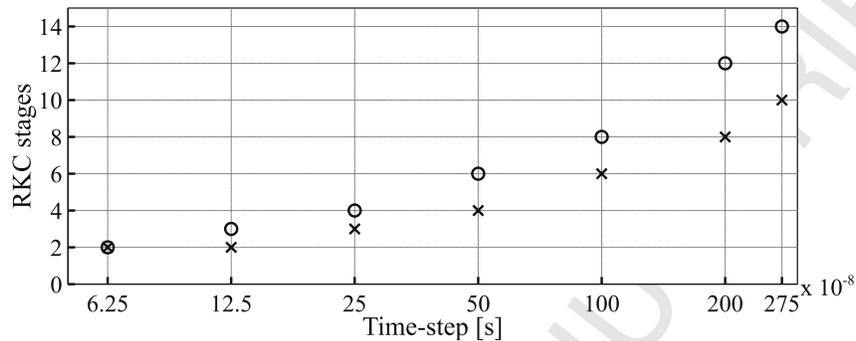


Figure 9: Minimum number of iterations of the RKC method to ensure stability for different values of Δt ranging from 6.25×10^{-8} s to 2.75×10^{-6} s. The configuration investigated is the freely propagating 1D premixed methane flame in a closed, periodic domain (see §4.1.2). Circle symbol (○): number of stages K . Cross symbol (×): number of stages L .

822 The present analysis demonstrates the power of the Runge-Kutta-Chebyshev method to per-
 823 form the integration of the diffusion terms. Only a few iterations are needed to ensure the stabil-
 824 ity of the algorithm, and for a range of time-steps up to about four orders larger than the critical
 825 time-step estimated by the von Neumann stability criterion. Note that although direct compar-
 826 isons cannot be performed because the test cases are not identical, the present paper shows that
 827 the number of iterations required by the RKC method is about an order less than the number
 828 of iterations reported by Yu et al. [28] who adopted a fractional step approach to integrate the
 829 diffusion operators.

830 4.5. 3D premixed turbulent flame in a lean methane/air mixture

831 **HOLOMAC** has been developed primarily for applications to DNS of low-Mach-number
 832 turbulent reacting flows with multistep kinetics. Figure 10 shows snapshots from three simu-
 833 lations of a premixed turbulent flame propagating in a three-dimensional domain in a lean
 834 methane/air mixture at an equivalence ratio of 0.6. Volume rendering of temperature is shown
 835 for three levels of turbulence intensity u' . When u' is normalised by the laminar flame speed S_L ,
 836 the corresponding values are 2, 5, and 10 in Figs. 10(a), (b), and (c), respectively. The domain
 837 is $12 \times 5 \times 5$ mm in size and uniform discretisation of $10 \mu\text{m}$ is employed. The flame thickness
 838 is of the order of about $100 \mu\text{m}$ and the Kolmogorov length scale for the highest turbulence in-
 839 tensity case is approximately $20 \mu\text{m}$. In other words, the Kolmogorov length scale and the flame
 840 are resolved. The figures are taken once the turbulent flame speed has reached a statistically
 841 steady state value. Detailed analysis of these and other results will be published separately. The
 842 objective of reporting these results here is only to demonstrate the capability of the code to cap-
 843 ture these complex turbulent flame structures. In particular, notice that as turbulence intensity is
 844 increased, there is increased wrinkling of the flame. This would, of course, result in increased
 845 turbulent flame speed. Also, notice that as turbulence intensity is increased, *islands* of burned

846 and unburned regions can be clearly identified. These islands are formed as a result of flame
847 extinction caused by turbulent strain. Subsequent to extinction, the mixture may locally reignite.
848 Accurate prediction of extinction, ignition and re-ignition often require multistep kinetics. It is
849 interesting to note that the highest absolute turbulence intensity values in these simulations are
850 about 2 m.s^{-1} and the highest turbulent flame speed is also about 2 m.s^{-1} . These speeds are suf-
851 ficiently low for this problem to lie in the low-Mach-number regime. **HOLOMAC** is well suited
852 for these fundamental studies carried out at low-Mach-number and requiring multistep kinetics
853 for accurate prediction of complex phenomena.

854 5. Conclusions

855 A novel and efficient algorithm with quasi-spectral accuracy has been presented in this paper
856 to conduct DNS of turbulent reacting flows with detailed chemistry under the low-Mach-number
857 assumption. The accuracy and efficiency of the algorithm have been assessed by employing dif-
858 ferent test problems. First, a freely propagating methane/air premixed flame was computed with
859 a detailed kinetic mechanism. Comparisons have been made with solutions from the reference
860 commercial software **CHEMKIN**, as well as self-convergence tests. It has been demonstrated
861 that a second-order accuracy is reached in time, which was expected, and this validates the imple-
862 mentation of an operator-split strategy. As the spatial discretisation is performed with high-order
863 compact schemes, it has been proven that the spatial convergence rate is sixth-order when the
864 domain is periodic, and between fourth and fifth-order accurate when walls or inflow/outflow
865 boundary conditions are employed. A 2D case consisting on the interaction between vortices
866 and a methane/air flame has been performed, demonstrating the ability of the algorithm to han-
867 dle vortical flow fields in the domain and at boundaries.

868 In the context of a fractional-step, projection-correction procedure, three different methods
869 have been proposed to resolve the variable-coefficient Poisson equation and to perform the cor-
870 rection in pressure of the velocity fields. An enhanced method, based on a mixed implicit-explicit
871 approach, has been proposed to speed-up the achievement of machine precision accuracy. Fur-
872 thermore, the convection-diffusion operators are treated with an explicit Runge-Kutta-Chebyshev
873 method. The performance of the method has been assessed and depending on the time-step, only
874 a few number of iterations are required to ensure numerical stability for large time-steps. Finally,
875 3D Direct Numerical Simulations of a premixed turbulent flame in a lean methane/air mixture
876 and for different turbulence level intensities are reported.

877 The code developed to implement the present algorithm has been called **HOLOMAC**, and
878 its efficiency opens the way to tackle DNS of reacting flows to understand complex turbulent and
879 chemical phenomena in flames.

880 Acknowledgment

881 The authors wish to thank Prof. Sylvain Laizet (Imperial College of London, UK) for his
882 valuable help about the **Incompact3D** framework and the **2DECOMP&FFT** libraries, Prof.
883 Vinicio Magi (University of Basilicata, Italy) for his comments through the development of the
884 present algorithm, Prof. Lawrence F. Shampine (Southern Methodist University, USA) for his
885 help on the implementation of the RKC method, as well as Dr. Muhsin Mameen and Zhiyan
886 Wang (Purdue University, USA) for providing validation test cases. The authors would also like
887 to thank eResearch South-Australia (eRSA) for providing the computational facilities and their
888 help on technical issues.

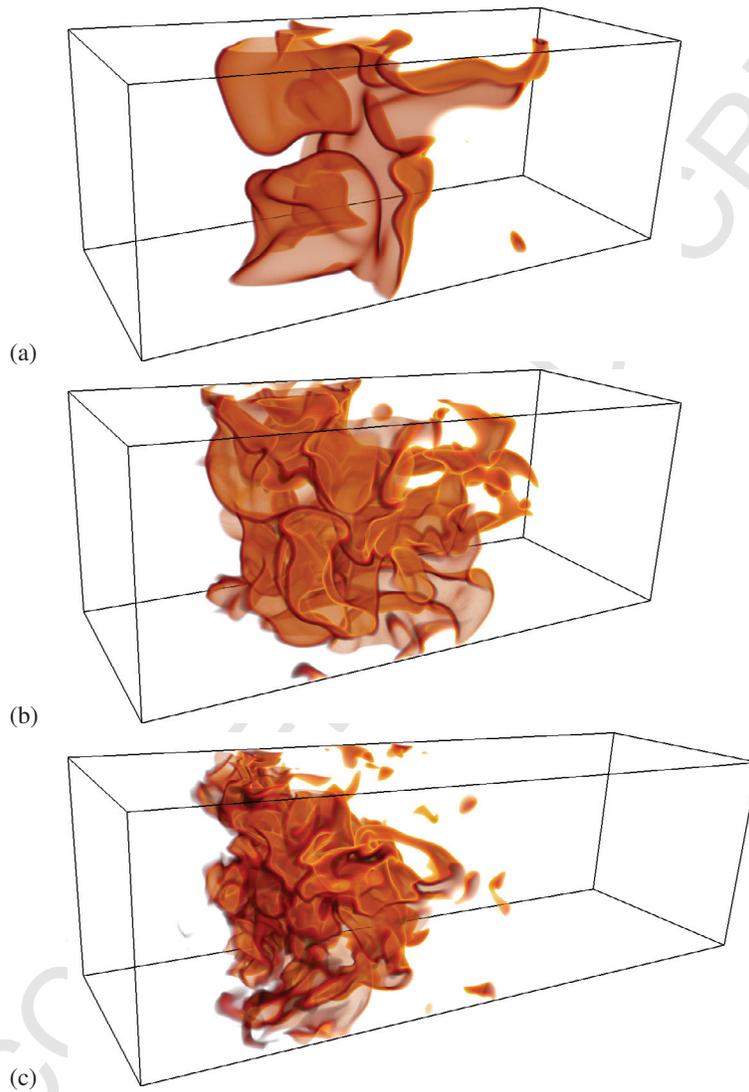


Figure 10: Volume rendering of temperature for 3D simulations of premixed turbulent flames in a lean methane/air mixture for normalised turbulence intensities u'/S_L of (a): 2, (b): 5 and (c): 10.

- 889 [1] N. Li, S. Laizet, 2DECOMP&FFT—A highly scalable 2d decomposition library and FFT interface, in: Cray User
890 Group 2010 conference, pp. 1–13.
- 891 [2] H. Pitsch, Large eddy simulation of turbulent combustion, *Ann. Rev. Fluid Mech.* 38 (2006) 453–482.
- 892 [3] L. Y. M. Gicquel, G. Staffelbach, T. Poinso, Large eddy simulations of gaseous flames in gas turbine combustion
893 chambers, *Prog. Energy Comb. Sci.* 38 (2012) 782 – 817.
- 894 [4] E. Motheau, F. Nicoud, T. Poinso, Mixed acoustic-entropy combustion instabilities in gas turbines, *J. Fluid Mech.*
895 749 (2014) 542–576.
- 896 [5] B. Franzelli, E. Riber, M. Sanjosé, T. Poinso, A two-step chemical scheme for Large-Eddy Simulation of kerosene-
897 air flames, *Combust. Flame* 157 (2010) 1364–1373.
- 898 [6] S. Mukhopadhyay, J. Abraham, Influence of heat release and turbulence on scalar dissipation rate in autoigniting
899 n-heptane/air mixtures, *Combust. Flame* 159 (2012) 2883 – 2895.
- 900 [7] J. de Charentenay, D. Thévenin, B. Zamuner, Comparison of direct numerical simulations of turbulent flames using
901 compressible or low-Mach number formulations, *Int. J. Numer. Meth. Fluids* 39 (2002) 497–515.
- 902 [8] G. Volpe, Performance of compressible flow codes at low Mach numbers, *AIAA Journal* 31 (1993) 49–56.
- 903 [9] E. Turkel, Preconditioned methods for solving the incompressible and low speed compressible equations, *J. Com-
904 put. Phys.* 72 (1987) 277 – 298.
- 905 [10] S. Venkateswaran, C. Merkle, Dual time stepping and preconditioning for unsteady computations, *AIAA Paper
906* (1995) 078.
- 907 [11] J.-S. Shuen, K.-H. Chen, Y. Choi, A coupled implicit method for chemical non-equilibrium flows at all speeds,
908 *J. Comput. Phys.* 106 (1993) 306 – 318.
- 909 [12] D. Lee, The design of local navier–stokes preconditioning for compressible flow, *J. Comput. Phys.* 144 (1998)
910 460 – 483.
- 911 [13] A. J. Chorin, A numerical method for solving incompressible viscous flow problems, *J. Comput. Phys.* 135 (1997)
912 118 – 125.
- 913 [14] F. Nicoud, Conservative high-order finite difference schemes for low-mach number flows, *J. Comput. Phys.* 158
914 (2000) 71–97.
- 915 [15] R. Knikker, A comparative study of high-order variable-property segregated algorithms for unsteady low Mach
916 number flows, *Int. J. Numer. Meth. Fluids* 66 (2011) 403–427.
- 917 [16] D. van der Heul, C. Vuik, P. Wesseling, A conservative pressure-correction method for flow at all speeds, *Comput.
918 Fluids* 32 (2003) 1113 – 1132.
- 919 [17] K. Nerinckx, J. Vierendeels, E. Dick, Mach-uniformity through the coupled pressure and temperature correction
920 algorithm, *J. Comput. Phys.* 206 (2005) 597 – 623.
- 921 [18] F. Cordier, P. Degond, A. Kumbaro, An asymptotic-preserving all-speed scheme for the euler and navier–stokes
922 equations, *J. Comput. Phys.* 231 (2012) 5685 – 5704.
- 923 [19] J. Doom, Y. Hou, K. Mahesh, A numerical method for DNS/LES of turbulent reacting flows, *J. Comput. Phys.*
924 226 (2007) 1136 – 1151.
- 925 [20] E. A. Sewall, D. K. Tafti, A time-accurate variable property algorithm for calculating flows with large temperature
926 variations, *Comput. Fluids* 37 (2008) 51 – 63.
- 927 [21] A. Majda, J. Sethian, The derivation and numerical solution of the equations for zero Mach number combustion,
928 *Combust. Sci. Tech.* 42 (1985) 185–205.
- 929 [22] P. A. McMurtry, W.-H. Jou, J. Riley, R. W. Metcalfe, Direct numerical simulations of a reacting mixing layer with
930 chemical heat release, *AIAA Journal* 24 (1986) 962–970.
- 931 [23] A. W. Cook, J. J. Riley, Direct numerical simulation of a turbulent reactive plume on a parallel computer, *J. Com-
932 put. Phys.* 129 (1996) 263 – 283.
- 933 [24] H. N. Najm, P. S. Wyckoff, O. M. Knio, A semi-implicit numerical scheme for reacting flow: I. stiff chemistry,
934 *J. Comput. Phys.* 143 (1998) 381 – 402.
- 935 [25] O. M. Knio, H. N. Najm, P. S. Wyckoff, A Semi-implicit Numerical Scheme for Reacting Flow: II. Stiff, Operator-
936 Split Formulation, *J. Comput. Phys.* 154 (1999) 428 – 467.
- 937 [26] M. S. Day, J. B. Bell, Numerical simulation of laminar reacting flows with complex chemistry, *Combust. Theory
938 and Modelling* 4 (2000) 535–556.
- 939 [27] C. Safta, J. Ray, H. N. Najm, A high-order low-Mach number AMR construction for chemically reacting flows,
940 *J. Comput. Phys.* 229 (2010) 9299 – 9322.
- 941 [28] R. Yu, J. Yu, X.-S. Bai, An improved high-order scheme for DNS of low Mach number turbulent reacting flows
942 based on stiff chemistry solver, *J. Comput. Phys.* 231 (2012) 5504 – 5521.
- 943 [29] V. Giovangigli, *Multicomponent Flow Modeling, Modeling and Simulation in Science, Engineering and Technol-
944 ogy*, Birkhauser, Boston, 1999.
- 945 [30] T. Poinso, D. Veynante, *Theoretical and Numerical Combustion*, Third Edition (www.cerfacs.fr/elearning), 2011.
- 946 [31] A. Ern, V. Giovangigli, Fast and accurate multicomponent transport property evaluation, *J. Comput. Phys.* 120
947 (1995) 105–116.

- 948 [32] T. E. Magin, G. Degrez, Transport algorithms for partially ionized and unmagnetized plasmas, *J. Comput. Phys.* 198 (2004) 424 – 449.
- 949
- 950 [33] J. O. Hirschfelder, C. F. Curtiss, R. B. Bird, *Molecular theory of gases and liquids*, John Wiley & Sons, New York, 1969.
- 951
- 952 [34] E. Hairer, G. Wanner, *Solving Ordinary Differential Equations II - Stiff and Differential-Algebraic Problems*. 2nd edition., volume 14, Springer-Verlag Berlin, 1996.
- 953
- 954 [35] G. Strang, On the construction and comparison of difference schemes, *SIAM J. Num. Math.* 5 (1968) 506–517.
- 955 [36] M. Duarte, S. Descombes, C. Tenaud, S. Candel, M. Massot, Time-space adaptive numerical methods for the simulation of combustion fronts, *Combust. Flame* 160 (2013) 1083 – 1101.
- 956
- 957 [37] H. N. Najm, O. M. Knio, Modeling low mach number reacting flow with detailed chemistry and transport, *J. Sci. Comput.* 25 (2005) 263–287.
- 958
- 959 [38] J. G. Verwer, B. P. Sommeijer, W. Hundsdorfer, RKC time-stepping for advection–diffusion–reaction problems, *J. Comput. Phys.* 201 (2004) 61 – 79.
- 960
- 961 [39] R. J. Kee, F. Rupley, E. Meeks, CHEMKIN-III: A FORTRAN chemical kinetics package for the analysis of gas-phase chemical and plasma kinetics, Technical Report SAND96-8216, Sandia National Laboratories, 1996.
- 962
- 963 [40] O. Desjardins, G. Blanquart, G. Balarac, H. Pitsch, High order conservative finite difference scheme for variable density low mach number turbulent flows, *J. Comput. Phys.* 227 (2008) 7125 – 7159.
- 964
- 965 [41] S. Laizet, E. Lamballais, High-order compact schemes for incompressible flows: A simple and efficient method with quasi-spectral accuracy, *J. Comput. Phys.* 228 (2009) 5989 – 6015.
- 966
- 967 [42] S. Laizet, N. Li, Incompact3d: A powerful tool to tackle turbulence problems with up to $O(10^5)$ computational cores, *Int. J. Numer. Meth. Fluids* 67 (2011) 1735–1757.
- 968
- 969 [43] M. Frigo, S. Johnson, The design and implementation of fftw3, *Proc. IEEE*. 93 (2005) 216–231.
- 970 [44] P. N. Brown, G. D. Byrne, A. C. Hindmarsh, Vode: A variable-coefficient ode solver, *SIAM J. Sci. Stat. Comput.* 10 (1989) 1038–1051.
- 971
- 972 [45] I. Orlanski, A simple boundary condition for unbounded hyperbolic flows, *J. Comput. Phys.* 21 (1976) 251–269.
- 973 [46] J. L. Guermond, P. Mineev, J. Shen, An overview of projection methods for incompressible flows, *Comput. Methods Appl. Mech. Eng.* 195 (2006) 6011 – 6045.
- 974
- 975 [47] P. M. Gresho, Incompressible fluid dynamics: Some fundamental formulation issues, *Ann. Rev. Fluid Mech.* 23 (1991) 413–453.
- 976
- 977 [48] F. C. Nicoud, Numerical study of a channel flow with variable properties, in: *Annual Research Briefs*, Center for Turbulence Research, NASA Ames/Stanford Univ., 1998, pp. 289–310.
- 978
- 979 [49] M. S. Dodd, A. Ferrante, A fast pressure-correction method for incompressible two-fluid flows, *J. Comput. Phys.* 273 (2014) 416 – 434.
- 980
- 981 [50] S. Lele, Compact finite difference schemes with spectral like resolution, *J. Comput. Phys.* 103 (1992) 16–42.
- 982 [51] D. V. Gaitonde, M. R. Visbal, High-order schemes for Navier-Stokes equations: algorithm and implementation into FDL3DI, Technical Report, DTIC Document, 1998.
- 983
- 984 [52] R. Sankaran, E. Hawkes, J. Chen, T. Lu, C. K. Law, Structure of a spatially developing turbulent lean methane–air bunsen flame, *Proc. Combust. Inst.* 31 (2007) 1291–1298.
- 985
- 986 [53] O. Colin, F. Ducros, D. Veynante, T. Poinso, A thickened flame model for large eddy simulations of turbulent premixed combustion, *Phys. Fluids* 12 (2000) 1843–1863.
- 987
- 988 [54] B. Lessani, M. V. Papalexandris, Time-accurate calculation of variable density flows with strong temperature gradients and combustion, *J. Comput. Phys.* 212 (2006) 218 – 246.
- 989
- 990 [55] S. Dong, J. Shen, A pressure correction scheme for generalized form of energy-stable open boundary conditions for incompressible flows, *J. Comput. Phys.* 291 (2015) 254 – 278.
- 991
- 992 [56] W. H. Raymond, H. L. Kuo, A radiation boundary condition for multi-dimensional flows, *Q. J. R. Meteorol. Soc.* 110 (1984) 535–551.
- 993
- 994 [57] M. Billson, L.-E. Eriksson, L. Davidson, Acoustic source terms for the linearized euler equations in conservative form, *AIAA Journal* 43 (2005) 752–759.
- 995
- 996 [58] N. Peters, G. Paczko, R. Seiser, K. Seshadri, Temperature cross-over and non-thermal runaway at two-stage ignition of n-heptane, *Combust. Flame* 128 (2002) 38 – 59.
- 997