



THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (Toulouse INP)

Discipline ou spécialité :

Dynamique des Fluides

Présentée et soutenue par :

M. THOMAS MARCHAL

le jeudi 20 avril 2023

Titre :

Extension de la méthode des Différences Spectrales à la combustion

Ecole doctorale :

Mécanique, Energétique, Génie civil, Procédés (MEGeP)

Unité de recherche :

Centre Européen de Recherche et Formation Avancées en Calcul Scientifique (CERFACS)

Directeur(s) de Thèse :

MME BÉNÉDICTE CUENOT

Rapporteurs :

M. MARIO RICCHIUTO, INRIA CENTRE DE RECHERCHE DE BORDEAUX

Membre(s) du jury :

M. ASHWIN CHINNAYYA, ENSMA POITIERS, Président

M. DANIEL MIRA, BARCELONA SUPERCOMPUTING CENTER, Membre

M. GUIDO LODATO, CORIA, Membre

M. JEAN-LUC ESTIVALEZES, ONERA TOULOUSE, Membre

Résumé

L'amélioration des outils d'ingénierie utilisés dans le design des dispositifs industriels de combustion est indispensable afin de respecter les demandes de plus en plus restrictives pour réduire les émissions de gaz à effets de serre. Parmi eux, la mécanique des fluides numériques (CFD) est devenue essentielle pour étudier et optimiser les chambres de combustion au cours des dernières décennies. Elle se complète parfaitement aux expériences réelles qui peuvent être très couteuses et avec lesquelles il est impossible d'obtenir des informations sur n'importe quelle quantité d'intérêt en tout point de la chambre de combustion. En utilisant les simulations aux grandes échelles (LES), la CFD décrit directement l'interaction entre les flammes et les structures turbulentes avec une faible modélisation. La qualité des résultats LES est ainsi très dépendante de la discrétisation utilisée incluant à la fois le maillage et également les propriétés de dissipation et de dispersion des méthodes numériques utilisées. Cependant, la plupart des codes LES employés de nos jours dans l'industrie utilisent des schémas de discrétisation spatiale de bas-ordre (LO) à cause de leur faible coût de calcul et leur facilité d'implémentation sur des maillages complexes. Pourtant, les méthodes numériques d'ordres élevés (HO) pour la LES sont développées depuis deux décennies et ont été appliquées sur des écoulements non-réactifs amenant à des résultats plus précis que les méthodes LO avec un plus faible coût de calcul. Bien que les méthodes HO semblent très prometteuses en combustion, en particulier pour mieux décrire le front de flamme, leur utilisation pour des écoulements réactifs restent encore à être démontrée. Au cours de ces travaux, les avantages et les bénéfices des méthodes HO en combustion sont évalués en utilisant la méthode des Différences Spectrales (SD) avec du raffinement hp .

Premièrement, il est démontré que la formulation originelle des SD est instable pour des écoulements multi-espèces avec des propriétés thermodynamiques variant avec la température et la composition. Il a été constaté que calculer les variables primitives aux points solutions puis de les extrapoler aux points flux, au lieu de faire l'inverse en extrapolant d'abord les variables conservatives, rend stable la méthode SD dans ce cas-ci. De plus, une nouvelle méthodologie, également plus stable pour calculer les flux diffusifs aux interfaces des cellules est détaillée. Enfin, les conditions aux limites caractéristiques et de murs ont été étendues aux écoulements multi-espèces dans le formalisme SD.

Avec ces développements, des flammes laminaires pré-mélangées 1D et 2D ont été simulées avec des mécanismes réduits à 2 réactions ou des mécanismes réduits analytiquement. Les résultats sont très proches de ceux obtenus avec des solveurs de référence bien établis en combustion. Il est montré que pour un même niveau d'erreur, il est plus efficace d'utiliser des maillages grossiers avec des grandes valeurs de p et non l'inverse. Par conséquent, le raffinement local en p , qui applique des grandes valeurs de p dans les régions d'intérêts seulement, permet de garder une bonne précision à un coût de calcul plus faible. Ceci est particulièrement intéressant pour des simulations de combustion où le front de flamme est très localisé et requiert une plus grande précision que le reste de l'écoulement. Il est également observé sur ces cas simples 1D et 2D que la méthode SD est moins sensible à la discrétisation du front de flamme que les solveurs volumes finis comme AVBP.

Pour terminer, deux différentes configurations de flammes 3D turbulentes ont été simulées avec l'algorithme des SD étendu aux écoulements réactifs. L'implémentation et la validation d'une méthodologie pour injecter de la turbulence dans le domaine de calcul ainsi que l'adaptation du modèle de flamme épaissie pour la LES dans le contexte des SD, 2 éléments nécessaires pour réaliser ces simulations, sont d'abord présentés. Les résultats ont démontré la capacité de la méthode SD proposée pour simuler des situations de combustion complexes et la positionnent comme un bon outil numérique pour le développement des futurs dispositifs industriels de combustion.

Mots-clés : Méthode d'ordre élevé, Méthode des Différences Spectrales, Combustion, Conditions aux limites caractéristiques pour Navier-Stokes, Injection de turbulence, Modèle TFLES.

Abstract

There is a strong necessity to improve engineering tools for designing the next generation of industrial combustion devices able to respect the always more restrictive demands for reducing and limiting greenhouse gas emissions. Among them, computational fluid dynamics (CFD) has become essential to study and optimize combustion chambers in the last decades. It is well complementary to experiments which may be more expensive and are unable to give information on any quantity of interest at any location. Using the large eddy simulations (LES) approach, CFD describes directly the interaction between flames and turbulent structures with limited modeling. The quality of LES results is then highly dependent on the discretization, including the mesh as well as the dissipation and dispersion properties of the numerical methods used. However, most industrial LES codes available today use low-order (LO) spatial discretization schemes due to their reduced computational cost and ease of implementation on complex meshes. Still, high-order (HO) numerical methods for LES have been developed in the two last decades, and successfully applied to non-reactive flows leading to more accurate results than LO methods at no extra computational cost. Although HO methods appear to be very promising for combustion, in particular to better describe the flame front, their application to reactive flows remains to be demonstrated. In the present work, the capacities and benefits of HO methods for combustion are assessed, using the Spectral Difference (SD) method with hp refinement.

Firstly, it is shown that the original SD formulation is unstable for multi-species gas flows, with thermodynamic properties varying with temperature and composition. It is found that computing primitive variables at solution points and extrapolating them at flux points, rather than conservative variables, make the SD method stable for such case. In addition, a new methodology, also more stable for computing the diffusive fluxes at cell interfaces is detailed. Finally, characteristic and wall boundary conditions are extended to multi-species flows in the SD framework.

With these developments, laminar 1D and 2D premixed flames are then simulated with two-steps or Analytically Reduced Chemistry mechanisms. Results are very similar to those obtained with well-established, reference combustion solvers. It is shown that for a given error level, it is more efficient to employ coarse grids with high p values and not the other way around. Consequently, local p -adaptation, which puts high values of p in regions of interest only, allows to keep a good accuracy at a lower computational cost. It is of particular interest for combustion simulations where the very localized flame front requires higher accuracy than the rest of the flow. It is also observed on these simple 1D and 2D cases that the SD method is less sensitive to the flame front discretization than finite volume solvers such as AVBP.

Finally, two different 3D turbulent flames configurations are carried out using the SD algorithm extended to reactive flows. The implementation and the validation of both a methodology to inject turbulence in the computational domain and the adaptation of the thickened flame model for LES within the SD framework, which are required to perform such simulations, are first presented. Results demonstrate the capability of the proposed SD method to simulate complex combustion situations and stands as a good numerical tool for the development of future industrial combustion devices.

Keywords: High-order method, Spectral Difference method, Combustion, Navier-Stokes characteristic boundary conditions, Turbulence injection, TFLES model.

Remerciements

C'est le moment préféré de la rédaction d'un manuscrit car on peut y écrire ce qu'on veut ! Je veux donc ici remercier toutes les personnes qui de près ou de loin m'ont permis de rester en bonne forme pendant cette thèse. Je m'excuse pour la longueur de ces remerciements mais je tenais à avoir un petit mot pour un maximum de personnes : c'est très important pour moi.

Je tiens d'abord à remercier les membres du jury pour avoir accepté d'évaluer mon travail de thèse. Merci beaucoup pour vos questions, remarques et conseils sur le manuscrit et également le jour de la soutenance. Ils m'ont permis d'améliorer la qualité de la version finale du manuscrit et aussi de prendre du recul sur ce que j'avais fait et c'était vraiment très plaisant.

La thèse c'est un travail d'équipe et donc ce travail n'aurait pas été possible sans l'aide de mes encadrants qui m'ont toujours laissé une grande liberté et autonomie dans mon travail. Donc je tiens à remercier tout d'abord ma directrice de thèse Bénédicte : merci beaucoup d'avoir apporté ton expertise et tes nombreux conseils sur la combustion, d'avoir également toujours maintenu le cap de la thèse notamment pour chercher à étudier l'intérêt des SD en combustion : je me souviens que ça ne m'a pas toujours rendu heureux, car j'ai mis du temps à comprendre ce que ça voulait dire, mais tu avais raison car beaucoup de personnes m'ont posé ces questions-là au final. Merci aussi d'avoir toujours pris le temps, quand j'en avais besoin, pour répondre à mes questions, corriger un papier ou le manuscrit ou pour m'aider sur les présentations notamment celle de la soutenance. Enfin, merci de t'être battu sur les papiers qu'on a soumis au symposium et en journal pour qu'on puisse être écouté avec notre nouvelle méthode dans ce petit monde de la combustion numérique. Ensuite je vais remercier mon co-directeur de thèse Hugues : sans qui je pense 50% de cette thèse n'aurait pas vu le jour. Hugues, il y a tellement de choses sur lesquelles j'ai envie de te remercier ! Merci infiniment pour ta présence au quotidien : tu as toujours été là je pouvais t'appeler n'importe quand tu répondais toujours à mes questions, peu importe qu'elles soient pertinentes ou non, tu ne m'as jamais jugé ou rabaisé car tu es d'une grande humilité. Du coup, ça m'a mis énormément en confiance car c'est rare d'avoir des personnes avec lesquelles on n'a pas « peur » de poser n'importe quelle question dans le travail ! Merci aussi pour ton soutien et tes encouragements dans les nombreux creux de la thèse : ça m'a redonné de la force ! En plus, on s'entend vraiment super bien (qu'est ce qu'on rigole bien ensemble haha) et pour moi ça va vraiment bien au-delà du travail : je te considère vraiment comme un ami. Merci pour ton expertise en numérique et en coding où tu m'as toujours impressionné et tu m'impressionneras toujours je pense : je suis fasciné par la façon que tu as de t'approprier les choses dans les papiers et de les développer pour tes applications et si ça ne fonctionne pas tu arrives toujours à trouver une explication ou une solution ! J'ai tellement appris à tes côtés et je suis super content que ça continue encore maintenant. Ta toujours grande curiosité fait que tu te poses énormément de questions et je pense que c'est ça qui fait ta force et selon moi fait que tu es un vrai chercheur. J'espère un jour arriver à ton niveau car tu resteras toujours un exemple pour moi. Ensuite à cause des règles de l'école doctorale, il n'était pas dans le jury mais il a été présent toute ma thèse comme encadrant c'est Jean-François. Merci beaucoup d'abord pour ton pragmatisme sur tout ce qu'on faisait : ça nous a vraiment permis de nous poser les bonnes questions de pas partir dans tous les sens par moment ! Merci pour ta franchise : c'est pour ça que j'ai toujours voulu ton avis sur les papiers, le manuscrit ou les présentations car si un truc n'allait pas tu me le disais sans langue de bois et très souvent tu avais raison ! Je savais que si tu me disais que c'était bien c'est que ça l'était ! Merci pour ton oeil aiguisé

sur les résultats physiques et pour comprendre quel paramètre numérique joue sur quel phénomène : ce qui est souvent difficile à identifier sur des cas 3D ! Enfin, merci pour ton partage de savoir sur GMSH et les techniques de maillage en général : cela m'a beaucoup apporté. Ensuite cette thèse n'aurait pas pu avoir lieu sans le financement de Safran Tech et donc je veux remercier aussi mes encadrants industriels : il y a eu d'abord Sophie pendant la 1ere année puis ensuite Renaud pour le reste de la thèse. Merci pour la confiance que vous m'avez accordée, pour votre suivi et vos questions/remarques pendant les points thèse et pour votre disponibilité dès que j'avais une question sur un outil Safran : c'était pas toujours simple à distance mais vous avez toujours répondu rapidement. Renaud merci également pour m'avoir accueilli quelques jours à Safran Tech à la fin de la thèse pour rencontrer et découvrir l'équipe à laquelle j'étais rattaché dont je remercie également les membres pour leur accueil. Enfin, merci d'avoir relu les articles et le manuscrit avec à chaque fois des commentaires/conseils qui ont permis d'améliorer la qualité de la rédaction !

Ensuite, il y a également 2 personnes qui n'étaient pas dans mon encadrement mais que je tiens à remercier particulièrement. D'abord Marc avait qui j'ai commencé sur JAGUAR il y a presque 5 ans maintenant en stage. Merci beaucoup d'avoir toujours pris du temps pour répondre à mes questions, souvent d'ordre informatique mais aussi théorique. Comme Hugues je savais que je pouvais te poser n'importe quelle question et que tes réponses seraient toujours réfléchies ! Ton expertise en informatique sur n'importe quel langage est vraiment très précieuse et m'a permis d'apprendre beaucoup de choses à tes côtés. Merci enfin pour ton aide (et surtout ta patience je pense haha) sur ANTARES qui m'a sorti d'énormément de galères de post-traitement souvent liées à ma mauvaise utilisation de l'outil. Enfin, je voudrais aussi remercier Guillaume Puigt à l'ONERA, un des papas de JAGUAR, à qui j'ai plusieurs fois demandé son avis sur des concepts en SD et qui a toujours pris le temps pour me répondre de manière détaillée et pertinente !

Merci également à Estelle Piot à l'ONERA pour avoir toujours fait en sorte que je travaille dans de bonnes conditions quand je venais sur site. Et merci à toutes les personnes de l'équipe STAT et plus généralement du DMPE avec lesquelles j'ai pu échanger ou poser des questions sur de nombreux sujets. Venir aux repas de Noel et aux barbecues d'unité et de département était toujours un plaisir car l'ambiance y est vraiment sympa. Côté CERFACS merci aux séniors de l'équipe CFD, notamment les combustionistes, pour votre partage d'expérience sur la combustion/chimie qui a été essentiel pour réaliser des simulations de combustion avec JAGUAR. Merci à toute l'équipe administrative, toujours là pour nous aider, Marie et Nicole que j'ai connu au début de mon aventure au CERFACS, Michèle, Chantal bien sûr, Lydia, Dominique, Brigitte, Philippine et Nathalie. Enfin merci à CSG, l'équipe info du CERFACS, pour la mise en place de la soutenance et aussi pour l'assistance très réactive sur les problèmes informatiques de toute sorte : c'est un vrai plus d'avoir l'équipe info directement sur place ! Donc merci à Fred, Fabrice, Gérard, Isabelle, Patrick et Nicolas. Mention spéciale à Isabelle pour les réponses à toutes les questions, souvent naïve de ma part, auxquelles elle a répondu ainsi qu'à Fabrice pour la préparation de la soutenance.

La thèse ça se passe aussi très bien grâce à l'ambiance qui règne au CERFACS et à l'ONERA avec les stagiaires, doctorants et post-doctorants.

D'abord au CERFACS ! Merci à la team LBM/Aero/Kitesurf : Gauthier qui m'a gentiment « parrainé » sur arXiv, Thomas Astoul pour qui les crémaillères sont des parcs d'attraction et dont l'expertise en dégonflage de porte reste inégalée, Florian qui est la seule personne que je connaisse pour qui le doliprane fait plus de mal que de bien, Thomas Gianoli pour les conseils en muscu car oui il faut faire confiance à ceux qui ont directement la salle de muscu chez eux, Catchi pour être venu à la soutenance et également pour nous donner beaucoup de fil à retordre sur cette magnifique application qu'est Mon Petit Gazon (MPG), Minh l'artiste du python et spécialiste de la tartiflette que je remercie également pour ses relectures et conseils sur mon anglais et à qui je souhaite bon courage pour sa soutenance qui arrive le 3 Juillet prochain, Adèle alias Miss Triangle ou Félintra pour m'avoir d'abord aidé dans JAGUAR (et aussi un peu GMSH) et pour nos nombreuses discussions sur le coding même

si très souvent les conversations sur les jacobiniennes déviaient sur des sujets très éloignés des SD ce qui est assez drôle et ne m'a jamais déplu ! J'espère qu'on pourra continuer à discuter de tout comme ça longtemps ! Paulo mon coach course/velo, dont les performances m'impressionnent et m'inspirent beaucoup, merci pour tes nombreux conseils et avis ainsi que pour ton partage de connaissance sur les post-traitement python et pour nos discussions MPG, Chrichri alias Faf pour tes coups de gueules saignants sur un trop grand nombre de choses mais qui me font bien rigoler à chaque fois ainsi que pour nos magouilles sur MPG qui se passent rarement comme prévues, Dave, le frrrrr, co-fondateur avec moi de "radio vocaux" pendant le confinement (qui m'a permis de le passer sereinement) : pas étonnant que tu sois avec une journaliste en fin de compte ! Toutes ces séances de muscu et de course qu'on a pu faire : c'était vraiment très motivant ! Je suis content qu'on arrive à se parler régulièrement malgré la distance : je ne désespère pas qu'on se retrouve un jour dans la même ville !

Merci à tous ceux passé par la team JAGUAR, dont je suis en train de devenir un ancêtre, qui grossit petit à petit et dans laquelle l'entraide est essentiel afin de dompter correctement celui qu'on nomme amicalement le félin ! Je remercie donc Romain avec qui j'ai pu échanger notamment sur les NSCBC en SD au début de ma thèse, Rémi pour avoir débroussailler le co-processing, Jean-François Monier pour avoir apporté le tripping virtuel et son pragmatisme sur les routines du code, Pierre qui avait commencé sa thèse sur les chocs en même temps que moi et avec qui j'ai beaucoup discuté avant qu'il n'arrête à la fin de sa 1ère année : je suis content de savoir que tu t'épanouis dans un autre boulot maintenant ! Christophe qui était en stage avec Hugues à l'ONERA avec qui j'ai pu échanger quelques idées sur les tools de JAGUAR et qui a réussi à faire des flammes de diffusion avec le code, Nadir pour son oeil de matheux sur les SD et son travail sur les chocs en triangles/tetra, Guillaume Daviller également pour les chocs et aussi pour sa persévérance à faire de l'acoustique avec le félin : ce qui n'est pas toujours simple ! Riwan pour m'avoir remis le nez dans les NSCBC au début de sa thèse (ce qui a même permis de trouver une erreur dans le code) et à qui je souhaite bon courage pour la fin de thèse l'année prochaine ! Eloïse pour ton aide précieuse sur tout ce qui touche à l'informatique dans n'importe quel langage et aussi pour tes qualités humaine d'altruisme, de gentillesse et de courage qui sont très inspirantes : continue de vivre comme tu en as envie car c'est ça le plus important finalement ! Arthur Berthelot qui fût mon premier stagiaire et sans qui je n'aurai pas présenter les résultats du VOLVO dans cette thèse : c'était très plaisant de travailler avec quelqu'un de motivé et de curieux comme toi ! Ne change rien de cette mentalité et je suis sûr que la thèse que tu as commencé cette année se passera très bien ! Arthur Colombié avec qui j'ai partagé les bancs de l'n7 et dont je remercie l'esprit critique sur ce qu'il y a dans le code et dont les nombreuses questions qu'il m'a posé m'ont fait réfléchir sur un tas de choses dans le code. Je suis très content que tu ai rejoins l'équipe des dresseurs ! Alexia pour avoir débroussailler et corriger l'implémentation du p -variable (et aussi du h -variable) sans qui je n'aurai pas pu faire de calculs de combustion avec un p non-constant ! Je te remercie aussi pour les nombreuses discussions qu'on a eu sur le code qui m'ont permis de mieux comprendre certaines choses et aussi d'identifier certaines erreurs. Enfin, merci encore à Marc, Hugues, Guillaume Puigt et Adèle qui font bien sûr partie de cette équipe de dresseurs et qui ont eux aussi pris du temps pour discuter avec moi du félin.

Merci à mes différents co-bureaux pour la super ambiance qu'on a toujours réussi à garder en J24 et l'aide qu'on s'apporte : je suis content qu'on se soit tous toujours très bien entendus malgré nos délires souvent bien perchés qui me font toujours autant rigoler ! Merci à Thibault, qui était dans le bureau quand je suis revenu avant la thèse, pour tous les délires qu'on a pu avoir, les lendemains de soirée difficiles et les discussions musique : bon courage pour ta fin de thèse et j'espère que tu réussiras à percer dans le monde de la techno car tu as un vrai talent ! Merci à mon Juanito national, que j'ai vu évolué pendant plusieurs années, pour ton humour deuxcroyable, tes anecdotes sur l'armée, les blagues Pom'Pote et à qui je suis content d'avoir appris qu'une orange se coupait en quartier comme une clémentine ! Je te dis merde pour la fin de la thèse également : je sais que quelque chose de bien sortira de tes 67 branches AVBP ! Merci à Louis pour son aide sur ANTARES et son investissement à l'EAC qui continue malgré sa migration à Airbus : j'attends avec impatience la crémaillère dans

ton nouvel appart ! Merci à Alexandre, le VRP d'AVBP, pour sa bonne humeur permanente, sa gentillesse et son altruisme à toujours être disponible pour aider les autres ! Je crains cependant qu'ils soient plusieurs dans sa tête et qu'un craquage complet du système en 3ème année de thèse n'est pas impossible : beaucoup d'aimants, de balles anti-stress et de chenilles seront nécessaires. Enfin, merci encore à Eloïse qui s'est très bien intégré à ce bureau de fou depuis 1 an ce qui est plus que louable !

Merci aux thésards/post-doc de la team combu/chimie/diphase pour vos conseils sur AVBP, ARCANE, sur les formalités administratives de la thèse, rédaction du manuscrit, préparation de la soutenance etc : votre expérience a toujours été très utile ! Lionel pour sa curiosité sur un grand nombre de sujets, Benjamin dont la soutenance est probablement la plus intéressante que j'ai vu jusqu'ici et qui m'a beaucoup inspiré, Antony pour les relais sur la sortie vélo où mon vélo ressemblait à une aile de kite, Aurélien mon partenaire de pétanque avec lequel je reste invaincu, Victor pour les discussions combustion, machine learning et les mêmes sur le CERFACS et AVBP (tradition qui s'est perdu d'ailleurs à mon grand regret...), Adrien qui m'a fait comprendre un jour que je confondais rhum et whisky et qui a éveillé ma curiosité sur l'ordinateur quantique : je te dis merde pour tes futurs projets là-dedans en tout cas, Etienne le Gone rarement content et dont les punchlines fracassantes m'ont toujours bien fait rire, Clovis pour les soirées jeux de société et les discussions politique autour de la faucille et du marteau, Guillaume Bogo pour tes passages après 19h dans le bureau, avec un pas dont je connais le rythme par coeur, pour nous partager avec humour tes soucis plasmatiques, Thomas Naess pour son aide sur ANTARES en début de thèse et ses connaissances en muscu, Yann pour m'avoir bien fait rigoler une certaine après-midi d'octobre 2021 en jouant au spikeball : je crois que sa hanche s'en souvient encore, Lukas pour ses conseils de drague pour les allemandes même si je reste encore un peu perplexe de sa stratégie, Nathanael pour sa connaissance pointilleuse du transport complexe et son humour qui irait très bien dans notre bureau J24 ! Je regrette cependant son manque d'analyse footballistique pour le résultat de la finale de la coupe de France. Félicia pour avoir toujours pris du temps pour répondre à mes questions sur la combustion hydrogène. Alexandre Dutka pour les discussions sur les SD qu'on a pu avoir et qui m'ont souvent rafraîchi la mémoire même si, comme avec d'autres, la discussion bifurque souvent vers tout autre chose que les SD. Luc alias Kelu qui est la première personne dont j'ai vu la soutenance (visiblement ça ne m'a pas fait peur) et qui m'a gentiment prêté du matos de kitesurf, Thibault Duranton pour les discussions kitesurf et sa gentillesse le jour de la soutenance pour avoir ramener quelques amis en voiture, Thomas Lesaffre également pour les relais dans la fameuse sortie vélo/kite et aussi pour ses performances monstrueuses avec l'EAC sans qui on prendrait sûrement plus de buts, Quentin pour sa disponibilité sur toutes les questions chimie/combustion avec ARCANE/Cantera/AVBP, Jonathan alias Jon avec qui j'ai commencé en stage et qui est toujours dispo, comme Quentin, pour répondre aux questions combustion et aussi administratives : sa gentillesse et son altruisme sont difficilement égalables, Thomas Lafarge pour ses conseils musculation, son humour que j'aime beaucoup avec lequel on ne peut pas s'ennuyer et son dévouement à l'ASM Clermont malgré les nombreuses finales perdues qui est une grande source d'inspiration ! Varun que j'ai appris à connaître au symposium l'été dernier et avec qui je regrette de ne pas avoir discuté plus tôt car on s'amuse beaucoup ensemble même si je dois dire que son budget banane est très embêtant pour les finances d'un voyage. Antoine Mouze mon expérimentateur préféré, bientôt converti au numérique et à latex on l'espère, toujours là pour dessiner une bêtise sur notre tableau ou faire des blagues téléphoniques, Théo pour m'avoir beaucoup aidé et conseillé sur les travaux de l'appart et aussi pour m'avoir prouvé qu'une enceinte peut tenir 6 mois d'hiver canadien et à peine 2h en Ariège, Nicolas Barléon pour son magret séché, sa connaissance de Fatal avec qui c'est un plaisir partager la scène et pour ce qu'on a vécu ensemble au Canada entre le Symposium et le roadtrip ensuite : ce sont vraiment des moments dont je me souviendrai longtemps ! Antoine Pestre pour son expertise sur les fêtes de Bayonne où j'ai passé de super moments (j'espère qu'on va reprendre ça vite) même si à un quart d'heure près on aurait dû pousser titine sur l'autoroute, son amour du canard, du saucisson et de la raclette sont aussi très impressionnants tout comme sa qualité à réaliser des ti punch traitres.

Merci à tous ceux passés par l'En Avant CERFACS, l'EAC, la prolifique et formidable équipe de foot du CERFACS avec qui c'est toujours un plaisir de jouer malgré quelques défaites, rarement méritées, qui forge le caractère de cette grande équipe pour qui on mourait sur le terrain !

Côté ONERA je remercie tous les stagiaires, doctorants et post-doctorants du DMPE que j'ai pu croisé quand j'étais de passage et avec qui j'ai toujours bien discuté de ce qu'on faisait ! Tout le monde m'a toujours bien intégré au groupe malgré le fait que je sois très rarement présent ! Merci notamment à Ludovic, Adrien, Victor, Lola, Félix, Thomas Alary, Morgane, Thomas Batmalle, Julien, Nico Urien, Boris avec qui je peux devenir rapidement « gné gné », et bien sûr mon Alexis !

Ensuite il y a aussi les amis hors thèse qui comptent énormément et dont une partie était là pour la soutenance ce qui me touche beaucoup alors que certains venaient de loin !

Je voudrais remercier tous les amis de Toulouse que j'ai pu rencontrer soit par l'n7 soit par le CERFACS ou l'ONERA pour les bouffées d'air que vous m'avez procuré et les soirées, week-end ou vacances que l'on a passé ensemble : j'espère qu'on continuera à se voir comme ça longtemps ! La plupart d'entre vous m'auront aidé pour les travaux dans mon nouvel appart, certains sont même revenus plusieurs fois, donc je vous remercie vraiment pour ça. Merci donc à Sophie : je ne désespère pas de t'apporter un jour une idée pour un brevet génial soit prête ! Pierre Daugan pour les discussions footballistiques, Marie à qui je souhaite bon courage pour la fin de thèse : tiens le coup ça en vaut la peine ! Charles : j'espère que le PSG va un jour passer ce cap..., Nico Urien pour les conseils précieux en post-traitement python et sur latex et aussi récemment les blablarun qui permettent de prendre des nouvelles, Marine qui a obtenu le titre de docteur en même temps qu'Antoine en ayant relu 10 fois plus le manuscrit de ce dernier que son jury, Kim pour sa rapidité à poster des story insta et pour m'avoir évité un nombre important de piqures pendant les randos au Canada cet été, Pauline pour être justement revenue 2 fois faire des travaux à l'appart mais aussi pour son popcorn maison qui est juste trop bon, Emeryk docteur professeur agrégé traileur pour les conseils sur la course, les parties de jeux de société et les échanges sur le FOOOOT, Adélie avec qui je rigolais bien quand elle était encore sur Toulouse avec Gauthier, Mai pour tes conseils sur les séries et pour tenir la baraque en vacances, Octavie l'entremetteuse Tinder et qui je pense est la seule personne qui peut me comprendre sur la difficulté d'apprendre le kitesurf, Magaly pour son côté artistique notamment pour les photos de profils et de paysages, Mathilde pour les soirées fin d'été dans sa campagne Toulousaine, ses idées pour rentrer au Café Oz et bien sûr les potins, Patrick pour ses bons plans bars et ses idées de bizutage en crémaillère, Tiffanie qui a toujours le sourire même quand j'avais loupé un rendez-vous prévu à l'avance pour son anniversaire, Mayeul que j'ai découvert récemment et qui prépare de super pâtisseries, Laura pour avoir essayé de m'aider à déchiffrer Adrien dans les jeux de société, Lauren pour m'avoir fait réaliser que je confondais droite et gauche, Luc pour ses connaissances viticoles, tennistiques et son appart toujours bien rangé (même si il ne me croit plus quand je dis ça m'enfin...), Ludivine avec qui je devrais un jour songer à discuter déco pour enfin embellir mon appart, Nico Cazard le meilleur motivateur pour faire des conneries en soirée et supporter indéfectible du TFC avec qui je suis très heureux d'avoir vécu la victoire du TFC au stade de France en coupe, Simon alias Rollo pour les astuces MPG, les explications des jeux de société et les paires de ski à 1 ski, Julie pour les parties de ping-pong en vacances et le décodage du Simon à codename, Camille pour les festivals, les soirées confinement/couvre-feu, les parties de fifa, la champions league et l'aide sur VSCODE/python : hâte que tu reviennes en France ! Gabriel alias Kikinou pour l'hospitalité à Paris quand j'en avais besoin et avec qui je rigole toujours autant même si j'ai l'impression que son manque de lactose ne lui permet plus de se lever à des heures raisonnables, Paulina pour nos discussions sur la culture mexicaine, Arthur le frerot ingénieur/plongeur/kitesurfeur et peut-être bientôt surfeur ? Pas étonnant avec tout ça que tu ne saches plus sur quelle planète nous sommes et que ta tête devienne toute carrée, Jana que je connais depuis cette année et qui m'a démontré qu'on ne pouvait pas guérir d'une grippe à l'aide de chartreuse : bon courage néanmoins pour l'opération au genou, Alexis pour m'avoir fait découvrir un nombre incalculable de jeux de société auxquels, il faut bien le reconnaître,

tu m'as très souvent bien battu ! Merci d'avoir proposé tous ces repas qu'on a pu se faire pendant la thèse où on discutait de tout : tu as toujours été une très bonne oreille pour moi et j'espère que cela continuera longtemps et enfin Enora pour toutes ces fois où j'ai rigolé en voyant ta réaction face à certaines blagues d'Alexis : l'expression de ton visage dans ces moments-là vaut vraiment de l'or et ne me lasse vraiment pas !

Je voudrai remercier maintenant les amis de Paris que je connais depuis l'enfance, l'adolescence ou mon séjour à Georgia Tech : merci beaucoup à vous tous pour les moments qu'on passe ensemble quand je remonte sur Paris ou quand on part en vacances c'est toujours un plaisir de passer du temps avec vous même si c'est moins qu'avant j'en suis conscient ! Merci à Nino pour avoir été un super guide pendant les randos qu'on a pu faire, Alexandra pour les heures à discuter ensemble de la life : ce qui m'a toujours fait plaisir, Alice pour la liberté de paroles qu'on a toujours eu ensemble et le partage de nos expériences sur un tas de sujets comme le sport, les voyages et les films/séries, Rodrigue, le nouvel amateur de techno, pour toutes les fois où j'ai pleuré de rire en jouant à des jeux de société avec toi où tu confondais symbole et position et aussi pour avoir essayé de me coder en python, Grégoire pour m'avoir transmis ta passion du basket (c'était pas gagné au départ) et avec qui j'ai toujours pris plaisir à faire du sport et sortir en soirée : quand reviens-tu en France ??? Antoine, mon historien de talent, pour ses podcasts de qualité sur l'Amérique du sud : bon courage pour la dernière année de thèse j'ai hâte qu'on fête ça, Elliot, monsieur poker face capable de jouer 8 parties de poker en même temps, merci pour ton humour qui me plaît toujours autant depuis plus de 12 ans maintenant : ne change rien stp ! Bastien, que je connais depuis la maternelle, qui a réussi à ne pas se tromper entre Toulouse et Marseille pour le jour de la soutenance comme ce jour où on devait atterrir à Porto mais que lui avait pris un billet pour Lisbonne... dur... Je suis vraiment très heureux qu'on réussisse à se voir plusieurs fois chaque années, depuis des années, malgré la distance : j'espère que ça continuera très longtemps ! Enfin, Davidou : l'as du volant, fan de Ferrari et grand amateur de liqueurs (c'est toujours un plaisir d'en déguster avec toi) ! Merci pour m'avoir aussi accueilli de nombreuses fois quand je remontais sur Paris et pour tous les moments qu'on a pu partager ensemble depuis les années de lycée et je sais que c'est loin d'être fini !

Enfin je voudrais remercier ma famille : ma soeur et mes parents pour la très bonne éducation qu'ils nous ont donné en nous transmettant les valeurs du travail, de l'écoute, d'altruisme, de gentillesse et pleins d'autres ! Merci de m'avoir embêté pour que je travaille à l'école alors que je sais que je n'étais pas toujours simple à gérer ! Je me souviens que vous me disiez que « je vous remercierai » et ben effectivement c'est le cas ! Merci d'avoir toujours tout fait pour qu'on ne manque vraiment de rien et de nous avoir ouvert au monde et aux autres en nous faisant beaucoup voyager ! Je suis très fier d'être votre fils et Hélène ton frère. Merci pour tous vos encouragements et votre soutien pendant cette thèse : ça m'a vraiment beaucoup aidé !

Contents

| | |
|--|--------------|
| Résumé | v |
| Abstract | vii |
| Remerciements | ix |
| Nomenclature | xviii |
| 1. Introduction | 1 |
| 1.1. Industrial context | 2 |
| 1.2. Large eddy simulations and high-order methods | 3 |
| 1.3. High-order methods for combustion | 7 |
| 1.4. A word on the lattice Boltzmann method | 9 |
| 1.5. Objectives of the thesis | 10 |
| 1.6. Outline of the manuscript | 10 |
| 1.7. Publications | 11 |
| I. Theoretical concepts and modeling of laminar and turbulent premixed combustion | 13 |
| 2. Laminar premixed combustion | 15 |
| 2.1. General definitions | 16 |
| 2.2. Chemical kinetics | 21 |
| 2.3. Choice of the chemical description | 23 |
| 2.4. Conservation equations for multi-species reacting flows | 26 |
| 2.5. Summary of this chapter | 33 |
| 3. Turbulent premixed combustion | 35 |
| 3.1. Simulation of turbulent flows | 36 |
| 3.2. Large Eddy Simulation of turbulent premixed combustion | 38 |
| 3.3. The Thickened Flame model for Large Eddy Simulations | 42 |
| 3.4. Summary of this chapter | 48 |
| II. The Spectral Difference method and its application to a mono-species and non-reacting gas mixture | 49 |
| 4. The Spectral Difference Method | 51 |
| 4.1. General principle in the one-dimensional case | 52 |
| 4.2. Formulation for hexahedral elements | 64 |
| 4.3. Riemann solvers for convective fluxes at interfaces | 71 |
| 4.4. Gradient computation in 3D for diffusive fluxes | 75 |
| 4.5. Local polynomial degree adaptation | 79 |
| 4.6. Temporal discretization | 81 |
| 4.7. Summary of this chapter | 84 |

| | |
|--|------------|
| 5. Boundary Conditions | 87 |
| 5.1. Navier-Stokes Characteristic Boundary Conditions for a multi-species non-reacting gas | 88 |
| 5.2. Symmetry and no-slip wall boundary conditions for a multi-species non-reacting gas | 96 |
| 5.3. Summary of this chapter | 100 |
| 6. Generation of turbulent inlet conditions in the context of the Spectral Difference method | 101 |
| 6.1. Turbulent flows in channels and pipes | 103 |
| 6.2. Time-average velocity profiles imposed at turbulent inlets | 110 |
| 6.3. Spectral description of turbulence | 111 |
| 6.4. Injection of turbulence using a synthetic random Fourier method | 115 |
| 6.5. Features to help reproduce near-wall turbulence | 118 |
| 6.6. Injection of turbulence through a periodic channel | 121 |
| 6.7. Turbulent wall-bounded channel flow | 129 |
| 6.8. Turbulent wall-bounded pipe flow | 137 |
| 6.9. Summary of this chapter | 144 |
| III. The Spectral Difference method and its application to a multi-species and reacting gas mixture | 147 |
| 7. Extension of the Spectral Difference method to simulate a multi-species and reacting gas mixture | 149 |
| 7.1. Handling contact discontinuities for a thermally perfect gas with the SD method | 150 |
| 7.2. One-dimensional laminar premixed flames | 155 |
| 7.3. Two-dimensional laminar premixed flames | 162 |
| 7.4. Evaluation of p -refinement on combustion cases | 167 |
| 7.5. Comparison between original and SDLIFT formulations for the diffusion scheme on 1D and 2D premixed flames using coarse meshes | 176 |
| 7.6. Summary of this chapter | 181 |
| 8. Simulations of turbulent premixed combustion using the Spectral Difference method | 183 |
| 8.1. TFLES model in the context of the Spectral Difference method | 184 |
| 8.2. Positivity-preserving limiter | 191 |
| 8.3. Simulation of the Cambridge flame burner | 193 |
| 8.4. Simulation of the VOLVO configuration | 204 |
| 8.5. Summary of this chapter | 215 |
| 9. Conclusion and perspectives | 217 |
| 9.1. Discussion | 217 |
| 9.2. Future work | 218 |
| Appendices | 219 |
| A. Useful calculations | 221 |
| B. Algorithms | 229 |
| C. Definitions and results on some families of orthogonal polynomials | 235 |
| D. Projection matrices for local polynomial adaptation | 239 |
| E. Coefficients of Runge-Kutta schemes used in this work | 245 |
| F. Diagonalization of the 3D Euler equations for a multi-species gas in isoparametric coordinates | 247 |
| G. Compute constants in VKP spectrum | 257 |

| | |
|---------------------------|------------|
| List of Figures | 260 |
| List of Tables | 268 |
| List of Algorithms | 269 |
| Bibliography | 271 |

Nomenclature

Acronyms

| | |
|-----------------|--|
| ρ UPY | Spectral Difference algorithm computing $\mathbf{Q} = (\rho, \mathbf{u}, P, Y_k)$ at SP from conservative variables at SP and then extrapolating \mathbf{Q} at FP. |
| AMR | Adaptative Mesh Refinement |
| ARC | Analytically Reduced Chemistry |
| AVBP | A Very Big Project |
| BR1, BR2 | Bassi and Rebay 1 and 2 diffusion schemes |
| CFD | Computational Fluid Dynamics |
| CFL | Courant-Friedrichs-Lewy number |
| CONS | Spectral Difference algorithm extrapolating conservative variables from SP to FP. |
| DG | Discontinuous Galerkin |
| DGSEM | Discontinuous Galerkin spectral element method |
| DNS | Direct Numerical Simulation |
| DOF | Degrees Of Freedom |
| ENO, WENO, TENO | Essentially non-oscillatory, weighted ENO and targeted ENO |
| EOS | Equation Of State |
| ERK | Explicit Runge-Kutta scheme |
| FD | Finite Difference |
| FE | Finite Element |
| FP | Flux Points |
| FR | Flux Reconstruction |
| FV | Finite Volume |
| GRC | Globally Reduced Chemistry |
| HIT | Homogeneous and Isotropic Turbulence |
| HLLC | Harten Lax and van Leer Contact Riemann Solver |
| HO | High-Order |
| HPC | High Performance Computing |
| HRR | Heat Release Rate |
| IC | I-continuous diffusion scheme |

| | |
|----------|---|
| IEA | International Energy Agency |
| JAGUAR | project of an Aerodynamic solver using General Unstructured grids And high order schemes |
| JANAF | Joint Army Navy Air Force |
| LBM | Lattice Boltzmann Method |
| LES | Large Eddy Simulations |
| LO | Low-Order |
| LW | Lax-Wendroff convection scheme |
| MPI | Message Passing Interface |
| NSCBC | Navier-Stokes Characteristic Boundary Conditions |
| NSE | Navier-Stokes Equations |
| PEA | Pre-Exponential Adjustment or Partial Equilibrium Analysis |
| PP | Passot-Pouquet |
| QSS | Quasi-Stationary State |
| RANS | Reynolds-Averaged Navier-Stokes |
| RHS | Right-Hand-Side |
| RK | Runge-Kutta |
| RMS | Root Mean Square |
| RT, SDRT | Raviart-Thomas and Spectral Difference RT |
| SD | Spectral Difference |
| SEDM | Spectral Element Dynamic Model |
| SGS | SubGrid-Scale |
| SP | Solution Points |
| SSP | Strong Stability Preserving |
| SV | Spectral Volume |
| TE | Truncation errors |
| TFLES | Thickened Flame model for LES |
| TKE | Turbulent Kinetic Energy |
| TTGC | Two-step Taylor Galerkin convection scheme version C |
| TUPY | Spectral Difference algorithm computing $\mathbf{Q} = (T, \mathbf{u}, P, Y_k)$ at SP from conservative variables at SP and then extrapolating \mathbf{Q} at FP. |
| TVD | Total Variation Diminishing |
| VKP | von Karman-Pao |
| WALE | Wall Adapting Local Eddy |

Greek Symbols

| | |
|-------------------------------|--|
| α, β | Coefficient of a RK scheme under Shu-Osher representation. α is also a constant used in VKP energy spectrum. β is also the exponent in Charlette's wrinkling factor model |
| $\alpha_{cold}, \alpha_{hot}$ | Fresh and burnt gas constants used to compute characteristic time τ_1 for the relaxation flame sensor in the TFLES model |
| γ | RK coefficients used to evaluate the solution at each stage with a low-order RK scheme |
| $\boldsymbol{\kappa}, \kappa$ | Wavenumber vector and its norm. κ can also be the iteration cost per DOF. |
| $\boldsymbol{\omega}$ | Rotational vector of \mathbf{u} |
| $\boldsymbol{\tau}$ | Viscous stress tensor |
| $\Delta\kappa_n$ | Difference between two consecutive turbulent modes which could be linear or logarithmic |
| Δt | Timestep |
| Δ | LES filter scale |
| δ_ν | Viscous length scale |
| Δ_e | Element characteristic size |
| δ_{ij} | Kronecker symbol between i and j |
| δ_L^0 | Laminar flame thickness based on temperature profile |
| δ_L^{blint} | Flame thickness based on Blint's formula |
| δ_r | Thickness of the reaction zone |
| Δ_{SP} | Characteristic size at a given SP |
| Δ_w | Element size in the normal direction to the wall |
| Δ_x | Mesh characteristic size in the general case without specifying a numerical method |
| $\dot{\omega}_{kj}$ | Reaction rate or net production rate of species k in reaction j |
| $\dot{\omega}_k$ | Reaction rate or net production rate of species k |
| $\dot{\omega}_T$ | Heat release rate |
| ϵ | Dissipation rate of turbulence |
| $\epsilon^{\dot{\omega}_T}$ | Error based on the heat release rate field in the 2D burner case |
| ϵ_P, ϵ_T | Minimum values allowed for pressure and temperature at SP when using the positivity-preserving limiter |
| η_κ | Kolmogorov's length scale |
| η_x, η_y, η_z | Metrics along η -directions (second line of J^{-1}) |
| Γ | Gamma function or function used in Charlette's wrinkling factor model or $1/(\tilde{\gamma} - 1)$ |
| γ | Heat capacity ratio |
| κ_e | Wavenumber associated to L_e in PP and VKP energy spectrums |
| κ_{Kol} | Wavenumber associated to Kolmogorov's length scale |
| Λ | Matrix of eigenvalues |
| λ | Thermal conductivity or eigenvalue or friction factor |
| $ \Omega_e $ | Volume of element Ω_e |

| | |
|---|---|
| $ \dot{\omega}_F _{1D}^{max}$ | Absolute maximum value of the fuel combustion source term estimated on a 1D flame |
| μ | Dynamic viscosity of the mixture |
| $\nabla_{\mathbf{x}}^2$ | Laplacian operator in physical space $\frac{\partial^2 \bullet}{\partial x^2} + \frac{\partial^2 \bullet}{\partial y^2} + \frac{\partial^2 \bullet}{\partial z^2}$ |
| $\nabla_{\mathbf{x}}, \nabla_{\xi}$ | Gradient operators in physical $\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)^T$ and isoparametric spaces $\left(\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}, \frac{\partial}{\partial \zeta}\right)^T$ |
| $\nabla_{\mathbf{x}\cdot}, \nabla_{\xi\cdot}$ | Divergence operator in physical $\frac{\partial \bullet}{\partial x} + \frac{\partial \bullet}{\partial y} + \frac{\partial \bullet}{\partial z}$ and isoparametric $\frac{\partial \bullet}{\partial \xi} + \frac{\partial \bullet}{\partial \eta} + \frac{\partial \bullet}{\partial \zeta}$ spaces |
| $\nabla_{\mathbf{x}} \times$ | Rotational operator in physical space |
| ν | Kinematic viscosity of the mixture |
| ν''_{kj} | Molar stoichiometric coefficient of species k in reaction j in the products |
| ν'_{kj} | Molar stoichiometric coefficient of species k in reaction j in the reactants |
| ν_{kj} | $\nu''_{kj} - \nu'_{kj}$ |
| Ω | Physical domain of study |
| Ω_e | Refer to an element of the physical domain Ω |
| ω_i | Weight associated to the i -th quadrature point ξ_i |
| $\omega_x, \omega_y, \omega_z$ | Components of the rotational vector of \mathbf{u} in x , y and z directions |
| \otimes | Tensor product operation |
| Φ | Fourier transform of the velocity correlation tensor |
| ϕ, ϕ_g | Local and global equivalence ratios of the mixture |
| Π | Constant pressure gradient in channel and pipe flows |
| ψ | Fictive species |
| ψ_0 | Target value of the fictive species source term, needed for the relaxation flame sensor in the TFLES model, when $\mathcal{S} > 0$ |
| ψ_n, σ_n | Phase and direction of the n -th Fourier mode associated to κ_n |
| ρ, ρ_k | Densities of the mixture and of species k |
| τ, τ_w | Total shear stress and wall shear stress. τ can also be the real time taken to simulate 1 ms of physical time. |
| τ_0, τ_1 | Characteristic times of the fictive species source term needed for the relaxation flame sensor in the TFLES model |
| τ_e | Large eddies turnover time |
| $\theta_e, \tilde{\theta}_e$ | Dimensionalized and non-dimensionalized sensor values of element Ω_e used in p -adapted simulations |
| $\theta_k, \theta_2, \theta_3$ | Parameters for limiting respectively species, pressure and temperature in a targeted element when using the positivity-preserving limiter |
| $\tilde{\gamma}$ | Effective heat capacity ratio or Roe's average of the heat capacity ratio |
| Ξ | Wrinkling factor |

| | |
|-----------------------------|---|
| ξ, η, ζ | Isoparametric coordinates |
| ξ_j, η_j, ζ_j | Coordinates of solution point j in 3D in the isoparametric domain |
| ξ_k | Atomic mass fraction of species k |
| ξ_k, η_k, ζ_k | Coordinates of flux point k in 3D in the isoparametric domain |
| ξ_x, ξ_y, ξ_z | Metrics along ξ -directions (first line of J^{-1}) |
| $\zeta_x, \zeta_y, \zeta_z$ | Metrics along ζ -directions (third line of J^{-1}) |
| a_{kj} | Number of atoms j in species k |

Indices and superscripts

| | |
|--------------------|--|
| * | Index or superscript referring to a quantity in star region when using HLLC scheme. For the superscript, it can also be a modified quantity after application of a NSCBC treatment or a prescribed value for the right state in symmetry and no-slip wall boundary conditions. |
| + | Superscript referring to a friction quantity |
| Δ | Index referring to a quantity taken at LES filter scale Δ |
| T | Superscript referring to the transpose operation |
| ' | Superscript referring to a fluctuating quantity |
| b | Superscript referring to burnt gases |
| bb | Index referring to a bluff-body quantity |
| C | Superscript referring to a continuous quantity |
| c, d | Indices referring to convective and diffusive fluxes |
| cf | Index referring to a quantity in a co-flow |
| D | Superscript referring to a discontinuous quantity |
| e | Index or superscript referring to a quantity in element Ω_e |
| $e - 1/2, e + 1/2$ | Indices referring to common values at left and right interfaces of element Ω_e |
| F | Index or superscript of a fuel quantity |
| f | Superscript referring to fresh gases |
| ff | Index or superscript referring to flame front |
| I | Superscript referring to an interface quantity |
| k | Index referring to a species $k \in \llbracket 1, N_s \rrbracket$ |
| L, R | Indices or superscripts referring to the left and right quantities at an interface |
| M | Index or superscript referring to a quantity on the Mortar element |
| n | Index referring to a quantity in a normal direction. Superscript referring to the considered time instant |
| O | Index or superscript of an oxidizer quantity |
| q | Index or superscript referring to the current stage of a RK method |

| | |
|------|---|
| st | Index of a quantity at a stoichiometric equivalence ratio |
| t | Superscript referring to target quantities at a NSCBC or to a turbulent quantity. Index also refers to a turbulent quantity |

Overline symbols

| | |
|-------------------|--|
| $\bar{\bullet}$ | Refer to a time-average or a resolved filtered quantity |
| $\hat{\bullet}$ | Refer to a quantity in the isoparametric domain |
| $\tilde{\bullet}$ | Refer to a reconstructed quantity using correction functions or to a Roe's average or to Favre time-average quantity or also to a resolved Favre-filtered quantity |

Roman Symbols

| | |
|---|--|
| β_j | Temperature exponent of reaction j in Arrhenius's law |
| \mathcal{D} | Diffusive terms in the normal direction of a NSCBC under Kim and Lee formulation |
| \mathcal{L} | Wave strengths of convective fluxes in the normal direction of a NSCBC under Kim and Lee formulation |
| \mathcal{N}, \mathcal{S} | Normal and tangential wave strengths of total fluxes (convective and diffusive) in the normal direction of a NSCBC under Fievet et al. formulation |
| $\mathcal{T}_c, \mathcal{T}_d$ | Convective and diffusive terms in tangential directions of a NSCBC under Kim and Lee formulation |
| $\mathbf{A}, \mathbf{b}, \mathbf{c}$ | The matrix and the two vectors in a Butcher tableau of a RK scheme |
| $\mathbf{P}^m(I)$ | Space of polynomial of degree m or less on I |
| $\Delta h_{f,k}^0$ | Mass enthalpy of formation of species k |
| ΔH_j^0 | Molar enthalpy change in reaction j |
| ΔS_j^0 | Molar entropy change in reaction j |
| $[X_k]$ | Molar concentration of species k |
| $ J $ | Determinant of the Jacobian matrix J |
| $\ \cdot\ _2$ | L^2 -norm operator |
| $\mathcal{A}_c, \mathcal{A}_d, \mathcal{A}$ | Convective, diffusive and total correction terms accounting for the mesh non-orthogonality at a NSCBC |
| $\mathbf{E}, \mathbf{F}, \mathbf{G}$ | Total fluxes (convective and diffusive) of \mathbf{U} respectively along x , y and z directions |
| $\mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z$ | Unit vectors of the Cartesian basis respectively $(1, 0, 0)^T$, $(0, 1, 0)^T$ and $(0, 0, 1)^T$ |
| \mathbf{F}_{trip} | Induced drag force used for artificial tripping |
| \mathbf{I} | Identity matrix |
| \mathbf{M}_k | Species flux of species k |
| \mathbf{n}, \mathbf{n}^u | Non-unit and unit normal vectors in the physical domain |
| \mathbf{Q} | Vector of primitive variables in the physical domain |
| \mathbf{q} | Energy flux vector |

| | |
|--|--|
| \mathbf{S} | Source term vector in the physical domain |
| \mathbf{S}_{trip} | Source term used for artificial tripping |
| $\mathbf{t}_1^u, \mathbf{t}_2^u$ | Unit tangential vectors to \mathbf{n}^u in the physical domain |
| \mathbf{U} | Vector of conservative variables in the physical domain |
| \mathbf{u} | Velocity vector |
| \mathbf{U}_h | Continuous solution vector of degree p in an element respectively in physical and isoparametric domains |
| \mathbf{V}^c | Correction velocity used under Hirschfelder and Curtiss approximation |
| \mathbf{V}_k | Diffusion velocity of species k |
| \mathbf{W} | Vector of characteristic variables in the physical domain |
| \mathcal{E} | Efficiency function |
| \mathcal{F} | Flux tensor in the physical domain $(\mathbf{E}, \mathbf{F}, \mathbf{G})^T$ or thickening factor |
| $\mathcal{F}_{c/d,n}$ | Convective/diffusive flux tensor projected on a normal direction in the physical domain |
| \mathcal{F}_{max} | Maximum thickening |
| \mathcal{G}_Δ | LES filter function of width Δ |
| \mathcal{H} | Standard hexahedral element $\{(\xi, \eta, \zeta), 0 \leq \xi, \eta, \zeta \leq 1\}$ |
| \mathcal{L} | Standard line element $\{\xi, 0 \leq \xi \leq 1\}$ |
| \mathcal{M} | Pope's criterion |
| $\mathcal{P}(X)$ | Probability density function of random variable X |
| \mathcal{Q} | Passage matrix for the rotation of an angle θ in the xy -plane |
| \mathcal{R} | Residual used in temporal integration to denote source terms and flux divergence discretization |
| $\mathcal{S}, \hat{\mathcal{S}}$ | Unfiltered and filtered relaxation flame sensors in the TFLES model |
| \bar{R} | Molar ideal gas constant set to $\bar{R} = 8.314 \text{ J.mol}^{-1}.\text{K}^{-1}$ |
| $\hat{\mathbf{E}}_h, \hat{\mathbf{F}}_h, \hat{\mathbf{G}}_h$ | Continuous flux polynomials of degrees $p+1$ in an element respectively along ξ, η and ζ flux points. |
| \tilde{C}_{pk} | Effective heat capacity at constant pressure of species k |
| A_{FP} | Area at a given flux point |
| A_j | Pre-exponential constant of reaction j in Arrhenius's law |
| C | Vortex strength |
| c | Progress variable or speed of sound |
| C_D | User-specified drag coefficient employed for artificial tripping |
| C_f | Skin-friction coefficient |
| C_p, C_{pk} | Mass specific heat capacities at constant pressure of the mixture and of species k |
| C_p^m, C_v^m | Molar specific heat capacity at constant pressure and volume of the mixture |
| C_v, C_{vk} | Mass specific heat capacities at constant volume of the mixture and of species k |

| | |
|-------------------------------------|---|
| $com(A)$ | Comatrix of matrix A |
| d | Space dimension |
| D_{bb} | Bluff-body diameter in the Cambridge burner configuration or edge length of the bluff-body equilateral triangle flame holder in the VOLVO configuration |
| D_k | Diffusion coefficient of species k into the rest of the mixture |
| D_{mk} | Binary mass diffusion coefficient of species m into species k |
| D_{th} | Thermal diffusivity |
| $diag(\mathbf{v})$ | Diagonal matrix with \mathbf{v} components on the diagonal and zero elsewhere |
| E | Total energy per unit mass of the mixture |
| e_s | Mass sensible energy of the mixture |
| Ea_j | Activation energy of reaction j in Arrhenius's law |
| g | Correction function used in SDLIFT formulation |
| g_{DG} | Correction function of degree $p + 1$ equal to the Right Radau polynomial of degree $p + 1$ |
| g_{Ga} | Correction function of degree $p+1$ which is zero at $\xi = 1$ and also at the p Gauss-Legendre points |
| g_e^L, g_e^R | Left and right correction functions |
| h, h_k | Mass enthalpies of the mixture and of species k . h can also be the channel half-height. |
| h_s, h_{sk} | Mass sensible enthalpies of the mixture and of species k |
| J | Jacobian matrix of the transformation from isoparametric to physical domain |
| K_{fj}, K_{rj} | Forward and backward reaction rate constants of reaction j |
| K_j^{eq} | Equilibrium constant of reaction j |
| $K_P, K_{\mathbf{u}}, K_T, K_{Y_k}$ | Relaxation rates for pressure, velocities, temperature and species mass fractions at NSCBC |
| K_{RES} | Resolved TKE |
| K_{SGS} | Subgrid-scale TKE |
| L | Length of channels or pipes |
| L_{ch}, R_{ch} | Length and radius of the chamber in the Cambridge burner computational domain. |
| L_e | Characteristic length of the most energetic structures in PP and VKP energy spectrums |
| L_f | Longitudinal integral length scale |
| L_f^{th} | Theoretical flame length in the 2D burner case |
| L_{inj} | Length of the injection tubes in the Cambridge burner computational domain. |
| l_i^{VT} | i -th 1D Lagrange polynomial built at vertex i |
| l_j^{SP} | j -th 1D Lagrange polynomial built at SP j |
| l_k^{FP} | k -th 1D Lagrange polynomial built at FP k |
| L_{ref}, L_c | Reference and cutting lengths for the injected turbulent spectrum at an inlet |
| L_{trip}, R_{trip} | Length and radius of the cylindrical artificial tripping |

| | |
|----------------------------|--|
| Le_k | Lewis number of species k |
| m, m_k | Masses of the mixture and of species k |
| M_i | i -th shape function |
| N | Number of turbulent Fourier modes |
| n, n_k | Number of moles of the mixture and of species k . n can also be a time instant. |
| N_{eq} | Number of conservative equations considered |
| N_e | Number of elements used to discretize Ω |
| $N_{FP}^{dir,d}$ | Number of flux points per direction in dimension d in a standard element |
| n_{FP}^I, n_{FP}^{tot} | Number of interface FP and total number of FP in the whole domain |
| n_{pts}^{ff} | Number of points in the flame front |
| N_r | Number of reactions |
| N_{SP}^{1d}, N_{FP}^{1d} | Number of solution points and flux points in a 1D line element |
| N_{SP}^d, N_{FP}^d | Number of solution points and flux points in dimension d in a standard element |
| N_s | Number of species |
| N_v | Number of vertices in element Ω_e |
| P | Static pressure of the mixture |
| p, p_e | Polynomial degree respectively in the general case and in an element Ω_e |
| P_U, P_Q | Transformation matrices respectively from conservative and primitive variables to characteristic variables |
| P_k | Partial pressure of species k or Legendre polynomial of degree $k \geq 0$ |
| p_{min}, p_{max} | Minimum and maximum polynomial degrees allowed for a p -adapted simulation |
| Pr | Prandtl number of the mixture |
| Q | Number of stages of a RK scheme |
| Q_{fj}, Q_{rj} | Forward and backward rates of progress of reaction j |
| Q_j | Rate of progress of reaction j |
| R, D | Pipe radius and diameter |
| r, θ, z | Physical coordinates in Cylindrical basis |
| R^{cart}, R^{cyl} | Velocity correlation tensor in Cartesian and cylindrical coordinates |
| R_c | Vortex radius |
| R_{gas} | Mass gas constant of the mixture \bar{R}/W |
| $R_{L,k}, R_{R,k}$ | Left and right Radau polynomials of degree $k \geq 1$ |
| S | Order of accuracy of a RK scheme |
| s | Mass stoichiometric ratio of a global reaction |
| S_* | Speed of the contact wave when using HLLC scheme |
| S_a, S_d, S_c | Absolute, displacement and consumption speed |

| | |
|-----------------------|---|
| s_k | Mass entropy of species k |
| S_L | Propagation speed |
| S_L, S_R | Speed of left and right acoustic waves when using HLLC scheme |
| S_L^0 | Unstretched laminar flame speed |
| Sc_k | Schmidt number of species k |
| T | Temperature |
| t | Time variable |
| t_{avg} | Averaging duration |
| T_{switch} | Temperature to switch from fresh to burnt gases when computing characteristic time τ_1 for the relaxation flame sensor in the TFLES model. |
| Ta_j | Activation temperature of reaction j in Arrhenius's law |
| $TV(U)$ | Total variation of variable U |
| u, v, w | Components of velocity vector in x, y and z directions |
| u'_Δ | Subgrid turbulent velocity up to scale Δ |
| u_τ | Friction velocity |
| u_θ^{max} | Maximum value of vortex velocity |
| U_b | Bulk velocity |
| U_{cl} | Centerline velocity |
| u_{rms} | Intensity of turbulence in each direction of the flow |
| u_r, u_θ, u_z | Velocity components of velocity vector in cylindrical coordinates |
| u_{tn} | Amplitude of the n -th Fourier mode associated to κ_n |
| V | Volume |
| W, W_k | Molar masses of the mixture and of species k |
| x, y, z | Physical coordinates in Cartesian basis |
| x^* | Space where thickened and non-thickened flames have the same structure |
| x^* | Non-dimensionalized axial coordinate in the VOLVO configuration $(x - x_{bb})/D_{bb}$ |
| x_{bb} | Axial location of the bluff-body equilateral triangle flame holder in the VOLVO configuration |
| x_c, y_c | Coordinates of vortex center in the xy -plane |
| x_i^e, y_i^e, z_i^e | Cartesian coordinates of vertex i in element Ω_e |
| X_k | Mole fraction of species k |
| Y_k | Mass fraction of species k |
| y_w | Wall normal coordinate of the closest grid point to the wall |
| Z | Mixture fraction |
| z | Normalized mixture fraction |

Nomenclature

| | |
|-----------------------|--|
| E | Mass kinetic energy density also called energy spectrum function |
| Fo | Fourier number |
| Re | Reynolds number |
| Re_b, Re_τ, Re_t | Bulk, friction and turbulent Reynolds numbers |

Introduction

Overview

| | |
|--|-----------|
| 1.1. Industrial context | 2 |
| 1.1.1. Environmental and economic context | 2 |
| 1.1.2. Technical context | 2 |
| 1.2. Large eddy simulations and high-order methods | 3 |
| 1.2.1. Numerical dissipation and dispersion | 3 |
| 1.2.2. Structured and unstructured meshes | 4 |
| 1.2.3. High-order schemes | 4 |
| 1.2.4. Towards new generation of high-order schemes | 5 |
| 1.3. High-order methods for combustion | 7 |
| 1.3.1. Specificities of numerical combustion | 7 |
| 1.3.2. Combustion simulations with DG discretization | 8 |
| 1.3.3. Combustion simulations with SD and FR discretizations | 8 |
| 1.3.4. Interest of discontinuous high-order methods for combustion | 9 |
| 1.4. A word on the lattice Boltzmann method | 9 |
| 1.5. Objectives of the thesis | 10 |
| 1.6. Outline of the manuscript | 10 |
| 1.7. Publications | 11 |
| 1.7.1. Non-peer-reviewed journals | 11 |
| 1.7.2. Peer-reviewed journals | 11 |
| 1.7.3. Peer-reviewed Conferences | 11 |

This thesis was conducted under a CIFRE project funded by Safran and the Association Nationale de la Recherche et de la Technologie (ANRT) with the research project number 2019/0794. The two laboratories involved in the project were the Centre Européen de Recherche et de Formation Avancées en Calcul Scientifique (CERFACS) and the Office National d'Etudes et de Recherches Aéronautiques (ONERA).

1.1. Industrial context

1.1.1. Environmental and economic context

The third and last part of the 6th assessment report on climate change, provided by the Intergovernmental Panel on Climate Change and published on April 4th 2022 gives direct measures to be taken in all sectors in order to limit global warming to 1,5 °C at the end of the 21st century [1]. In particular, the reduction of greenhouse gas emissions (mostly composed of CO₂) has to be conducted. The International Energy Agency (IEA) has released guidance to reach net zero CO₂ emissions globally by 2050, called the net zero pathway [2]. Among the multiple objectives announced, the end of thermal engine cars marketing by 2035 (approved on June 8th 2022 by European Parliament for Europe) and the use of 50% low-emissions fuels in aviation by 2040 are major challenges that the combustion sector must face. Consequently, alternative fuels such as hydrogen, ammonia, sustainable aviation fuels and synthetic fuels are currently considered to replace oil and kerosene in future cars and planes. The impact on the design of combustion chambers is significant, both in terms of security and efficiency. In particular, designing engines with lower emissions and higher energy efficiency, which drive the technology in opposite directions, is very challenging. That is why, improving the engineering tools to build the next generation of combustion chambers is extremely important and a key part of the net zero pathway proposed by the IEA.

1.1.2. Technical context

Combustion chambers are systems where very complex physical phenomena appear and interact, and are studied both experimentally and numerically. The advantage of experimental studies is to directly check the final product that will be used at the end of the design process to make sure everything works well. Nevertheless, it has two main drawbacks:

- Multiple prototypes will be built before a satisfactory final product is obtained, making experimental campaigns very costly.
- In combustion chambers, the complexity of the geometry and the high temperature make it very hard to obtain accurate measurements of the flow, flame and major species at various locations.

Thus, experiments are nowadays most often completed with numerical studies, where equations describing the considered physics are solved using discretization methods applied to a mesh (also called a grid). When the domain is fluid dynamics, inside which combustion is a particular case, it is called Computational Fluid Dynamics (CFD). The best benefit of CFD is that it is able to determine at all locations of the computational domain, any quantity of interest like temperature or composition, for any operating point. It is then very useful and cheaper than experiment to check some trends during the design phase. However, CFD has also two main kinds of limitations:

- **Intrinsic limits:** the equations solved are already models that are built to describe real flow behaviors.
- **Computational limits:** the resolution needed to perfectly solve these equations in industrial configurations, is not affordable with computational power currently available. For instance in aeronautical chambers at high pressure, the flame thickness is around 0.1 mm whereas the cell size is typically of the order of 1 mm. Using a mesh refined enough to capture the 0.1 mm flame

thickness is affordable only on academic tests cases with simple geometries. This implies to use additional models to describe the sub-grid scale phenomena.

The computational limits linked to the ratio of the physical scales to the grid cell size, are at the origin of the three main categories of simulation approach for turbulent flows. In Direct Numerical Simulations (DNS), the grid is fine enough to capture all turbulent scales and no additional model is needed. As already said, it is still out of reach for industrial configurations. As it will be shown in Chapter 3, when only the mean flow is computed the modeling concerns all the turbulent eddies. This approach is called Reynolds-Averaged Navier-Stokes (RANS) and is widely used in the aeronautical industry for its very low computational cost. However, RANS method does not give access to the intermittent part of turbulence. That is why, with the continuous improvement of high performance computing (HPC), Large Eddy Simulations (LES) are more and more employed for the design of combustion chambers. In LES, the large eddies of turbulence are resolved by the mesh and the small eddies are modeled, as they have a more universal behavior [3]. This approach is well suited for studying combustion phenomena since turbulent combustion is primarily linked to turbulent mixing processes at large scales.

This classification also holds for turbulent combustion modeling, where both the turbulent flow and the flame front must be modeled. In Chapter 3, classical turbulent combustion models will be introduced. They all have the same objective: define a way to model species and energy combustion source terms able to reproduce the turbulence-flame interaction that the mesh cannot tackle.

1.2. Large eddy simulations and high-order methods

1.2.1. Numerical dissipation and dispersion

LES is meant to explicitly describe the large eddies of the flow. To be useful, it needs to be accurate and therefore associated with discretization methods with minimum numerical error. By construction, numerical methods intrinsically create dissipation and dispersion due to the truncation of the Taylor-series expansions of derivatives [4]:

- The dissipation of a numerical scheme is linked to the even derivative terms in the truncation error (TE) and tends to smooth all gradients of the solution. It is also called implicit artificial viscosity as it acts as diffusion, and as opposed to explicit artificial viscosity which is purposely added by the user to smooth out non-physical perturbations.
- The dispersion of a numerical scheme is linked to the odd derivative terms in the TE and modifies the propagating velocity of perturbations, leading to the distortion of the solution.

These two errors increase when the numerical scheme order decreases. Mathematically speaking, a numerical method is said to have an order of accuracy k if the solution error e is proportional to the grid size h to the power of k :

$$e \propto h^k \quad (1.1)$$

According to a survey sent in 2007 to the members of the technical committee of the CFD Algorithm Discussion Group and other researchers, high-order (HO) methods refer to third-order methods and higher [5]. Because the TE terms become more and more smaller when the scheme order increases, dissipation and dispersion errors are low for HO methods.

Dissipation and dispersion are strong issues for LES, since they can deform and dissipate large eddies that should be captured by the mesh. They have a huge impact for vortex-dominated flows, like the flow around a helicopter propeller (accurate resolution of unsteady vortices), or for aeroacoustics where broadband acoustic waves propagate over long distances [5].

1.2.2. Structured and unstructured meshes

The concepts of structured and unstructured meshes are important to fully understand the chronology of numerical methods and the growing use of HO approaches. A mesh is a subdivision of the considered domain into small volumes called elements or cells. Each of these elements is defined by its nodes (also called vertices), edges and faces for the 3D elements.

In the case of structured meshes, the domain is discretized using either quadrilaterals (in 2D) or hexahedra (in 3D) elements where each node is associated with a unique (i, j) doublet in 2D or (i, j, k) triplet in 3D. Therefore, neighbors of a given element or node are easily retrieved. However, the generation of structured meshes for complex geometries can be very long which limits their use to simple configurations only.

In the case of unstructured meshes, the domain can be discretized with elements of other shapes than only quadrilaterals or hexahedra: triangles and tetrahedra, prisms and pyramids, or even more complex shapes. Each mesh element is defined by its list of nodes numbered in the global list for the whole mesh. This element-nodes association is called the connectivity table since it also allows to know which node is shared by which elements, and therefore which elements are direct neighbors in the mesh. Node coordinates are also stored in a coordinate table. Thus, in comparison with structured meshes, it is more complicated to find the coordinates of a given element since this requires to find the corresponding nodes in the connectivity table and the node coordinates in the coordinate table. The advantage of unstructured meshes however is a much faster generation for complex geometries. Moreover, the use of multiple types of elements makes them more flexible to mesh any kind of geometry. That is why, unstructured meshes are commonly employed for industrial configurations, and any numerical method considered for industrial application should be able to deal with unstructured meshes.

1.2.3. High-order schemes

The first numerical methods for CFD, which are still widely used are the finite difference (FD) and the finite volume (FV) approaches. Both may be employed with HO schemes. There are also the finite element (FE) method and spectral-like methods but they are less popular in CFD compare to FD and FV probably due to their more complex formalism. Then, it was chosen to not detail them in this paragraph.

1.2.3.1. The finite difference method

In the FD formulation, the computational domain is discretized using structured meshes and the solution is computed at the mesh nodes. The derivatives are also computed at nodes with truncated Taylor-series expansions using the solution values at neighboring nodes. The number of neighbors is called the stencil of the scheme and the higher the stencil the higher the scheme order. Therefore, HO FD schemes are easily obtained by increasing the stencil. To gain in efficiency, compact FD schemes have been designed to reach high order without very large stencils [6]. Still, HO FD methods suffer from their large stencil in parallel computing as they imply an increase of data communication between processors. In addition, FD methods require structured meshes which, as mentioned in paragraph 1.2.2, are not well adapted to complex geometries unless non-body fitted conditions are used like for instance in the CONVERGE CFD software [7].

1.2.3.2. The finite volume method

The FV formulation is based on the mesh elements, considered as control volumes over which the equations to solve are integrated, ending up with flux balances to update the solution in the cell. As the control volume can have any shape, FV methods are more adapted to unstructured meshes. The solution is stored either at the cell center or at the cell nodes. Most industrial codes

use second-order FV methods [5] but higher-order, up to fifth-order FV methods also exist, such as essentially non-oscillatory (ENO) [8–10], weighted ENO (WENO) [11] or also k -exact [12, 13] schemes. Recently, a so-called targeted ENO (TENEO) [14] scheme has been developed to improve the resolving of small turbulent eddies by ENO-like methods. However, all these HO FV schemes remain complex to implement and still employ extensive stencils which makes it difficult to keep parallel efficiency.

1.2.4. Towards new generation of high-order schemes

As a conclusion of the above paragraph 1.2.3, HO methods adapted to unstructured grids and parallel efficiency must avoid the use of large stencils. This has motivated the recent development of HO discontinuous methods, which all follow the same principle: a HO representation of the solution, usually a polynomial of degree p , is computed in each mesh element and coupled with element interface treatments to ensure conservativity. The scheme order is directly linked to the value of p which can be easily changed in space and time during a simulation. Efficient parallelization is achieved because solving the equations only involves direct neighbors through interface treatments whatever the value of p is in this element. With such method, increasing the spatial resolution may be achieved either by increasing locally the polynomial degree p , called p -refinement (or p -adaptation), or by doing local mesh refinement, called h -refinement. Typically, p -refinement is used to locally reduce dissipation and dispersion errors in regions where complex physics occur, whereas h -refinement stays necessary in regions with geometrical and physical discontinuities [15, 16].

1.2.4.1. The Discontinuous Galerkin method

The most famous HO discontinuous approach is the Discontinuous Galerkin (DG) method developed by Reed and Hill in 1973 for the neutron transport equation [17]. Almost twenty years later, Cockburn and Shu applied the DG approach to conservation laws and more specifically to the Navier-Stokes equations (NSE) [18–20]. Since then, DG has been widely employed to perform LES of various problems such as turbulent jets [21], laminar to turbulent transition [22], shock waves [23] and combustion [24–28]. The DG method combines the principles of the finite elements (FE) method and of the FV method. As in FE, the solution is approximated by a linear combination of basis functions and the equations are integrated within each element in a weak formulation. Fluxes at element interfaces are treated as in FV, using approximate Riemann solvers to resolve the flux discontinuity introduced by the basis functions. There are actually three main categories of DG approaches depending on the chosen set of basis functions:

- the modal DG approach is the original one used by Cockburn and Shu [18–20] where the orthogonal basis functions are Legendre or Lobatto polynomials [15]. This approach has shown super-convergence properties on scalar conservation laws [29] meaning that for a polynomial of degree p , the scheme order is strictly higher than $p + 1$. Nevertheless, as the scheme order increases, the computational cost soars fastly. This is due to the fact that flux integrals appearing in the weak form must be evaluated exactly using quadrature rules (based on the orthogonal basis functions chosen) more and more costly as the scheme order increases [30]. To reduce this cost, non-exact quadrature rules are usually employed but this can create huge aliasing errors, leading to the blow up of the calculation, especially when flows are highly turbulent. Moreover, the timestep scales as $(2p + 1)^{-1}$ which becomes very restrictive when p goes up.
- the nodal DG approach was introduced by Hesthaven and Warburton [31], where the basis functions are Lagrange polynomials, reducing the cost of the quadrature rules compared to modal DG. However, super-accuracy is lost. This is actually not critical since this property was verified for scalar conservation laws only, whereas CFD equations are more complex. However, and as for modal DG, the timestep also scales as $(2p + 1)^{-1}$.

- the DG spectral element method (DGSEM) presented by Kopriva and Gassner [32] also uses Lagrange polynomials but these are built at the same quadrature points than those used to evaluate flux integrals. This allows to relax the timestep limitation, which scales now as $(p + 1)^{-1}$. However, the use of DGSEM on hybrid meshes composed of quadrilaterals and triangles in 2D or hexahedra and tetrahedrals in 3D is more complicated than for modal or nodal DG [33].

1.2.4.2. The Spectral Volume method

The Spectral Volume (SV) method was introduced by Z.J. Wang [34–36] for 2D conservation laws on unstructured meshes, was later extended to 3D by Liu [37] and to viscous flows by Sun [38]. It consists in subdividing each mesh element into small control volumes called sub-volumes inside which a classical FV technique is applied. The number of sub-volumes is linked to the targeted polynomial degree. As for the DG method, the SV approach uses Riemann solvers at element boundaries only since the solution is built continuous at sub-volumes boundaries. Additionally, the flux evaluations also require HO quadrature rules to compute surface integrals making the method very costly especially for 3D flows. Harris *et al.* [39, 40] proposed to overcome this issue with a SV method which does not need quadrature rules. Another issue is that for orders of accuracy higher than two, the division of a spectral volume into sub-volumes is not unique and has a considerable influence on the stability of the method, as studied by Van den Abeele *et al.* [41]. Mainly for this reason, the SV method has lost interest during the last decades compared to other HO methods.

1.2.4.3. The Spectral Difference method

In paragraphs 1.2.4.1 and 1.2.4.2, the presented HO methods are based on a weak integral form of the equations and require costly quadrature rules for flux computations. Therefore, new HO discontinuous methods built to solve directly the strong form of the equations, meaning without integrating them, have emerged. They are usually called strong HO discontinuous methods as opposed to DG and SV-like methods often named as weak HO discontinuous methods. One of these strong discontinuous methods is the Spectral Difference (SD) approach originally introduced by Kopriva and Kolias in 1996 [42, 43] as the staggered-grid Chebyshev multidomain method, applied to structured quadrilateral elements using a tensor-product formulation. In 2006, Liu *et al.* [44] extended the method to triangular elements and conservation laws. Then, Wang *et al.* [45] used it for the Euler equations and the extension to Navier-Stokes equations (NSE) was finally done by May and Jameson [46] for triangular grids and by Sun *et al.* [47] for hexahedral elements.

For tensor-product cells such as quadrilaterals or hexahedra, the standard SD method builds a polynomial of degree p to express the solution using values at what are called solution points (SP) and a polynomial of degree $p + 1$ to express fluxes using values at another set of points called flux points (FP). This gives a scheme order of $p + 1$ [42]. As for DG and SV methods, interface treatments such as Riemann solvers are employed to ensure conservation across elements. Recently, Chen *et al.* [48] presented a new formulation for this kind of elements where the flux divergence is built from the flux values at SP completed by flux values at interface FP, avoiding to interpolate from SP to internal FP. They called this approach the collocated-grid SD method. Stability of the standard method on tensor-product elements was investigated by Van den Abeele *et al.* [49] and Jameson [50] who concluded that SP positions have no influence on the stability whereas FP positions do have one. In particular, the SD method on tensor-product cells is stable for all orders of accuracy if the interior FP are placed at the roots of the corresponding Legendre polynomial of degree p . For triangular and tetrahedral elements, also called simplex cells, the stability also depends on FP but the set of stable FP is much difficult to find. The use of Raviart-Thomas (RT) elements on triangles in SD, named as the SDRT method, was proven to be linearly stable up to the 4th order by May *et al.* [51]. It seems to be the more promising approach and has been applied recently on various test cases [52–54]. Moreover, Veilleux *et*

al. [55] extended the SDRT method on triangle elements up to 6th order and on tetrahedral elements up to 3rd order [56].

1.2.4.4. The Flux Reconstruction method

Another class of strong HO discontinuous method is the Flux Reconstruction (FR) technique, also called the Correction Procedure for Reconstruction, developed by Huynh [57] in 2007. In this method the solution and the flux polynomials are defined on the same set of points: SP and FP are collocated. It implies that the flux divergence is no longer a polynomial of degree p and also that the scheme is not conservative because the flux is discontinuous at element interfaces. Flux correction functions, also named lifting operators and taken as polynomials of degree $p + 1$, are used to tackle these issues and make the FR method conservative [57]. These functions have a huge impact on the characteristics of the FR scheme employed. In particular, specific flux corrections functions can link the FR method with DG or SD methods [57, 58]. It is also possible to derive mathematical expressions for the lifting operators that guarantee an energy stable scheme [59]. Jameson *et al.* studied the stability of the method in both linear [60] and nonlinear [61] cases and the dissipation and dispersion properties of the FR approach were investigated in [62, 63]. Extensions to triangular [64] and tetrahedral [65] elements for any order of accuracy have been considered. Consequently, the FR method was very popular for multi-element meshes compared to the SD method where for instance tetrahedral elements are not stable for $p > 3$ as stated in paragraph 1.2.4.3. That is why, the FR technique has been applied to multiple flow configurations over the past ten years [66–69]. It should be mentioned that recently Romero *et al.* [70] proposed the direct FR method where the existing FR implementation of the nodal DG scheme is recovered with a much simpler correction procedure than the original one proposed by Huynh [57]. It was then extended to triangular meshes [71] making this approach a valuable candidate to be used in the future of FR techniques.

1.3. High-order methods for combustion

In Section 1.2, it was shown that LES needs low-dissipation and low-dispersion errors to perform well, promoting the use of HO numerical schemes. Discontinuous HO methods seem to be excellent candidates to handle complex geometries with unstructured meshes while maintaining high orders of accuracy with efficient parallelization. Yet, although now widely employed in pure aerodynamic cases, HO methods have only been used in a few combustion cases, whether with DG, SD or FR methods.

1.3.1. Specificities of numerical combustion

In pure aerodynamics, the mono-species non-reacting gas is usually considered as air with a constant heat capacity ratio γ . In combustion, the gas is composed of N_s species and evolves in a flow where the temperature T varies importantly due to the flame. In that case, the conservation equations for a multi-species reacting gas are employed with a varying $\gamma = \gamma(T, Y_k)$, Y_k being the mass fraction of species $k \in \llbracket 1, N_s \rrbracket$. Thus, numerics for pure aerodynamics and combustion simulations differ in three main ways:

1. the multi-species mixture brings additional numerical constraints: $\sum_{k=1}^{N_s} Y_k = 1$ and $0 \leq Y_k \leq 1$ must be ensured at each grid point, which can lead to additional treatments [72].
2. the computation of combustion source terms introduces stiffness when fast intermediate species are considered in the flow, which can entail the use of dedicated time-integration procedures [28, 73, 74].

3. the temperature and species dependence of γ creates strong and spurious oscillations at interfaces due to different γ values when a conservative form of the multi-species equations is solved, independently of the numerical method [75]. More precisely in case of contact discontinuity, i.e., constant pressure and velocity with varying density (and so a temperature), it was shown in [75] that the first iteration of any FV numerical scheme preserves the constant velocity but not the constant pressure if $\gamma = \gamma(T, Y_k)$ (multi-species case) varies. This is due to the use of the total energy ρE conservation equation, often preferred in CFD since ρE is a conserved quantity. Working with the pressure equation solves the problem, but is no more conservative. Once pressure has gone wrong, the velocity also goes wrong at the next step and oscillations appear, which can be amplified in the presence of combustion source terms. One popular method to avoid this behavior in HO solvers is the double flux technique initially developed in the FV context [76, 77] but it is again intrinsically non-conservative. Other methods are based on the transport of additional quantities [75, 78–84], which increases the computational cost.

1.3.2. Combustion simulations with DG discretization

Combustion simulations with discontinuous HO methods are still little considered since it requires to adapt the numerical scheme to the multi-species reacting flow as highlighted in paragraph 1.3.1. Because the DG approach is the most mature discontinuous technique, well established for pure aerodynamics, it is also the most used for combustion cases. A review is given below in chronological order:

- Billet *et al.* [85] in 2011 were the first ones to apply the double flux methodology with the DG method to solve the Euler equations for a multi-species non-reacting gas.
- Later, Billet *et al.* [24] presented in 2014 a DG algorithm taking into account viscous and source terms in the multi-species reacting flow equations.
- Lv *et al.* [25], also in 2014, employed the double flux approach combined with a fully conservative upwind-biased flux formulation for the simulation of the multi-species and reacting flow equations. They validated the method on 1D and 2D multi-species advection cases and shock problems such as detonations showing the robustness of their algorithm.
- Lv *et al.* [26] in 2017 improved their methodology by employing an entropy-residual shock indicator coupled with an artificial viscosity approach. They validated it on a bluff-body stabilized turbulent premixed flame and on more complex detonation cases.
- Johnson *et al.* [27] recently in 2020 developed a DG approach to solve multi-species reacting flow problems without using the double-flux methodology. They also presented results with local p -refinement in a combustion context which is, to the best of the authors' knowledge, still unique in the DG community.
- Du *et al.* [28] very recently in 2022 published a methodology for preserving $0 \leq Y_k \leq 1$ in the DG framework. In particular, they employed a conservative modified exponential Runge-Kutta (CMERK) method [86] for time integration instead of Strong Stability Preserving (SSP) Runge-Kutta schemes [87] classically considered in most works on DG for combustion. Validation was done on 1D and 2D detonation test cases.

1.3.3. Combustion simulations with SD and FR discretizations

At the time of start of this work in 2019, the only published work about the use of SD and FR approaches for combustion was from Gupta *et al.* [88], who applied SD to 1D detonations in a multi-species reacting gas. However, nothing was specified about a specific treatment of discontinuities with varying γ , as mentioned above. Later on in 2021, Tofaili *et al.* [89] also simulated 1D detonations solving the multi-species reacting Euler equations with the SD method combined with a recently

developed HO shock capturing approach based on characteristic waves sensors [90] and an extended limiting procedure originally introduced for pure aerodynamics [91]. These papers have demonstrated the capability of the SD method to simulate 1D detonations whereas in the present work it is proposed to reproduce 1D laminar flames to 3D turbulent flames using the SD technique. The very recent work of Peyvan *et al.* [92] adapting the double flux method to the FR discretization of the multi-species Euler equations has also to be underlined.

1.3.4. Interest of discontinuous high-order methods for combustion

As it can be seen, the use of discontinuous HO methods in the combustion field is very recent but the methodology is very appealing especially due to p -refinement that locally changes the order of accuracy of the spatial scheme without modifying the mesh. In the current state of the art, spatial refinement is performed with Adaptive Mesh Refinement (AMR) techniques [93] coupled with FV discretization. In the AMR framework, a physically-based sensor is used to locate the regions of the mesh that should be refined but the order of the spatial scheme remains the same. The p -refinement has been applied in pure aerodynamics simulations either with the DG [94, 95] or the SD [96] method. Results showed a reduction by almost 50% of the number of mesh points with p -refinement, leading to a significant reduction of the computational time between 25% to 50% [96]. Thinking of flames, which are very local phenomena requiring high numerical accuracy, the potential of p -refinement is very attractive. Nonetheless, the only published paper showing local p -refinement in the context of discontinuous HO methods applied to combustion is the very recent work of Johnson and Kercher (2020) [27], already mentioned in paragraph 1.3.2.

1.4. A word on the lattice Boltzmann method

The review of the promising methods for LES of combustion would be incomplete without the lattice Boltzmann method (LBM). LBM was originally introduced by McNamara and Zanetti in 1988 [97]. It relies on the resolution of the lattice Boltzmann equations, which describe the flow at the mesoscopic scale, while the macroscopic quantities such as density, velocity or temperature are recovered using a quadrature rule applied to the solution of these lattice Boltzmann equations [98, 99]. The method has reached an excellent level of maturity for non-reacting flows, with the simulation of full scale applications in the last decade: full scale cars [100], full scale engines [101] and full scale aircrafts [102]. It compares well with classical FV solvers in terms of accuracy but has less computational cost [103]. LBM has three main advantages:

1. The mesh generation even of complex geometries is extremely fast since LBM uses non-body fitted Cartesian meshes [104].
2. The parallelization of LBM is easy and efficient [105].
3. LBM has low-dissipative properties [106] which is key for LES.

LBM however suffers from its low-Mach number formulation, although some extensions have been recently proposed with promising results [107, 108]. Very recently, LBM has been coupled to FD solvers for energy and species transport equations to tackle laminar [109, 110] and turbulent [111] reacting flows. Results show that this approach is a valuable candidate, as HO discontinuous methods, for the next generation of LES combustion codes. Nevertheless, one drawback of coupling LBM with a FD solver is that it no longer has the low-dissipative properties of LBM for energy and species transport equations.

1.5. Objectives of the thesis

Despite their increasing popularity for LES in both academic and industrial sectors, HO methods still suffer from a lack of robustness and reliability compared to current low-order (LO) methods. In particular, discontinuous HO methods which were originally introduced for simpler equations than flow equations, require additional treatments to be used in CFD. Some studies have tried to compare DG, FR and SD techniques but there is no final conclusion about which HO method should be used for a given application. Actually, in Liang *et al.* [112], it was shown that FR is most efficient and DG is the slowest one but in terms of accuracy DG is the best and FR is the worst. For both efficiency and accuracy the SD method was found to stand in between. In a more recent study, Cox *et al.* [113] stated that the SD method is more robust and accurate than FR. This is very encouraging for pursuing the development of the SD method to simulate complex flows which is the purpose of this work.

In this context, CERFACS and ONERA have joined forces to develop the code JAGUAR (proJect of an Aerodynamic solver using General Unstructured grids And high ordeR schemes) based on the SD method. JAGUAR was first applied to non-reacting, mono-species compressible flows showing interesting and promising performances [96, 114, 115]. JAGUAR was then extended to simplex and hybrid grids [55, 56, 116] and with p -refinement [96]. Finally, characteristic boundary conditions specially adapted to the SD framework [117, 118] were implemented in JAGUAR to cope with acoustics.

The first objective of the present work is to find, implement and validate a stable SD algorithm able to simulate the combustion of a multi-species mixture. It implies the extension to multi-species gas, adaptation of the characteristic boundary conditions and also of the usual walls and symmetry boundary conditions, implementation of the thermochemistry and of turbulent combustion model. Then, a second objective will be to demonstrate the performances of the SD method for combustion both in terms of accuracy and computational cost. In particular the use of local p -adaptation will be key for that.

1.6. Outline of the manuscript

To achieve the above objectives, the manuscript is organized in three main parts:

- **Part I** deals with the theoretical background of laminar and turbulent combustion:
 - ★ **Chapter 2** introduces basic definitions for describing a mixture of gases, the computation of chemical source terms and finally the governing equations of a multi-species reacting gas mixture employed throughout this thesis.
 - ★ **Chapter 3** briefly explains the concept of turbulent flows and how they can be solved using numerical simulations, with a focus on LES used in this work. Turbulent combustion modeling is also introduced, and The Thickened Flame model used in this work is presented.
- **Part II** is dedicated to the SD method, presented in the context of a mono-species and non-reacting gas, except for the characteristic boundary conditions which are presented for the general case:
 - ★ **Chapter 4** presents the SD method in detail. Starting from the one-dimensional hyperbolic or parabolic equations, the derivation of the three-dimensional discretization on hexahedral elements is made. The concept of Riemann solvers and the diffusion schemes are also introduced, along with the local polynomial adaptation technique. Finally, the time-discretization is exposed.
 - ★ **Chapter 5** addresses the subject of boundary conditions in the SD context for a multi-species gas mixture. Navier-Stokes characteristic boundary conditions (NSCBC) adapted to the SD formalism are firstly explained. Then, the treatment of symmetry and wall boundary conditions, still in SD framework, is tackled.

- ★ **Chapter 6** makes a first validation step for non-reacting flows, with the simulation of turbulent channel and pipe flows with the SD method. These test cases require turbulence injection through the inlet boundary condition.
- **Part III** finally brings up the application of the SD method to combustion:
 - ★ **Chapter 7** first demonstrates the validity of the implementation on one-dimensional and two-dimensional premixed laminar flames using different chemistries. This requires first to tackle the issue of the SD instability when considering a multi-species mixture, as it can occur according to the analysis made in paragraph 1.3.1. A first evaluation of the performance of the SD method is made via the impact of the spatial resolution of the flame front.
 - ★ **Chapter 8** is the final step of this work, applying SD to 3D turbulent premixed combustion cases. The implementation of the Thickened Flame model in the SD context is firstly presented. Two different turbulent combustion cases are then computed: the Cambridge burner [119, 120] and the VOLVO configuration [121, 122]. In both cases, the influence of the polynomial degree within each element is studied and results are compared with experimental and numerical data already available in the literature.

Finally, overall conclusions and perspectives for future work are drawn.

1.7. Publications

1.7.1. Non-peer-reviewed journals

- T. Marchal, H. Deniau, J-F. Boussuge, B. Cuenot, R. Mercier, *Extension of the Spectral Difference method to combustion*, <https://arxiv.org/pdf/2112.09636.pdf>.

1.7.2. Peer-reviewed journals

- T. Marchal, H. Deniau, J-F. Boussuge, B. Cuenot, R. Mercier, *Extension of the Spectral Difference method to laminar and turbulent combustion*, Submitted to Flow, Turbulence and Combustion in November 2022.

1.7.3. Peer-reviewed Conferences

- T. Marchal, A. De Brauer, H. Deniau, J-F. Boussuge, B. Cuenot, R. Mercier, *Efficiency of the high-order Spectral Difference method in combustion*, 39th International Symposium on Combustion. July 24-29th, 2022.

Part I.

Theoretical concepts and modeling of laminar and turbulent premixed combustion

Laminar premixed combustion

Overview

| | |
|--|-----------|
| 2.1. General definitions | 16 |
| 2.1.1. Chemical species and mixture composition | 16 |
| 2.1.2. Mixing rules | 17 |
| 2.1.3. Equivalence ratio and mixture fraction | 18 |
| 2.1.4. Progress variable | 19 |
| 2.1.5. Laminar flame speed | 19 |
| 2.1.6. Laminar flame thickness | 20 |
| 2.2. Chemical kinetics | 21 |
| 2.2.1. Rate of progress of a chemical reaction | 21 |
| 2.2.2. Species source term | 22 |
| 2.2.3. Heat release rate | 22 |
| 2.3. Choice of the chemical description | 23 |
| 2.3.1. Detailed chemistry | 23 |
| 2.3.2. Globally reduced chemistry | 23 |
| 2.3.3. Analytically reduced chemistry | 25 |
| 2.4. Conservation equations for multi-species reacting flows | 26 |
| 2.4.1. Conservation of mass of the mixture | 26 |
| 2.4.2. Conservation of momentum | 26 |
| 2.4.3. Conservation of mass for species | 26 |
| 2.4.4. Conservation of energy | 29 |
| 2.4.5. Models for diffusion coefficients | 30 |
| 2.4.6. Computing pressure and temperature from transported variables | 32 |
| 2.4.7. Summary of equations | 32 |
| 2.5. Summary of this chapter | 33 |

This chapter is dedicated to the definition of basic combustion concepts that will be used throughout the manuscript. Only premixed combustion is considered in this work. General definitions to characterize the species, the mixture and the flame are firstly introduced. The ways to model and compute combustion source terms is then assessed. Finally, the governing equations that are numerically solved in this work are summed up.

2.1. General definitions

The combustion phenomenon involves multi-species gases, also called mixtures, reacting through multiple chemical reactions which produce heat. Consequently, additional quantities to characterize the mixture and the species composing it are introduced..

2.1.1. Chemical species and mixture composition

For each species k of a given mixture, the following quantities can be defined:

- The **mass fraction** Y_k is the ratio between the mass m_k of species k and the total mass m of the mixture:

$$Y_k = \frac{m_k}{m} \quad (2.1)$$

The mass fractions must satisfy:

$$\sum_{k=1}^{N_s} Y_k = 1 \quad (2.2)$$

where N_s is the number of species that compose the mixture.

- The **mole fraction** is the ratio between the number of moles n_k of species k and the total number of moles n of the mixture:

$$X_k = \frac{n_k}{n} \quad (2.3)$$

Since, $n_k = m_k/W_k$, where W_k is the **molar mass**, and $n = m/W$, where W is the **mixture mean molar mass** (see Eq. (2.9) and Eq. (2.10)), the link between X_k and Y_k is given by:

$$X_k W_k = Y_k W \quad (2.4)$$

- The **molar concentration** $[X_k]$ $[\text{mol.m}^{-3}]$ is the number of moles n_k of species k per unit volume V :

$$[X_k] = \frac{n_k}{V} = \frac{n}{V} X_k = \frac{\rho X_k}{W} = \frac{\rho Y_k}{W_k} \quad (2.5)$$

where ρ is the density of the mixture.

- The **density** ρ_k $[\text{kg.m}^{-3}]$ of species k :

$$\rho_k = \rho Y_k \quad (2.6)$$

- The **mass specific heat capacity** at constant pressure $C_{pk}(T)$ $[\text{J.kg}^{-1}.\text{K}^{-1}]$ and at constant volume $C_{vk}(T)$ $[\text{J.kg}^{-1}.\text{K}^{-1}]$. Both quantities depend on temperature and the way they are obtained in this work is detailed in paragraph 2.4.6.
- The **mass enthalpy** $h_k(T)$ $[\text{J.kg}^{-1}]$ is the sum of the chemical and sensible enthalpies:

$$h_k(T) = \Delta h_{f,k}^0 + h_{sk}(T) = \Delta h_{f,k}^0 + \int_{T_0}^T C_{pk}(T') dT' \quad (2.7)$$

where $\Delta h_{f,k}^0$ is the mass enthalpy of formation of species k at a reference temperature T_0 and h_{sk} is the mass sensible enthalpy of species k at temperature T . The formation enthalpy $\Delta h_{f,k}^0$ is a constant defined for each species k .

Note that any **molar quantity** f_k^m , is linked to the associated mass quantity f_k with:

$$f_k^m = f_k \times W_k \quad (2.8)$$

2.1.2. Mixing rules

Mixing rules define the mixture properties from those of the species present inside the mixture. The molar mass of the mixture is computed using the mass fractions and the molar mass of each species:

$$\frac{1}{W} = \sum_{k=1}^{N_s} \frac{Y_k}{W_k} \quad (2.9)$$

or equivalently using the mole fractions:

$$W = \sum_{k=1}^{N_s} X_k W_k \quad (2.10)$$

Similarly, the mass specific heat capacity at constant pressure and at constant volume of the mixture is computed using the mass fractions:

$$C_p = \sum_{k=1}^{N_s} Y_k C_{pk} \quad \text{and} \quad C_v = \sum_{k=1}^{N_s} Y_k C_{vk} \quad (2.11)$$

To compute molar specific heat capacities, Eq. (2.11) is multiplied by the molar mass of the mixture:

$$C_p^m = C_p \times W \quad \text{and} \quad C_v^m = C_v \times W \quad (2.12)$$

The mass enthalpy of the mixture at temperature T is also defined:

$$h(T) = \sum_{k=1}^{N_s} Y_k h_k(T) = \sum_{k=1}^{N_s} Y_k \left(\Delta h_{f,k}^0 + \int_{T_0}^T C_{pk}(T') dT' \right) = \sum_{k=1}^{N_s} Y_k \Delta h_{f,k}^0 + h_s(T) \quad (2.13)$$

where $h_s(T)$ is the mass sensible enthalpy of the mixture at temperature T given by:

$$h_s(T) = \sum_{k=1}^{N_s} Y_k h_{sk} = \int_{T_0}^T C_p(T') dT' \quad (2.14)$$

Finally, if the mixture is composed of N_s thermally perfect gases, its static pressure is given by the sum of all species partial pressures (Dalton's law), noted P_k [Pa], which follow the perfect gas law:

$$P = \sum_{k=1}^{N_s} P_k \quad \text{with} \quad P_k = \rho_k \frac{\bar{R}}{W_k} T \quad (2.15)$$

where $\bar{R} = 8.314 \text{ J.mol}^{-1}.\text{K}^{-1}$ is the molar ideal gas constant. Consequently, the mixture also follows the perfect gas law:

$$P = \sum_{k=1}^{N_s} P_k = \sum_{k=1}^{N_s} \rho_k \frac{\bar{R}}{W_k} T = \rho \bar{R} T \sum_{k=1}^{N_s} \frac{Y_k}{W_k} = \rho \frac{\bar{R}}{W} T \quad (2.16)$$

The quantity \bar{R}/W is named the mass gas constant of the mixture noted R_{gas} and Eq. (2.16) is called the equation of state (EOS) of the mixture. Since the perfect gas law is employed, R_{gas} can also be computed using Mayer's relation:

$$R_{gas} = C_p - C_v \equiv R_{gas} \quad (2.17)$$

2.1.3. Equivalence ratio and mixture fraction

Combustion involves reactants and products, and strongly depends on the ratio of the fuel and oxidizer concentrations. Definitions are given below for premixed flames only, where the fuel and oxidizer are fully premixed prior to react in the flame front.

2.1.3.1. Global equivalence ratio

The fuel-oxidizer ratio is usually measured with the global equivalence ratio of the mixture, which compares the fuel-oxidizer ratio in the mixture to the stoichiometric value. Consider a global one-step chemical reaction of the type [3]:



with ν'_F and ν'_O respectively the stoichiometric coefficients of fuel F and oxidizer O , determined from the atomic balance between reactants and products. The mass stoichiometric ratio s of the global reaction in Eq. (2.18) is the inverse mass ratio corresponding to stoichiometric proportions (fuel and oxidizer completely converted into products):

$$s = \left(\frac{Y_F}{Y_O} \right)^{-1}_{st} = \frac{\nu'_O W_O}{\nu'_F W_F} \quad (2.19)$$

where W_F and W_O are respectively the molar mass of the fuel and oxidizer. Using Eq. (2.19), a global equivalence ratio is introduced for any mixture:

$$\phi_g = s \frac{Y_F}{Y_O} = \frac{\left(\frac{Y_F}{Y_O} \right)}{\left(\frac{Y_F}{Y_O} \right)_{st}} \quad (2.20)$$

with Y_F and Y_O respectively the fuel and oxidizer mass fractions in the mixture. Depending on the values of ϕ_g , three different regimes are introduced:

- $\phi_g < 1$: the lean regime where the oxidizer is in excess and the fuel is completely turned into products.
- $\phi_g = 1$: the stoichiometric regime where both the oxidizer and the fuel are completely turned into products.
- $\phi_g > 1$: the rich regime where the fuel is in excess and the oxidizer is completely turned into products.

2.1.3.2. Mixture fraction and local equivalence ratio

In the case of premixed gaseous flames, and in the absence of dilution or secondary injection, the equivalence ratio is constant throughout the flame, i.e., even in the burnt gases, and is equal to the global equivalence ratio [3]. However, in practical systems with multiple injections, dilution or liquid fuel droplets, the local equivalence ratio ϕ may vary significantly. This local equivalence ratio is often re-written in the form of a mixture fraction, noted z here, which evaluates the local mixing between reactants in the combustion chamber, whatever the way they have been introduced. A very convenient definition, because it remains valid in the burnt gases, was introduced by Bilger [123]:

$$Z = \sum_{k=1}^{N_s} \frac{\xi_k}{W_k} Y_k \quad (2.21)$$

with ξ_k the sum of atomic mass fractions in species k , weighted by stoichiometric coefficients given by [123]:

$$\xi_k = \frac{2}{W_C} \times W_C \times a_{kC} + \frac{1}{2W_H} \times W_H \times a_{kH} - \frac{1}{W_O} \times W_O \times a_{kO} \quad (2.22)$$

where a_{kj} is the number of atoms j in species k . For instance: $a_{CH_4,H} = 4$, $a_{CH_4,C} = 1$ and $a_{CH_4,O} = 0$. Thus, ξ_k are known constants and Z is a local variable function of Y_k . Finally, Z is rescaled using Z^O and Z^F , the values of Z respectively in the oxidizer and fuel tanks to obtain the normalized mixture fraction:

$$z = \frac{Z - Z^O}{Z^F - Z^O} \quad (2.23)$$

From z , a local equivalence ratio can be deduced using Eq. (2.24) [3]:

$$\phi = \frac{z}{1 - z} \times s \times \frac{Y_F^0}{Y_O^0} \quad (2.24)$$

with s defined in Eq. (2.19), Y_F^0 the initial fuel mass fraction in the fuel tank and Y_O^0 the initial oxidizer mass fraction in the oxidizer tank. For premixed flames, z is constant across the domain so that $\phi = \phi_g$ in that case as mentioned at the beginning of this paragraph.

2.1.4. Progress variable

The progress variable c quantifies the evolution of the chemical system through a flame, from the fresh gases, where $c = 0$, towards the burnt gases where $c = 1$. For simple cases, it can be based either on temperature as in Eq. (2.25):

$$c = \frac{T - T^f}{T^b - T^f} \quad (2.25)$$

or on fuel mass fraction as in Eq. (2.26):

$$c = \frac{Y_F - Y_F^f}{Y_F^b - Y_F^f} \quad (2.26)$$

with the superscripts f and b referring respectively to fresh and burnt gas conditions. In practice for more complex cases, it is usually a linear combination of some species mass fractions. The progress variable is very useful to localize the flame front as an iso-surface of c , leading to the normal vector to the flame front (pointing towards the fresh gases):

$$\mathbf{n}_{ff} = -\frac{\nabla c}{\|\nabla c\|_2} \quad (2.27)$$

where underscript ff means *flame front* and ∇ and $\|\cdot\|_2$ are respectively the gradient and L^2 -norm operators.

2.1.5. Laminar flame speed

The laminar flame speed S_L is a key property of premixed flames, also named propagation speed. It is defined in laminar flow, for a steady and unstretched flame. It may be related to multiple velocities that can be considered:

- (i) The absolute speed S_a .
- (ii) The displacement speed S_d .

(iii) The consumption speed S_c .

The first two velocities are local quantities whereas the last one is a global one. In this work, only S_L and S_c have been used and are briefly recalled respectively in paragraphs 2.1.5.1 and 2.1.5.2. The absolute speed is the velocity at which the flame moves in a fixed reference, while the displacement speed is the velocity at which the flame moves relatively to the flow.

2.1.5.1. The propagation speed

The propagation speed S_L is the speed at which a one-dimensional premixed flame propagates in a quiescent fresh gas mixture. The flame is steady in the sense that its structure does not change, in other words, it corresponds to a steady solution in a reference frame moving at speed S_L . In that case S_L can be expressed thanks to the mass balance equation:

$$S_L = \frac{\rho^b u^b}{\rho^f} = \frac{\rho u}{\rho^f} \quad (2.28)$$

expressing that in one-dimensional steady case, ρu is constant.

2.1.5.2. The consumption speed

The consumption speed is the integral of the fuel consumption rate $\dot{\omega}_F$ in the direction normal to the flame front. It is defined by:

$$S_c = \frac{-1}{\rho^f Y_F^f - \rho^b Y_F^b} \int_{-\infty}^{+\infty} \dot{\omega}_F d\mathbf{n}_{ff} \quad (2.29)$$

2.1.5.3. Velocities in laminar steady and unstretched flame

For a steady laminar unstretched flame, it can be shown that:

$$S_a = 0 \quad (2.30)$$

$$S_d = u - S_a = u \quad (2.31)$$

$$S_L = S_c = \frac{\rho}{\rho^f} S_d \quad (2.32)$$

In that case, S_L is often written S_L^0 and corresponds to what is called the unstretched laminar flame speed.

2.1.6. Laminar flame thickness

The flame thickness is also a very important property of a flame, in particular when using numerical simulations since it controls the required mesh resolution. Usually, it is obtained from the temperature profile through the flame with the following formula:

$$\delta_L^0 = \frac{T^b - T^f}{\max(|\nabla T \cdot \mathbf{n}_{ff}|_2)} \quad (2.33)$$

Using Eq. (2.33) requires a flame solution. It can be replaced by an approximate estimation, obtained for a constant Prandtl number (see paragraph 2.4.5.3), constant C_p and a Sutherland law for the dynamic viscosity (see paragraph 2.4.5.1), known as the Blint's formula [124]:

$$\delta_L^{blint} = 2 \frac{D_{th}}{S_L} \left(\frac{T^b}{T^f} \right)^{0.7} \quad (2.34)$$

where $D_{th} = \lambda^f / \rho^f C_p$ is the thermal diffusivity of the fresh gases, λ^f being the thermal conductivity. The flame thickness given by Eq. (2.34) only requires the flame speed and the burnt gases temperature, which can be obtained from a thermochemical equilibrium computation. It should be mentioned that the flame thicknesses given by Eq. (2.33) and Eq. (2.34) differ, and are also far from the thickness of the reaction zone (see paragraph 2.2.2 for a definition of the reaction rate), noted δ_r , and which is even smaller.

2.2. Chemical kinetics

The N_r chemical reactions occurring during a combustion process involving N_s species are written:

$$\sum_{k=1}^{N_s} \nu'_{kj} M_k \rightleftharpoons \sum_{k=1}^{N_s} \nu''_{kj} M_k \text{ for } j = 1, N_r \quad (2.35)$$

where M_k is the symbol used for species k , ν'_{kj} and ν''_{kj} are the molar stoichiometric coefficients of species k in reaction j , in the reactants and in the products respectively. The double arrow means that the reaction can occur in both directions, from left to right (forward) or right to left (backward). To ensure mass conservation, they must satisfy:

$$\sum_{k=1}^{N_s} \nu'_{kj} W_k = \sum_{k=1}^{N_s} \nu''_{kj} W_k \iff \sum_{k=1}^{N_s} \nu_{kj} W_k = 0 \quad (2.36)$$

with $\nu_{kj} = \nu''_{kj} - \nu'_{kj}$. The progress of these reactions creates or consumes species and also produces or consumes heat.

2.2.1. Rate of progress of a chemical reaction

The rate of progress of reaction j is given by:

$$Q_j = Q_{fj} - Q_{rj} = K_{fj} \prod_{k=1}^{N_s} [X_k]^{\nu'_{kj}} - K_{rj} \prod_{k=1}^{N_s} [X_k]^{\nu''_{kj}} \quad (2.37)$$

with Q_{fj} and Q_{rj} respectively the forward and backward rates of progress, $[X_k]$ the molar concentration per unit volume of species k , and K_{fj} and K_{rj} respectively the forward and backward reaction rates constants. They are formulated with the Arrhenius law:

$$K_j(T) = A_j T^{\beta_j} \exp\left(\frac{-Ea_j}{RT}\right) = A_j T^{\beta_j} \exp\left(\frac{-Ta_j}{T}\right) \quad (2.38)$$

where K_j refers to K_{fj} or K_{rj} , A_j is the pre-exponential factor, β_j is the temperature exponent, $Ta_j = Ea_j/\bar{R}$ and Ea_j are respectively the activation temperature and activation energy. The values of A_j , β_j and Ea_j are provided in tables using dedicated formats such as CHEMKIN [125] or CANTERA [126]. Eq. (2.38) gives a way to compute both K_{fj} and K_{rj} independently, but these two quantities are linked via the equilibrium constant of reaction j noted K_{eq}^j :

$$K_{eq}^j(T) \equiv \frac{K_{fj}(T)}{K_{rj}(T)} = \left(\frac{P_0}{RT}\right)^{\sum_{k=1}^{N_s} \nu_{kj}} \exp\left(\frac{\Delta S_j^0(T)}{\bar{R}} - \frac{\Delta H_j^0(T)}{RT}\right) \quad (2.39)$$

where $P_0 = 1$ bar, ΔS_j^0 [J.mol⁻¹.K⁻¹] and ΔH_j^0 [J.mol⁻¹] are respectively the molar entropy and enthalpy changes in reaction j given by:

$$\Delta S_j^0(T) = \sum_{k=1}^{N_s} \nu_{kj} W_k s_k(T) \quad (2.40)$$

$$\Delta H_j^0(T) = \sum_{k=1}^{N_s} \nu_{kj} W_k \left(\Delta h_{f,k}^0 + h_{sk}(T) \right) \quad (2.41)$$

with $s_k(T)$ [J.kg⁻¹.K⁻¹] the mass entropy of species k at temperature T . Therefore, the backward reaction rate is usually computed from the forward reaction rate and the equilibrium constant as:

$$K_{rj}(T) \equiv \frac{K_{fj}(T)}{K_{eq}^j(T)} \quad (2.42)$$

2.2.2. Species source term

Using the rates of progress, each species k will be either produced or consumed according to a source term $\dot{\omega}_k$ [kg.m⁻³.s⁻¹] called reaction rate or net production rate. It is the sum of the reaction rates $\dot{\omega}_{kj}$ generated by the N_r reactions:

$$\dot{\omega}_k = \sum_{j=1}^{N_r} \dot{\omega}_{kj} = W_k \sum_{j=1}^{N_r} \nu_{kj} Q_j \quad \text{for } k = 1, N_s \quad (2.43)$$

where $\dot{\omega}_{kj} = W_k \nu_{kj} Q_j$ is the reaction rate of species k in reaction j so that Q_j is in mol.m⁻³.s⁻¹. This reaction rate defined in Eq. (2.43) describes the rate of production (if positive) or destruction (if negative) of species k due to the chemical reactions. Its units show that one kg of species k is either produced or consumed (depending on the sign of $\dot{\omega}_k$) per unit volume and time. If all reaction rates $\dot{\omega}_k$ are added together and thanks to Eq. (2.36), it can be seen that total mass is conserved during a combustion process since:

$$\sum_{k=1}^{N_s} \dot{\omega}_k = \sum_{k=1}^{N_s} \left(W_k \sum_{j=1}^{N_r} \nu_{kj} Q_j \right) = \sum_{j=1}^{N_r} \left(Q_j \underbrace{\sum_{k=1}^{N_s} \nu_{kj} W_k}_{=0} \right) = 0 \quad (2.44)$$

Note that units of reaction rates constants depend on the stoichiometric coefficients of each reaction. According to Eq. (2.37), since Q_j is in mol.m⁻³.s⁻¹ and $[X_k]$ in mol.m⁻³, K_{fj} and K_{rj} have units that

respectively depend on $\sum_{k=1}^{N_s} \nu'_{kj}$ and $\sum_{k=1}^{N_s} \nu''_{kj}$. K_{fj} will be in mol.m⁻³.s⁻¹ / (mol.m⁻³) ^{$\sum_{k=1}^{N_s} \nu'_{kj}$} and K_{rj} in mol.m⁻³.s⁻¹ / (mol.m⁻³) ^{$\sum_{k=1}^{N_s} \nu''_{kj}$} .

2.2.3. Heat release rate

The heat generated by the combustion process through chemical reactions results from the balance of reactants - products formation enthalpies. The heat release rate (HRR) then writes:

$$\dot{\omega}_T = - \sum_{k=1}^{N_s} \Delta h_{f,k}^0 \dot{\omega}_k \quad (2.45)$$

Its units in the SI system are: kg.m⁻¹.s⁻³ which can be also seen as J.m⁻³.s⁻¹. Therefore, the heat release rate is an energy released per unit volume and time.

2.3. Choice of the chemical description

The chemical reactions happening during a combustion process are described in what is called a chemical mechanism or also a chemical scheme. Basically, such mechanism contains the different reactions introduced in Eq. (2.35) along with their rate constants needed to compute the rates of progress of each reaction. Different kinds of mechanisms are available in the literature with different levels of description of the chemistry, from detailed chemistry to Analytically Reduced Chemistry (ARC) and simplified Globally Reduced Chemistry (GRC). This section explains the main features of these three types of chemistry.

2.3.1. Detailed chemistry

Detailed mechanisms are employed for very accurate simulations focusing on the chemical processes of combustion. They can be composed of hundreds of species reacting through thousands of elementary reactions. As an example, GRI mechanisms [127, 128] describing the combustion of natural gas, i.e., lightest hydrocarbon fuels, already contain 53 species and 325 reactions. For heavier hydrocarbons, the CRECK modeling group proposes a full detailed mechanism as the POLIMI_C1_C16 [129] which counts 368 species and 14462 reactions. These detailed mechanisms are built to reproduce experimental data from shock tube experiments and for major flame properties such as auto-ignition, extinction, laminar flame speed. They are thus quite generic and able to describe chemistry in all combustion situations. However, their complexity make them unsuitable for both academic and industrial 3D configurations. More precisely,

- (i) their extremely high number of species dramatically increases the system to be solved since a transport equation as the one in Eq. (2.91) has to be solved for each species,
- (ii) they are very stiff, due to highly reactive intermediate species having very short life times and requiring very small time steps,
- (iii) they demand very fine space discretization to represent all intermediate reaction steps, which may be very narrow.

Therefore, even with the constant increase of computing power, detailed chemistry is very little used in LES and DNS nowadays.

2.3.2. Globally reduced chemistry

As opposed to detailed chemistry, Globally Reduced Chemistry (GRCs) schemes do not describe the true elementary reactions, but model the process with one to four global reactions and of the order of ten transported species. They are therefore very fast to compute. Reaction rates are still expressed with an Arrhenius-like law similar to Eq. (2.38) where the rate constants are globally fitted by comparison to experimental data of flame auto-ignition and laminar flame speed. However, due to the over-simplification it is usually not possible to find a set of rate constants that is valid for a wide range of conditions. One method to overcome this issue, as in the 2S_BFER methodology [130] is to use non-constant pre-exponential factors and make them functions of the mixture conditions. This method is called the Pre-Exponential Adjustment (PEA) and is employed to correct the overestimation of the laminar flame speed in rich cases [131]. The 2S_BFER assumes a two-steps mechanism for any fuel F reacting with O_2 :



where the CO/CO_2 equilibrium (see Eq. (2.47)) is added to obtain a correct gas temperature in the rich burnt gas, and x, y and z are respectively the stoichiometric coefficients for O_2 , CO and H_2O in

the fuel oxidation reaction. For example, the CH₄/Air-2S-BFER chemical scheme [130], that is used in this work, contains six species and the two reactions Eq. (2.46) and Eq. (2.47) with $x=1.5$, $y=1$ and $z=2$. For this mechanism, the forward rates of progress are computed as [130]:

$$Q_{f1} = \overbrace{A_1 \exp\left(\frac{-Ea_1}{RT}\right)}^{K_{f1}} [\text{CH}_4]^{n_{\text{CH}_4}} [\text{O}_2]^{n_{\text{O}_2,1}} \times f_1(\phi) \quad (2.48)$$

$$Q_{f2} = \underbrace{A_2 T^{0.7} \exp\left(\frac{-Ea_2}{RT}\right)}_{K_{f2}} [\text{CO}]^{n_{\text{CO}}} [\text{O}_2]^{n_{\text{O}_2,2}} \times f_2(\phi) \quad (2.49)$$

where $n_{k,j}$ is the reaction exponent for species k in reaction j and does not to be confused with the stoichiometric coefficient of species k in reaction j . Values of the $n_{k,j}$ have been tuned to retrieve the expected pressure dependency on the laminar flame speed whereas the A_j and Ea_j are adjusted to match directly the laminar flame speed [130]. Values of pre-exponential constants, activation energy and reaction exponents are summed up in Table 2.1.

| | CH ₄ oxidation | CO – CO ₂ equilibrium |
|-----------------------------|--|--|
| Pre-exponential constant | 4.9×10^9 [mol ^{-0.15} .cm ^{0.45} .s ⁻¹] | 2×10^8 [mol ^{-0.5} .cm ^{1.5} .s ⁻¹] |
| Activation energy [cal/mol] | 3.55×10^4 | 1.2×10^4 |
| Reaction exponents [-] | $n_{\text{CH}_4} = 0.50$, $n_{\text{O}_2,1} = 0.65$ | $n_{\text{CO}} = 1.00$, $n_{\text{O}_2,2} = 0.50$ |

Table 2.1. – Pre-exponential constant, activation energy and reaction exponents for the CH₄/Air-2S-BFER chemical scheme.

In Eq. (2.48) and Eq. (2.49), f_1 and f_2 are the two PEA correction functions depending on the local equivalence ratio, given by [131]:

$$f_1(\phi) = \frac{2}{\left[1 + \tanh\left(\frac{\phi_{0,1} - \phi}{\sigma_{0,1}}\right)\right] + B_1 \left[1 + \tanh\left(\frac{\phi - \phi_{1,1}}{\sigma_{1,1}}\right)\right] + C_1 \left[1 + \tanh\left(\frac{\phi - \phi_{2,1}}{\sigma_{2,1}}\right)\right]} \quad (2.50)$$

$$f_2(\phi) = \frac{1}{2} \left[1 + \tanh\left(\frac{\phi_{0,2} - \phi}{\sigma_{0,2}}\right)\right] + \frac{B_2}{2} \left[1 + \tanh\left(\frac{\phi - \phi_{1,2}}{\sigma_{1,2}}\right)\right] + \frac{C_2}{2} \left[1 + \tanh\left(\frac{\phi - \phi_{2,2}}{\sigma_{2,2}}\right)\right] \times \left[1 + \tanh\left(\frac{\phi_{3,2} - \phi}{\sigma_{3,2}}\right)\right] \quad (2.51)$$

where the function parameters are gathered in Table 2.2 and the local equivalence ratio ϕ is computed from Eq. (2.24).

| | $\phi_{0,j}$ | $\sigma_{0,j}$ | B_j | $\phi_{1,j}$ | $\sigma_{1,j}$ | C_j | $\phi_{2,j}$ | $\sigma_{2,j}$ | $\phi_{3,j}$ | $\sigma_{3,j}$ |
|---------|--------------|----------------|---------------------|--------------|----------------|--------|--------------|----------------|--------------|----------------|
| $j = 1$ | 1.1 | 0.09 | 0.37 | 1.13 | 0.03 | 6.7 | 1.6 | 0.22 | - | - |
| $j = 2$ | 0.95 | 0.08 | $2.5 \cdot 10^{-5}$ | 1.3 | 0.04 | 0.0087 | 1.2 | 0.04 | 1.2 | 0.05 |

Table 2.2. – Coefficient values for $f_1(\phi)$ and $f_2(\phi)$ for the CH₄/Air-2S-BFER.

The shapes of f_1 and f_2 are represented in Fig. 2.1. The correction of the fuel oxidation reaction f_1 first increases when ϕ is above stoichiometry and later decreases to slow down combustion, while f_2 goes very fast to zero to accelerate the evolution towards equilibrium. It can also be noticed that both functions are closed to one for lean combustion since the rate constants of Table 2.1 lead to the correct flame speed in the lean side.

GRC schemes are quite attractive as they are simple and fully affordable in 3D simulations. This is however at the cost of lack of accuracy, and detail about the chemical pathways. Moreover they do not contain pollutant chemical processes which is critical in the current combustion context.

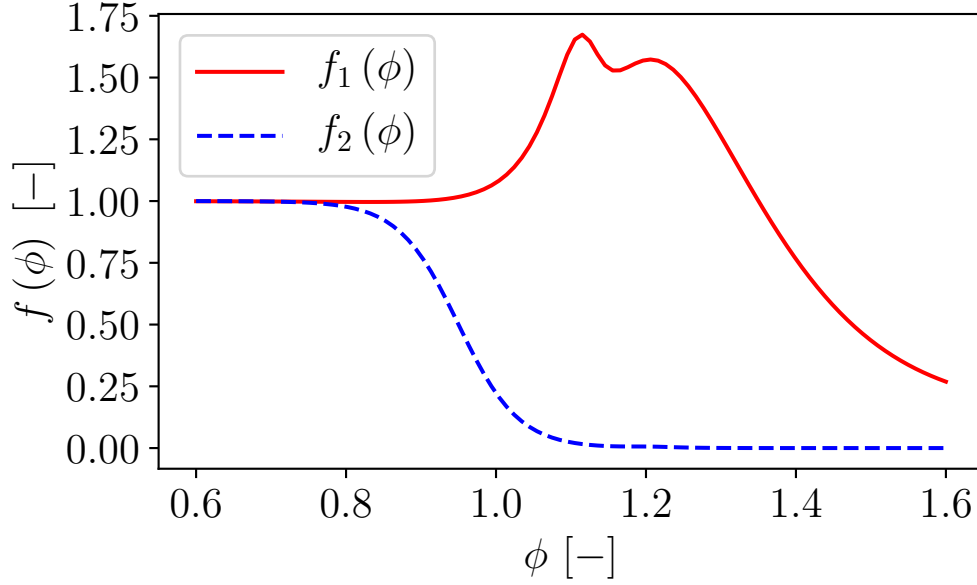


Figure 2.1. – Evolution of PEA functions f_1 and f_2 with equivalence ratio for the CH₄/Air-2S-BFER.

2.3.3. Analytically reduced chemistry

As an intermediate between detailed chemistry and GRC, the so-called Analytically Reduced Chemistry (ARC) makes a good compromise between cost and accuracy. Detailed mechanisms are reduced with physically-oriented methods in order to obtain typically between 10 to 30 species to transport which is today affordable in CFD simulations, while keeping predictive capabilities over a wide range of conditions. The reduction process is basically done in two consecutive steps:

- (i) a first step reduces the detailed mechanism to a so-called skeletal mechanism. The user selects different flame parameters to target such as flame speed, flame temperature or NO_x concentration for instance. He also gives an error tolerance on each of these parameters, noted ϵ , using the detailed mechanism results as references. With these constraints, a number of species and reactions are discarded using either sensitivity analysis as in [132], principal component analysis (PCA) [133] or graph methods such as the direct relation graph (DRG) [134] and DRG with error propagation (DRGEP) [135]. In this work, a skeletal mechanism was employed for simulating a one-dimensional hydrogen-air premixed flame. It is the so-called San Diego skeletal mechanism [136], named here as H2AIR_9_21_0_SD since it is composed of 9 transported species reacting through 21 chemical reactions.
- (ii) a second step further reduces the skeletal mechanism on the basis of a timescale analysis to spot the fast-reacting species. For the very short-time living species it may be considered that their net chemical source term is zero, i.e., they are in a Quasi-Stationary State (QSS):

$$\frac{d[X_k]}{dt} \equiv \dot{\omega}_k \approx 0 \quad (2.52)$$

Expressing $\dot{\omega}_k$ with the species concentrations then gives an estimate of the QSS species k , which therefore has no longer to be computed with a transport equation. Another advantage is that removing QSS species reduces stiffness, as by definition QSS species are the most stiff.

Several methods have been developed in the last century to identify QSS species. They are based on different analyses of characteristic times and can be classified into three groups: The Computational Singular Perturbation (CSP) method [137–139], The Partial Equilibrium Analysis (PEA) [140] and The Level Of Importance (LOI) [141] method.

The reduced mechanism obtain after both steps (skeletal mechanism and QSS) is called an analytically reduced mechanism. In this work, a methane-air ARC mechanism is employed. It has been derived from the GRI-3.0 [142] detailed mechanism with the software ARCANE [143] using successively the DRGEP method to get the skeletal mechanism followed by a LOI method to identify the QSS species. The resulting scheme is composed of 16 transported species, 250 chemical reactions and 10 species in QSS, and is named in the following as CH4_16_250_10_QC.

2.4. Conservation equations for multi-species reacting flows

The combustion phenomenon is described with the three-dimensional multi-species reacting compressible Navier-Stokes equations (NSE) which arise from the conservation of mass (for both the mixture and the species), momentum and energy. The objective of this section is to introduce these equations and explain how they are solved in CFD.

2.4.1. Conservation of mass of the mixture

The total mass conservation does not change compared to the non-reacting case since the combustion process does not modify the total mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.53)$$

where $\mathbf{u} = (u, v, w)^T$ [m.s⁻¹] is the velocity vector along physical coordinates (x, y, z) and $\nabla \cdot$ is the divergence operator applied to a vector (resulting in a scalar).

2.4.2. Conservation of momentum

The conservation of momentum in reacting flows also obeys the same equation as in a non-reacting flow:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + P \mathbf{I}) = \nabla \cdot (\boldsymbol{\tau}) \quad (2.54)$$

where \otimes means the tensor product, \mathbf{I} is the identity matrix, P is the static pressure, $\nabla \cdot$ is the divergence operator applied to a matrix (which gives a vector) and $\boldsymbol{\tau}$ [kg.m⁻¹.s⁻²] is the viscous stress tensor defined under Stokes hypothesis for a Newtonian fluid:

$$\boldsymbol{\tau} = \mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - \frac{2}{3} \nabla \cdot (\mathbf{u} \mathbf{I}) \right) \quad (2.55)$$

with μ [kg.m⁻¹.s⁻¹] the dynamic viscosity of the mixture. Although there are no reaction terms in Eq. (2.54), the flow is modified by combustion through the pressure increase as well as the temperature increase which modifies the viscosity $\mu(T)$ with a ratio from 1:8 to 1:10 [3]. Density also decreases in the same ratio through gas expansion. Consequently, the local Reynolds number varies importantly through the flame and differs in the burnt and fresh gas. For instance, a turbulent non-reacting jet may become laminar after ignition [144].

2.4.3. Conservation of mass for species

The conservation of mass for species k is an additional equation that is not present in the non-reacting case. It is due to the multi-species mixture and is given by:

$$\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho (\mathbf{u} + \mathbf{V}_k) Y_k) = \dot{\omega}_k \text{ for } k = 1, N_s \quad (2.56)$$

where \mathbf{V}_k is the diffusion velocity of species k . The diffusion velocity is a key element of the combustion process and its modeling is not straightforward as will be shown in the next paragraphs.

By summing equations Eq. (2.56) from $k = 1$ to $k = N_s$, the total mass conservation of the mixture must be recovered:

$$\begin{aligned}
 \sum_{k=1}^{N_s} \left[\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho (\mathbf{u} + \mathbf{V}_k) Y_k) \right] &= \underbrace{\sum_{k=1}^{N_s} \dot{\omega}_k}_{=0} \\
 \Rightarrow \underbrace{\frac{\partial \left(\sum_{k=1}^{N_s} \rho Y_k \right)}{\partial t}}_{\frac{\partial \rho}{\partial t}} + \nabla \cdot \underbrace{\left(\sum_{k=1}^{N_s} \rho \mathbf{u} Y_k \right)}_{\rho \mathbf{u}} + \nabla \cdot \left(\sum_{k=1}^{N_s} \rho \mathbf{V}_k Y_k \right) &= 0 \\
 \Rightarrow \underbrace{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u})}_{=0} + \rho \nabla \cdot \left(\sum_{k=1}^{N_s} \mathbf{V}_k Y_k \right) &= 0 \\
 \Rightarrow \sum_{k=1}^{N_s} Y_k \mathbf{V}_k &= 0
 \end{aligned} \tag{2.57}$$

The last equality of Eq. (2.57) is not ensured with all models for diffusion velocities and may require additional corrections.

2.4.3.1. Diffusion velocity

The diffusion velocities \mathbf{V}_k are the solutions of the following linear system of size N_s^2 which technically has to be solved in each direction at each mesh point and at each time instant (if the flow is unsteady) [145]:

$$\nabla X_m = \sum_{k=1}^{N_s} \frac{X_m X_k}{D_{mk}} (\mathbf{V}_k - \mathbf{V}_m) + (Y_m - X_m) \frac{\nabla P}{P} + \frac{\rho}{P} \sum_{k=1}^{N_s} Y_m Y_k (\mathbf{f}_m - \mathbf{f}_k) \quad \text{for } m = 1, N_s \tag{2.58}$$

where $D_{mk} = D_{km} \left[\text{m}^2 \cdot \text{s}^{-1} \right]$ is the binary mass diffusion coefficient of species m into species k and \mathbf{f}_m and \mathbf{f}_k are respectively the volume forces acting on species m and k . This system is derived by stating that the total momentum is conserved in collisions so that the momentum of species m can be changed only by collisions with molecules of other types $k \neq m$ (the two first terms on the right-hand-side (RHS) of Eq. (2.58)) and by volume forces acting on species m (the last term on RHS of Eq. (2.58)) [145]. It should be mentioned that the diffusion of mass due to temperature gradients, also called the Soret effect, is neglected in Eq. (2.58). In a CFD context, Eq. (2.58) is very costly to solve and most of industrial codes use simplified transport approaches discuss hereafter [3].

2.4.3.2. Fick's law

If pressure gradients are sufficiently small and volume forces are neglected, Eq. (2.58) reduces to:

$$\nabla X_m = \sum_{k=1}^{N_s} \frac{X_m X_k}{D_{mk}} (\mathbf{V}_k - \mathbf{V}_m) \quad \text{for } m = 1, N_s \tag{2.59}$$

Eq. (2.59) can be analytically solved for two specific cases:

- (i) **Binary diffusion:** in a mixture with only two species, both having the same binary mass diffusion coefficient D_{12} . For this case, it can be shown that:

$$\mathbf{V}_1 = -D_{12} \nabla [\ln(Y_1)] \quad (2.60)$$

and \mathbf{V}_2 is found using Eq. (2.57) with $N_s = 2$.

- (ii) **Multi-species diffusion with the same binary mass diffusion coefficient for all species:** in a mixture with $N_s > 2$ species and $D_{ij} = D$ is constant, the result shown by Eq. (2.60) can be extended to any species k of the mixture:

$$\mathbf{V}_k = -D \nabla [\ln(Y_k)] \quad (2.61)$$

Eq. (2.60) and Eq. (2.61) are respectively the Fick's law in binary and multi-species mixtures. Usually, combustion chemistry involves more than two species so that binary diffusion cannot be considered. On the other hand, the approximation $D_{ij} = D$ when $N_s > 2$ is usually too strong. Consequently, Fick's law is almost never used in CFD for combustion.

2.4.3.3. Hirschfelder and Curtiss approximation

In a lot of combustion codes, the resolution of Eq. (2.58) is replaced by the Hirschfelder and Curtiss approximation which is actually a first-order approximation of Eq. (2.58). This approximation is written as [146]:

$$\mathbf{V}_k X_k = -D_k \nabla X_k \quad (2.62)$$

or using Y_k by substituting Eq. (2.4) into Eq. (2.62):

$$\mathbf{V}_k Y_k = -\frac{D_k W_k}{W} \nabla X_k \quad (2.63)$$

D_k is not a binary diffusion coefficient but an equivalent diffusion coefficient of species k into the rest of the mixture [3]. Its expression will be discussed in paragraph 2.4.5. As opposed to Fick's law in the multi-species case, the diffusion coefficients differ between species which is expected to be more accurate. The diffusion coefficients are linked to the heat diffusivity noted D_{th} , through the Lewis numbers defined by:

$$Le_k = \frac{D_{th}}{D_k} \quad (2.64)$$

Lewis numbers are usually quasi-constants in flames of most hydrocarbon/air systems, but may vary significantly with light fuels such as hydrogen [3].

Consequently under this approximation, Eq. (2.56) can be recast into:

$$\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho \mathbf{u} Y_k) = \nabla \cdot \left(\rho D_k \frac{W_k}{W} \nabla X_k \right) + \dot{\omega}_k \text{ for } k = 1, N_s \quad (2.65)$$

As already mentioned, the total mass conservation is ensured only if the sum of equation in Eq. (2.65) for all species leads to the density conservation equation, i.e., with a zero RHS. It means that the quantity $\rho \sum_{k=1}^{N_s} D_k (W_k/W) \nabla X_k = \mathbf{0}$. This is not true in general, and is obtained with a correction velocity, noted \mathbf{V}^c , added to the diffusion flux of Eq. (2.65) which becomes:

$$\begin{aligned} \frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho \mathbf{u} Y_k) &= \nabla \cdot \left(\rho D_k \frac{W_k}{W} \nabla X_k - \rho Y_k \mathbf{V}^c \right) + \dot{\omega}_k \text{ for } k = 1, N_s \\ \iff \frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho \mathbf{u} Y_k) &= \nabla \cdot (\mathbf{M}_k) + \dot{\omega}_k \text{ for } k = 1, N_s \end{aligned} \quad (2.66)$$

where \mathbf{V}^c and the species diffusion flux \mathbf{M}_k are given by:

$$\mathbf{V}^c = \sum_{k=1}^{N_s} D_k \frac{W_k}{W} \nabla X_k \quad (2.67)$$

$$\mathbf{M}_k = \rho \left(D_k \frac{W_k}{W} \nabla X_k - Y_k \mathbf{V}^c \right) \quad (2.68)$$

The Hirschfelder and Curtiss approximation uses mole fractions in Eq. (2.66) whereas mass fractions are usually preferred in CFD for mass conservation. Thus, the diffusion flux \mathbf{M}_k is rewritten using Y_k instead of X_k . Starting from the following result:

$$\frac{W_k}{W} \nabla X_k = \frac{W_k}{W} \nabla \left(\frac{Y_k W}{W_k} \right) = \frac{1}{W} \left(W \nabla Y_k + Y_k \underbrace{\nabla W}_{-W^2 \nabla \left(\frac{1}{W} \right)} \right) = \nabla Y_k - Y_k W \nabla \left(\frac{1}{W} \right) \quad (2.69)$$

where $\nabla (1/W)$ is deduced from ∇Y_k following Eq. (2.9):

$$\nabla \left(\frac{1}{W} \right) = \sum_{k=1}^{N_s} \frac{\nabla Y_k}{W_k} \quad (2.70)$$

\mathbf{V}^c and then \mathbf{M}_k are expressed as:

$$\mathbf{V}^c = \sum_{k=1}^{N_s} D_k \left[\nabla Y_k - Y_k W \nabla \left(\frac{1}{W} \right) \right] \quad (2.71)$$

$$\mathbf{M}_k = \rho \left(D_k \left[\nabla Y_k - Y_k W \nabla \left(\frac{1}{W} \right) \right] - Y_k \mathbf{V}^c \right) \quad (2.72)$$

It should be mentioned that although Eq. (2.67) (respectively Eq. (2.68)) and Eq. (2.71) (respectively Eq. (2.72)) are mathematically exactly equal, their numerical discretization is not. The different steps to compute \mathbf{M}_k are summed up in Algorithm 1 of paragraph B.1.1.

2.4.4. Conservation of energy

There are multiple ways to write the conservation of energy depending on which energy variable is chosen among sensible or total enthalpies/energies and temperature. All equations are listed in [3]. Here, the total non-chemical energy per unit mass $E = e_s + ||\mathbf{u}||_2^2/2$ [J.kg⁻¹] is used, with $e_s = h_s - P/\rho$ the sensible energy of the mixture. The associated conservation equation writes:

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\mathbf{u} (P + \rho E)) = \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{u} - \mathbf{q}) + \dot{\omega}_T \quad (2.73)$$

where the energy flux vector \mathbf{q} [J.s⁻¹.m⁻²] is written as:

$$\mathbf{q} = -\lambda \nabla T - \sum_{k=1}^{N_s} h_{sk} \mathbf{M}_k \quad (2.74)$$

with λ [J.s⁻¹.m⁻¹.K⁻¹] the thermal conductivity of the mixture (see paragraph 2.4.5.3). This flux considers a heat diffusion term expressed by Fourier's Law and a second term associated with the diffusion of species with different enthalpies which is specific of a multi-species gas [3].

The heat diffusion term is based on the temperature T , which may be calculated from the sensible energy $e_s = \int C_v(T)dT$ where $C_v(T)$ is the mixture heat capacity at constant volume. To avoid computing an additional gradient, the temperature gradient is obtained ∇T from the perfect gas law:

$$\nabla T = \nabla \left(\frac{PW}{\rho R} \right) = \frac{1}{R} \left[\frac{W}{\rho} \nabla P + PW \nabla \left(\frac{1}{\rho} \right) + \frac{P}{\rho} \nabla W \right] = \frac{1}{R} \left[\frac{W}{\rho} \nabla P - \frac{PW}{\rho^2} \nabla \rho - \frac{PW^2}{\rho} \nabla \left(\frac{1}{W} \right) \right] \quad (2.75)$$

which with Eq. (2.16) becomes:

$$\nabla T = \frac{T}{P} \nabla P - \frac{T}{\rho} \nabla \rho - TW \nabla \left(\frac{1}{W} \right) \quad (2.76)$$

Finally, the energy flux introduced in Eq. (2.74) is given by:

$$\mathbf{q} = -\lambda \left(\frac{T}{P} \nabla P - \frac{T}{\rho} \nabla \rho - TW \nabla \left(\frac{1}{W} \right) \right) - \sum_{k=1}^{N_s} h_{sk} \mathbf{M}_k \quad (2.77)$$

The different steps to compute \mathbf{q} are summed up in Algorithm 2 of paragraph B.1.2.

2.4.5. Models for diffusion coefficients

In momentum, species and energy equations, the following diffusion coefficients appear:

- Dynamic viscosity μ in the momentum equations.
- Diffusion coefficient D_k of species k into the mixture in species equations.
- Thermal conductivity of the mixture λ in the energy equation.

2.4.5.1. Dynamic viscosity

Based on the kinetic theory of gases, the dynamic viscosity of a mixture can be computed using the Wilke model [147]:

$$\mu = \sum_{i=1}^{N_s} \frac{X_i \mu_i}{X_i + \sum_{j=1, i \neq j}^{N_s} X_j \phi_{ij}} \quad (2.78)$$

where ϕ_{ij} is given by:

$$\phi_{ij} = \frac{\left(1 + \left(\frac{W_j}{W_i} \right)^{1/4} \sqrt{\frac{\mu_i}{\mu_j}} \right)^2}{\sqrt{8 \left(1 + \frac{W_i}{W_j} \right)}} \quad (2.79)$$

with μ_k the viscosity of species k .

However simpler models exist with little influence on the results. In particular it can be assumed that μ is independent of the gas composition and only depends on the temperature, as summed up in Table 2.3. For both Sutherland and Power laws, T_{ref} and μ_{ref} are respectively a reference temperature and the dynamic viscosity at this reference temperature.

In this work, power laws with user-defined parameters were employed for the three different mechanisms used. The values of these parameters are gathered in Table 2.4.

| Sutherland's law | Power law |
|---|---|
| $\mu(T) = \mu_{ref} \left(\frac{T}{T_{ref}} \right)^{3/2} \frac{T_{ref} + S}{T + S}$ | $\mu(T) = \mu_{ref} \left(\frac{T}{T_{ref}} \right)^n$ |
| • S is the Sutherland's temperature. | • n is the power law exponent. |

Table 2.3. – Laws for dynamic viscosity as a function of temperature T only.

| Mixture name | μ_{ref} | $\text{kg.m}^{-1}.\text{s}^{-1}$ | T_{ref} [K] | n [–] |
|------------------|-----------------------|----------------------------------|---------------|---------|
| CH4/Air-2S-BFER | 1.84×10^{-5} | | 300 | 0.676 |
| CH4_16_250_10_QC | 6.87×10^{-5} | | 2129 | 0.643 |
| H2AIR_9_21_0_SD | 8.06×10^{-5} | | 2645 | 0.648 |

Table 2.4. – Values of μ_{ref} , T_{ref} and n used in the power law formula for CH4/Air-2S-BFER, CH4_16_250_10_QC and H2AIR_9_21_0_SD mechanisms.

2.4.5.2. Species diffusion coefficients

Under the Hirschfelder and Curtiss approximation, D_k is expressed as a function of the binary diffusion coefficients such that:

$$D_k = \frac{1 - Y_k}{\sum_{j \neq k} \frac{X_j}{D_{jk}}} \quad (2.80)$$

However, computing the binary diffusion coefficients is numerically costly and a further approximation may be done, assuming constant Schmidt numbers $Sc_k \equiv \mu / (\rho D_k)$ for all species k so that Eq. (2.80) is replaced by Eq. (2.81):

$$D_k = \frac{\mu}{\rho Sc_k} \quad (2.81)$$

Schmidt numbers are specified as inputs to define the considered mixture. Their values are determined so as to reproduce the correct flame behavior, in particular the laminar flame speed.

2.4.5.3. Thermal conductivity of the mixture

As for dynamic viscosity, a formula for the thermal conductivity of a mixture can be obtained from kinetic theory of gases using the Mathur model [148]:

$$\lambda = \frac{1}{2} \left(\sum_{k=1}^{N_s} X_k \lambda_k + \frac{1}{\sum_{k=1}^{N_s} \frac{X_k}{\lambda_k}} \right) \quad (2.82)$$

where λ_k is the conductivity of species k . However, as for species diffusion coefficients, an approximation is done considering that the Prandtl number $Pr \equiv \mu C_p / \lambda$ of the mixture is constant, so that λ can be obtained using Eq. (2.83):

$$\lambda = \frac{\mu C_p}{Pr} \quad (2.83)$$

The Prandtl number is also specified as an input to define the mixture. The Prandtl numbers associated to each of the three mechanisms used in this work are gathered in Table 2.5.

| Mixture name | Pr [–] |
|------------------|----------|
| CH4/Air-2S-BFER | 0.7 |
| CH4_16_250_10_QC | 0.673 |
| H2AIR_9_21_0_SD | 0.662 |

Table 2.5. – Values of Prandtl numbers for CH4/Air-2S-BFER, CH4_16_250_10_QC and H2AIR_9_21_0_SD mechanisms.

2.4.6. Computing pressure and temperature from transported variables

Both the temperature and the pressure explicitly appear in the conservation equations described above, and must therefore be calculated. However this is not direct for a multi-species thermally perfect gas mixture, as $\rho e_s(T, Y_k)$, T and P are not anymore linked through a simple linear relation as it was the case in the mono-species mixture.

In this work, all thermodynamic functions for each species k are tabulated. Starting from the JANAF enthalpy tables [149] giving $h_{sk}(T)$ for 100 K intervals $[T_1^n, T_2^n]$, tables for e_{sk} , C_{pk} and C_{vk} are generated:

$$e_{sk}(T_1^n) = h_{sk}(T_1^n) - \frac{\bar{R}T_1^n}{W_k}, C_{pk}(T_1^n) = \frac{h_{sk}(T_2^n) - h_{sk}(T_1^n)}{T_2^n - T_1^n}, C_{vk}(T_1^n) = \frac{e_{sk}(T_2^n) - e_{sk}(T_1^n)}{T_2^n - T_1^n} \quad (2.84)$$

Consequently, the calculation of pressure and temperature from the conservative variables is done in five steps:

1. Compute ρe_s from $\rho e_s = \rho E - \|\rho \mathbf{u}\|_2^2 / (2\rho)$.
2. For each interval $[T_1^n, T_2^n]$, because the ρY_k are known, compute:

$$\rho e_s(T_1^n) = \sum_{k=1}^{N_s} \rho Y_k e_{sk}(T_1^n) \text{ and } \rho e_s(T_2^n) = \sum_{k=1}^{N_s} \rho Y_k e_{sk}(T_2^n) \quad (2.85)$$

3. When $\rho e_s(T_1^n) \leq \rho e_s \leq \rho e_s(T_2^n)$, compute T as:

$$T = \left(n + \frac{\rho e_s - \rho e_s(T_1^n)}{\rho e_s(T_2^n) - \rho e_s(T_1^n)} \right) \times 100 \quad (2.86)$$

4. Once T is known, compute from JANAF tables $C_{pk}(T)$, $C_{vk}(T)$, and

$$R_{gas} \equiv C_p(T) - C_v(T) = \sum_{k=1}^{N_s} (\rho Y_k) [C_{pk}(T) - C_{vk}(T)] / \rho \quad (2.87)$$

5. Finally, get P from Eq. (2.16).

2.4.7. Summary of equations

The transport equations in conservative form for a combustion problem in 3D under Hirschfelder and Curtiss approximation are:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.88)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + P \mathbf{I}) = \nabla \cdot (\boldsymbol{\tau}) \quad (2.89)$$

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\mathbf{u} (P + \rho E)) = \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{u} - \mathbf{q}) + \dot{\omega}_T \quad (2.90)$$

$$\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho \mathbf{u} Y_k) = \nabla \cdot (\mathbf{M}_k) + \dot{\omega}_k \text{ for } k = 1, N_s \quad (2.91)$$

Actually, Eq. (2.88) is not necessary since for a multi-species gas:

$$\rho = \sum_{k=1}^{N_s} \rho Y_k \quad (2.92)$$

Consequently in practice, Eq. (2.88) is not solved and the density is obtained by using Eq. (2.92).

2.5. Summary of this chapter

In this chapter, variables used to describe multi-species turbulent reacting flows have been introduced in a context of ideal gas. The main difference with a mono-species gas is that all thermodynamic data vary with the local composition. Basic flame properties, the flame speed and the flame thickness have also been introduced since they are essential to describe the combustion process.

A general description of the computation of combustion source terms has also been presented. The combustion process generates heat thanks to the chemical reactions for which reaction rates are modeled using an Arrhenius law. Combustion chemistry is essential and may be described with different levels of detail, usually grouped in three categories: the detailed, global and reduced (or semi-detailed) chemical schemes. It appears that detailed chemistry is well too expensive and not relevant for the purpose of this work which is to show that combustion can be successfully simulated using the SD method. Therefore only global and reduced (ARC) schemes are used here.

Finally, the equations modeling the combustion process have been introduced. They are based on the 3D multi-species NSE under Hirschfelder and Curtiss approximation where a constant Schmidt number for each species is assumed. These equations will be solved using the SD method for both reacting and non-reacting flows, as will be detailed in chapter 4.

The equations presented in this chapter describe all combustion processes, steady or unsteady, at all scales in space and time. They correspond to the equations solved in the fully resolved Direct Numerical Simulation (DNS) mode, without any model for turbulence and turbulent combustion. However, most combustion applications can not be computed with a DNS approach, which would be far too costly in computing time. That is why, modeling of turbulent reactive flows has to be introduced, either in RANS or LES contexts. This is the topic of the next chapter focusing on turbulent premixed combustion.

Turbulent premixed combustion

Overview

| | |
|--|-----------|
| 3.1. Simulation of turbulent flows | 36 |
| 3.1.1. The concept of energy cascade | 36 |
| 3.1.2. Different approaches to simulate turbulent flows | 37 |
| 3.2. Large Eddy Simulation of turbulent premixed combustion | 38 |
| 3.2.1. LES filters | 38 |
| 3.2.2. Favre-filtered LES equations for turbulent reactive flows | 39 |
| 3.3. The Thickened Flame model for Large Eddy Simulations | 42 |
| 3.3.1. Principle of the Thickened Flame model | 42 |
| 3.3.2. How to compute the thickening factor | 43 |
| 3.3.3. Subgrid-scale modeling in the Thickened Flame model | 44 |
| 3.3.4. Models for the wrinkling factor | 45 |
| 3.3.5. Synthesis on the use of the TFLES model | 48 |
| 3.4. Summary of this chapter | 48 |

This chapter is dedicated to turbulent premixed combustion and its modeling in LES. Firstly, the physics of turbulence and their modeling in the three main families of simulation methods are recalled. Among them, the LES approach is retained in this work as it appears to make a good compromise between cost and accuracy. Thus, turbulent combustion equations in the context of LES are then introduced along with the unclosed subgrid-scale terms that need to be modeled. Models for these unclosed terms are finally described, including turbulent transport and turbulent combustion. A focus is made on the TFLES combustion model used in this work.

3.1. Simulation of turbulent flows

The objective of this section is to introduce briefly the phenomena involved in non-reacting and reacting turbulent flows and how they can be simulated using CFD.

3.1.1. The concept of energy cascade

Turbulence is a complex, non-linear phenomenon and its description is one of the most challenging topic in the field of fluid mechanics. A turbulent flow is identified when flow variables (pressure, velocities, temperature, species mass fractions...) fluctuate randomly within structures having a defined spectral content. Any of these variables, noted f can be decomposed into its statistical average part \bar{f} and its fluctuating part f' so that:

$$f = \bar{f} + f' \text{ with } \bar{f'} = 0 \quad (3.1)$$

Eq. (3.1) is referred to as the Reynolds decomposition [150]. To determine whether a given flow is laminar or turbulent, the Reynolds number defined in Eq. (3.2) is used:

$$\text{Re} = \frac{UL}{\nu} \quad (3.2)$$

where U and L are respectively a characteristic velocity and length scale of the flow and $\nu = \mu/\rho$ is the kinematic viscosity of the fluid. This number evaluates the ratio between inertial and viscous forces. Laminar flows are characterized by small values of Re where viscous effects dominate, while for large values of Re inertial forces are sufficiently strong to trigger turbulence. An important process in turbulent flows is the energy cascade, firstly introduced by Richardson [151] in 1922 through this famous quotation:

*Big whorls have little whorls,
Which feed on their velocity;
And little whorls have lesser whorls,
And so on to viscosity.*

It means that a turbulent flow may be viewed as a set of vortices, also called eddies, of various sizes and characteristic life times interacting with each other, where the large eddies break up and transfer their energy to the smaller ones and so on. This concept was formalised in 1941 by Kolmogorov [152] in the case of a homogeneous isotropic turbulent flow, suggesting that most turbulent eddies range between an integral length scale l_t , for the most energetic eddies, and the Kolmogorov length scale η_κ for the least energetic eddies. Using a turbulent Reynolds number computed with the integral length scale l_t and the turbulent velocity fluctuation at l_t , noted U' ,

$$\text{Re}_t = \frac{U'l_t}{\nu} \quad (3.3)$$

the ratio of the largest to the smallest scales in the energy cascade can be expressed as [150]:

$$\frac{l_t}{\eta_\kappa} = \text{Re}_t^{3/4} \quad (3.4)$$

In between these two length scales, there is the inertial zone which is an intermediate range of scales where energy is transferred from largest to smallest scales as initially suggested by Richardson [151]. The energy cascade is illustrated on Figure 3.1 where mass kinetic energy density E of each eddy is plotted against its wavenumber κ , i.e., inversely proportional to its size. In the case of homogeneous isotropic turbulence, the kinetic energy in the inertial range decreases linearly with κ in a log-log scale, with a slope of $-5/3$: $E \propto \kappa^{-5/3}$.

The introduction of these three regions for turbulent scales is the starting point of the different classes of method used in CFD to simulate turbulent flows.

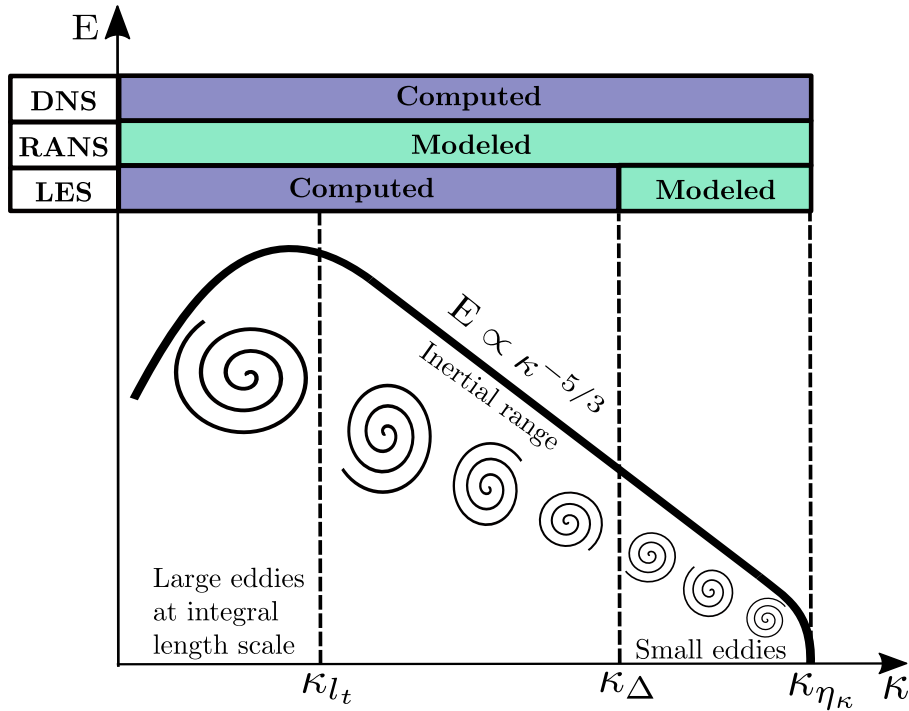


Figure 3.1. – Sketch of the energy cascade showing mass kinetic energy density $E \left[\text{m}^3 \cdot \text{s}^{-2} \right]$ versus wavenumber $\kappa \left[\text{m}^{-1} \right]$, in log-log scale, for homogeneous isotropic turbulence.

3.1.2. Different approaches to simulate turbulent flows

3.1.2.1. Direct Numerical Simulation

The reactive NSE have been introduced in Section 2.4. Solving these equations for all turbulent and combustion scales is called the DNS approach as highlighted on Figure 3.1. In that case, the mesh characteristic size Δ_x is smaller than both η_κ and δ_L^0/n_{pts}^{ff} where n_{pts}^{ff} is the number of points needed to correctly discretize the gradients within the flame front, and is usually between ten to twenty depending on the chemical scheme [3]. Although it is very accurate, this approach is still limited to simple geometries and relatively small turbulent Reynolds numbers due to its extremely high cost in terms of computational time. Indeed, it can be shown that the total number of mesh points in 3D required to correctly resolve all turbulent eddies, noted N_{pts}^{DNS} , scales as [3]:

$$N_{pts}^{\text{DNS}} \propto \text{Re}_t^{9/4} \quad (3.5)$$

This mesh size may be even more increased by the flame resolution. Current computational power allows today up to few billions grid points at most so that values of Re_t that can be simulated with DNS are of the order of 10^4 . Therefore DNS is mostly employed to better understand specific turbulence and flame phenomena, and their interactions, in order to provide guidance for modeling. A review of some DNS performed on turbulent combustion cases can be found in Poinso and Veynante [3].

3.1.2.2. Reynolds-Averaged Navier-Stokes

As opposed to DNS, the RANS approach only solves the average quantities \bar{f} defined in Eq. (3.1). Thus, RANS does not solve the equations presented in Section 2.4 but their statistical-average obtained by applying to them the average operator $\bar{\bullet}$. As a consequence, only the mean flow is accessible with RANS. In addition, as the statistical average is usually replaced by a time-average, RANS is limited to statistically steady flows. Another consequence is that all turbulent structures need to be modeled as illustrated in Figure 3.1. Mathematically, this modeling is introduced by the unresolved terms resulting from the averaging process applied to the non-linear terms of NSE. Although this approach does not give much information on the fluctuating part of the flow, it is still widely employed in industry because of its low computing cost: the resolution of average gradients can be reached on coarse meshes.

3.1.2.3. Large Eddy Simulations

LES is an intermediate approach between DNS and RANS, where the large scales of turbulence are resolved by the mesh up to a certain size Δ whereas the small scales are modeled. This is represented in Figure 3.1 where Δ is called the LES filter scale, with associated wavenumber κ_Δ , delimiting the resolved and modeled scales. The scale separation is obtained with a spatial filter which can be either explicit, using a spatial filter function (box filter, Gaussian filter...) or implicit through the mesh $\Delta = \mathcal{O}(\Delta_x)$. Basically, eddies smaller than $2\Delta_x$ will not be captured. In both cases, similarly to the RANS averaging operator, the filtering operator is applied to NSE ending up with LES equations for the filtered variables with unclosed terms related to the behavior of eddies with $\kappa > \kappa_\Delta$. LES improves very well the prediction of turbulent flows compared to RANS since most of the physics is contained within the large and intermediate scales which are fully resolved in LES. The extension of LES to turbulent combustion will be considered in Section 3.2. Usually the flame front is not fully resolved on the LES mesh and subgrid-scale (SGS) modeling is required also for the turbulent combustion.

With the constantly increasing performances of HPC, LES is becoming a viable and efficient tool to simulate turbulent combustion and is more and more employed in the industry. That is why, this approach was chosen for the simulation of turbulent combustion in this work.

3.2. Large Eddy Simulation of turbulent premixed combustion

LES of turbulent combustion relies on the modeling of both subgrid turbulent scales and subgrid flame front scales arising from the filtering of the NSE. Thus, the objective of this section is to introduce the concept of LES from a mathematical point of view, then detail the LES equations for turbulent combustion and how they are implemented in a CFD solver.

3.2.1. LES filters

Mathematically speaking, LES solves the filtered quantities, denoted by \bar{f} in this section, and defined by:

$$\bar{f}(\mathbf{x}) = \int f(\mathbf{x}') \mathcal{G}_\Delta(\mathbf{x} - \mathbf{x}') d\mathbf{x}' \quad (3.6)$$

where \mathcal{G}_Δ is the LES filter function of width Δ . As stated in paragraph 3.1.2.3, when \mathcal{G}_Δ is a mathematical function explicitly applied in the CFD code, it is called explicit LES filter whereas if the filter is the mesh it is called implicit LES filter. Eq. (3.1) still holds where \bar{f} is the filtered quantity and $f' = f - \bar{f}$ is the sub-filter (or sub-grid) part of f . Contrary to the averaging operator in RANS, the filtered sub-grid part is non-zero: $\bar{f}' \neq 0$. Three properties have to be satisfied by the LES filter function:

1. conservation of constant values, implying $\int \mathcal{G}_\Delta(\mathbf{x}') d\mathbf{x}' = 1$.
2. linearity: $\overline{f + g} = \bar{f} + \bar{g}$.
3. commutativity with time and spatial derivatives:

$$\frac{\partial \bar{f}}{\partial t} = \overline{\frac{\partial f}{\partial t}} \text{ and } \frac{\partial \bar{f}}{\partial \mathbf{x}} = \overline{\frac{\partial f}{\partial \mathbf{x}}} \quad (3.7)$$

One consequence of Eq. (3.7), is that gradient and divergence operators commute with filter operation $\bar{\bullet}$:

$$\overline{\nabla f} = \nabla \bar{f} \text{ and } \overline{\nabla \cdot f} = \nabla \cdot \bar{f} \quad (3.8)$$

The first two properties are very often verified whereas the third one is not ensured when Δ varies in space and time. However, different studies have shown that the impact of commutativity errors are negligible [153, 154].

For variable density flows such as the ones considered here, a filtering operation weighted by the density is preferred. It is called Favre-filter and is defined as:

$$\bar{\rho} \tilde{f}(\mathbf{x}) \equiv \overline{\rho f} = \int \rho f(\mathbf{x}') \mathcal{G}_\Delta(\mathbf{x} - \mathbf{x}') d\mathbf{x}' \quad (3.9)$$

where \tilde{f} is the Favre-filtered quantity of variable f so that the following decomposition can also be done:

$$f = \tilde{f} + f'' \quad (3.10)$$

with f'' the subgrid Favre-filtered part of f . As for the classical filter, applying Favre filter to f'' does not give zero: $\tilde{f}'' \neq 0$. There are two main advantages in using Favre-filtered quantities when writing LES equations for a compressible fluid [3]:

1. Less unclosed terms appear, which reduces the number of turbulent models to be used.
2. The obtained Favre-filtered equations have a similar structure than their corresponding unfiltered formulation which is easier for generalizing modeling results coming from incompressible turbulence theory.

3.2.2. Favre-filtered LES equations for turbulent reactive flows

The Favre-filtered LES equations for turbulent reactive flows are derived by applying on the reactive compressible NSE introduced in Section 2.4, the Favre-filter operator defined in Eq. (3.6). The final set of LES equations is obtained [3]:

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) = 0 \quad (3.11)$$

$$\frac{\partial \bar{\rho} \tilde{\mathbf{u}}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \otimes \tilde{\mathbf{u}} + \bar{P} \mathbf{I}) = \nabla \cdot (\bar{\boldsymbol{\tau}} + \bar{\boldsymbol{\tau}}^t) \quad (3.12)$$

$$\frac{\partial (\bar{\rho} \tilde{E})}{\partial t} + \nabla \cdot (\tilde{\mathbf{u}} (\bar{P} + \bar{\rho} \tilde{E})) = \nabla \cdot (\bar{\boldsymbol{\tau}} \cdot \tilde{\mathbf{u}} - \bar{\mathbf{q}} - \bar{\mathbf{q}}^t) + \bar{\omega}_T \quad (3.13)$$

$$\frac{\partial \bar{\rho} \tilde{Y}_k}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{Y}_k) = \nabla \cdot (\bar{\mathbf{M}}_k + \bar{\mathbf{M}}_k^t) + \bar{\omega}_k \text{ for } k = 1, N_s \quad (3.14)$$

where Eq. (3.11), Eq. (3.12), Eq. (3.13) and Eq. (3.14) are the filtered conservation equations respectively of mass, momentum, energy and species.

3.2.2.1. Filtered laminar fluxes

The filtered laminar fluxes of momentum, energy and species equations namely $\bar{\tau}$, $\bar{\tau} \cdot \bar{\mathbf{u}} - \bar{\mathbf{q}}$ and $\bar{\mathbf{M}}_k$ are computed from Favre-filtered variables \tilde{f} as follows:

$$\bar{\tau} = \overline{\mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - \frac{2}{3} \nabla \cdot (\mathbf{u} \mathbf{I}) \right)} \approx \bar{\mu} \left(\nabla \tilde{\mathbf{u}} + (\nabla \tilde{\mathbf{u}})^T - \frac{2}{3} \nabla \cdot (\tilde{\mathbf{u}} \mathbf{I}) \right) \text{ with } \bar{\mu} \approx \mu(\tilde{T}) \quad (3.15)$$

$$\bar{\tau} \cdot \bar{\mathbf{u}} - \bar{\mathbf{q}} = \overline{\bar{\tau} \cdot \bar{\mathbf{u}} + \lambda \nabla T + \sum_{k=1}^{N_s} h_{sk} \mathbf{M}_k} \approx \bar{\tau} \tilde{\mathbf{u}} + \bar{\lambda} \nabla \tilde{T} + \sum_{k=1}^{N_s} \tilde{h}_{sk} \bar{\mathbf{M}}_k \text{ with } \bar{\lambda} = \frac{\bar{\mu} C_p(\tilde{T})}{Pr} \quad (3.16)$$

$$\bar{\mathbf{M}}_k = \overline{\rho \left(D_k \left[\nabla Y_k - Y_k W \nabla \left(\frac{1}{W} \right) \right] - Y_k \mathbf{V}^c \right)} \approx \bar{\rho} \left(\bar{D}_k \left[\nabla \tilde{Y}_k - \tilde{Y}_k \tilde{W} \nabla \left(\frac{1}{\tilde{W}} \right) \right] - \tilde{Y}_k \tilde{\mathbf{V}}^c \right) \quad (3.17)$$

with $\bar{D}_k = \bar{\mu} / (\bar{\rho} Sc_k)$ and $\tilde{\mathbf{V}}^c = \sum_{k=1}^{N_s} \bar{D}_k \left[\nabla \tilde{Y}_k - \tilde{Y}_k \tilde{W} \nabla \left(\frac{1}{\tilde{W}} \right) \right]$. That is why, these filtered laminar fluxes are not usually modeled since thanks to the approximations made in Eq. (3.15), Eq. (3.16) and Eq. (3.17), they can be computed from resolved variables \tilde{f} provided by the LES solver.

3.2.2.2. Subgrid turbulent fluxes

The unclosed subgrid fluxes namely $\bar{\tau}^t$, $\bar{\mathbf{q}}^t$ and $\bar{\mathbf{M}}_k^t$ appear after the filtering operation applied to the convective terms of the reactive compressible NSE. The first one is the subgrid Reynolds stress tensor which has the following expression:

$$\bar{\tau}^t = -\bar{\rho} \left(\widetilde{\mathbf{u} \otimes \mathbf{u}} - \tilde{\mathbf{u}} \otimes \tilde{\mathbf{u}} \right) \quad (3.18)$$

This term needs to be modeled because $\widetilde{\mathbf{u} \otimes \mathbf{u}}$ is not known from the LES. An eddy-viscosity model is usually considered for the modeling of the subgrid Reynolds stress tensor where $\bar{\tau}^t$ is expressed as:

$$\bar{\tau}^t = \mu^t \left(\nabla \tilde{\mathbf{u}} + (\nabla \tilde{\mathbf{u}})^T - \frac{2}{3} \nabla \cdot (\tilde{\mathbf{u}} \mathbf{I}) \right) \quad (3.19)$$

with μ^t the SGS turbulent viscosity. Various models exist in the literature to compute μ^t : the Smagorinsky [155], Wall Adapting Local Eddy (WALE) viscosity [156], Vreman [157] or Sigma [158] models are the most common ones. They all compute μ^t from resolved velocity gradients. Recently, Chapelier and Lodato [159] developed the spectral element dynamic model (SEDM) which is well suited for the SD method and more generally for HO discontinuous spectral element methods. It actually adapts the amount of numerical dissipation within each element based on an estimate of the local subgrid turbulent kinetic energy. Consequently, elements with high p values, will have a higher value of μ^t than elements with low p values since the last ones have more numerical dissipation [159].

The second unclosed subgrid term is the subgrid energy flux which reads:

$$\bar{\mathbf{q}}^t = \bar{\rho} \left(\widetilde{\mathbf{u} E} - \tilde{\mathbf{u}} \tilde{E} \right) \quad (3.20)$$

It is commonly modeled as a diffusive contribution based on a turbulent conductivity λ^t expressed via a constant turbulent Prandtl number Pr^t specified for the computation:

$$\bar{\mathbf{q}}^t = -\lambda^t \nabla \tilde{T} \text{ with } \lambda^t = \frac{\mu^t C_p(\tilde{T})}{Pr^t} \quad (3.21)$$

Similarly, the third unclosed subgrid term, which is the subgrid species flux defined by:

$$\overline{\mathbf{M}}_k^t = -\bar{\rho} \left(\widetilde{\mathbf{u}Y_k} - \widetilde{\mathbf{u}}\widetilde{Y_k} \right) \text{ for } k = 1, N_s \quad (3.22)$$

in Eq. (3.15), is usually modeled by a diffusive contribution of the form:

$$\overline{\mathbf{M}}_k^t = \bar{\rho} \left(D_k^t \left[\nabla \widetilde{Y_k} - \widetilde{Y_k} \widetilde{W} \nabla \left(\frac{1}{\widetilde{W}} \right) \right] - \widetilde{Y_k} \widetilde{\mathbf{V}}^{c,t} \right) \text{ with } D_k^t = \frac{\mu^t}{\bar{\rho} Sc_k^t} \text{ for } k = 1, N_s \quad (3.23)$$

where D_k^t is the turbulent species diffusivity, Sc_k^t is the turbulent Schmidt number of species k specified for the computation and $\widetilde{\mathbf{V}}^{c,t}$ is the turbulent correction velocity needed, as in the laminar case, to ensure mass conservation:

$$\widetilde{\mathbf{V}}^{c,t} = \sum_{k=1}^{N_s} D_k^t \left[\nabla \widetilde{Y_k} - \widetilde{Y_k} \widetilde{W} \nabla \left(\frac{1}{\widetilde{W}} \right) \right] \quad (3.24)$$

In practice, Sc_k^t is taken constant at the same value for all species so that Eq. (3.24) leads to $\widetilde{\mathbf{V}}^{c,t} = \mathbf{0}$.

3.2.2.3. Subgrid chemical source terms

Finally, the energy and species equations contain unclosed source terms, respectively the filtered heat release rate $\bar{\omega}_T$ and the filtered reaction rates $\bar{\omega}_k$. Usually the flame thickness is much smaller than Δ so that the filtered reaction rates can not be computed from the Arrhenius law with the filtered temperature and concentrations [3]: $\bar{\omega}_k \neq \dot{\omega}_k(\widetilde{T}, \widetilde{Y_k})$. A large literature is available on the topic of modeling the filtered source terms. Examples are the models based on the G-equation [160], the surface density concept [161] or probability density functions [162]. A complete classification of LES premixed combustion models can be found in Poinso and Veynante [3]. In this work, the thickened flame (TF) approach [163] is chosen and is presented in Section 3.3.

3.2.2.4. A comment on the LES approach used in this work

During this work, no turbulence subgrid model was employed ($\mu^t = 0$), which explains why these models are not detailed in the manuscript. Mathematically speaking, this implies that all turbulent fluxes are zero:

$$\mu^t = 0 \implies \lambda^t = D^t = 0 \implies \overline{\boldsymbol{\tau}}^t = \overline{\mathbf{q}}^t = \overline{\mathbf{M}}_k^t = \mathbf{0} \quad \forall k \in \llbracket 1, N_s \rrbracket \quad (3.25)$$

Eq. (3.25) can be viewed as an implicit turbulence subgrid modeling [164, 165] since only the dissipation associated with the numerical scheme acts on the subgrid turbulent scales. On the other hand, using a turbulence subgrid model is assimilated to explicit turbulence subgrid modeling and is used to better control the energy transfers from resolved to subgrid-scales.

HO discontinuous methods are based on an upwind treatment of the convective fluxes at interfaces (Riemann solvers), and therefore contain some numerical dissipation [166]. Consequently, they can be employed in the context of implicit turbulence subgrid modeling provided that the turbulence level is not too high, as commonly done in the SD literature [167–169]. Indeed, when going to high turbulent Reynolds number, the numerical dissipation induced by the SD method is not sufficient to dissipate energy at the subgrid-scale and subgrid modeling is required, as evidenced in Chapelier *et al.* [170] and in Chapelier and Lodato [159]. It was also investigated in the context of underresolved DGSEM simulations by Gassner *et al.* [171]. Fortunately all turbulent cases studied in this work have a low Reynolds number, allowing to omit SGS turbulent viscosity. A posteriori checks using Pope's criterion [172], to evaluate the quality of the LES, have been conducted on both the Cambridge flame

burner (see paragraph 8.3.8) and the VOLVO configuration (see paragraph 8.4.7) simulated in this work. This criterion is defined as:

$$\mathcal{M}(\mathbf{x}, t) = \frac{K_{SGS}(\mathbf{x}, t)}{K_{SGS}(\mathbf{x}, t) + K_{RES}(\mathbf{x}, t)} \quad (3.26)$$

where $K_{RES} = (1/2) \overline{(\tilde{\mathbf{u}} - \widetilde{\tilde{\mathbf{u}}}) \cdot (\tilde{\mathbf{u}} - \widetilde{\tilde{\mathbf{u}}})}$ is the resolved turbulent kinetic energy (TKE) computed from the statistics of the resolved velocity field $\tilde{\mathbf{u}}$ and K_{SGS} is the subgrid-scale TKE estimated by:

$$K_{SGS} = \left(\frac{\mu^t}{\rho \Delta_{SGS}} \right)^2 \quad (3.27)$$

with $\Delta_{SGS} = C_{SGS} \Delta_x$, C_{SGS} being the SGS model constant. Pope's criterion gives a measure of the fraction of the TKE which is resolved by the LES mesh. It is commonly accepted in the CFD community that a good LES mesh must solve at least 80% of the TKE meaning that: $\mathcal{M} < 0.2$.

Finally, Eq. (3.11), Eq. (3.12), Eq. (3.13) and Eq. (3.14), combined with Eq. (3.25) and without combustion source terms for simplicity, correspond to the compressible NSE of Section 2.4. Thus, in terms of implementation, these equations are the ones that are basically solved by the CFD code with $\rho = \bar{\rho}$, $\mathbf{u} = \tilde{\mathbf{u}}$, $E = \tilde{E}$ and $Y_k = \tilde{Y}_k$ and if explicit turbulence subgrid modeling is desired it is added in the fluxes and source terms of the compressible NSE.

3.3. The Thickened Flame model for Large Eddy Simulations

The principle of the TF model is to artificially thicken the flame front making it resolvable on the LES coarse mesh. The objective of this section is to explain the basics of this model in a general framework without considering its implementation for a given numerical method. This will be detailed in Section 8.1 in the context of the SD method.

3.3.1. Principle of the Thickened Flame model

The TF model takes its origin from Butler and O'Rourke [173] and is based on laminar premixed flame properties: combustion source terms and diffusion coefficients may be modified in a way that thickens the flame while preserving the laminar flame speed. Indeed, keeping the correct flame speed is essential because it is one key quantity when designing a new combustion chamber, as it gives the speed at which the fuel is consumed. Following the theory of laminar premixed flames, the laminar flame speed S_L^0 and the flame thickness δ_L^0 scale as [145]:

$$S_L^0 \propto \sqrt{D\bar{\omega}} \quad , \quad \delta_L^0 \propto \sqrt{\frac{D}{\bar{\omega}}} \quad (3.28)$$

where D is the molecular diffusivity (gathering both thermal and species diffusivities) and $\bar{\omega}$ is the mean reaction rate. Thus, applying the following scaling operation with a factor \mathcal{F} to D and $\bar{\omega}$:

$$D \rightarrow \mathcal{F}D \quad \text{and} \quad \bar{\omega} \rightarrow \frac{\bar{\omega}}{\mathcal{F}} \quad (3.29)$$

impacts flame speed and thickness as:

$$\delta_L^0 \rightarrow \mathcal{F}\delta_L^0 \quad \text{and} \quad S_L^0 \rightarrow S_L^0 \quad (3.30)$$

Therefore, if $\mathcal{F} > 1$, a thicker flame is obtained but with the same flame speed as the original one. The variable \mathcal{F} is called the thickening factor. It can be uniform over the whole domain (constant

thickening) or locally conditioned on a flame sensor (dynamic thickening) to thicken only the reaction zone and avoid species over-diffusion and mixing outside the flame. In practice, the implementation of the TF model in a general CFD code that solves reacting compressible NSE is done as follows at each mesh point and time step:

- λ and all the D_k are multiplied by \mathcal{F} .
- All the $\dot{\omega}_k$ are divided by \mathcal{F} .

3.3.2. How to compute the thickening factor

3.3.2.1. Original formulation of Legier et al.

The constant thickening approach is the simplest way of thickening the flame front by setting a uniform value of \mathcal{F} over the whole computational domain. However, it changes the mixing properties of the flow in regions where there is no flame. Thus, a dynamic thickening procedure was introduced where \mathcal{F} is computed locally on the fly during the simulation and applied only in the flame region. This procedure is called the dynamic TF model (DTF) and was originally proposed by Legier et al. [174] where \mathcal{F} is given by:

$$\mathcal{F} = 1 + (\mathcal{F}_{max} - 1) \mathcal{S} \quad (3.31)$$

with \mathcal{S} a local flame sensor which is built to have $\mathcal{S} = 1$ inside the flame front and $\mathcal{S} = 0$ elsewhere. In Eq. (3.31), \mathcal{F}_{max} is the maximum thickening set by the user to have an expected number of points n_{pts}^{ff} of size Δ_x in the laminar flame thickness δ_L^0 :

$$\mathcal{F}_{max} = \frac{n_{pts}^{ff} \Delta_x}{\delta_L^0} \quad (3.32)$$

Values of n_{pts}^{ff} range from 5 to 20 for very stiff chemical schemes. In Legier et al. [174], \mathcal{S} was based on a hyperbolic tangent function, defined from the fuel reaction source term computed with an Arrhenius-like expression:

$$\mathcal{S}(\mathbf{x}, t) = \tanh \left(\beta' \frac{\Omega(\mathbf{x}, t)}{|\dot{\omega}_F|_{1D}^{max}} \right) \quad \text{with} \quad \Omega(\mathbf{x}, t) = Y_F^{\nu_F} Y_O^{\nu_O} \exp \left(-\Gamma \frac{Ea}{RT} \right) \quad (3.33)$$

where $\beta' = 50$, $\Gamma = 0.5$ and $|\dot{\omega}_F|_{1D}^{max}$ is the maximum absolute value of the fuel reaction source term extracted from a one-dimensional premixed flame computation in representative conditions. The activation energy Ea is the one of the fuel oxidation reaction, and the value of Γ triggers the sensor at lower temperature than the current fuel reaction in the scheme [174].

3.3.2.2. Relaxation sensor of Jaravel et al.

The sensor presented in paragraph 3.3.2.1 has some limitations highlighted in the thesis of Benedetta Franzelli [131] and is not adapted to ARC mechanisms. A more generic approach was developed in the thesis of Thomas Jaravel [175], leading to the so-called relaxation flame sensor. This new sensor is adapted to two-reactions chemistries as well as to more complex chemistries like ARC schemes. In its general formulation, the sensor is based on the fuel source term (but any other species can be employed) and is given by Eq. (3.34):

$$\mathcal{S}(\mathbf{x}, t) = \max \left[\min \left(2\mathcal{F}_{max} \frac{|\dot{\omega}_F(\mathbf{x}, t)|}{|\dot{\omega}_F|_{1D}^{max}} - 1, 1 \right), 0 \right] \quad (3.34)$$

where $|\dot{\omega}_F(\mathbf{x}, t)|$ is the absolute value of the fuel source term and \mathcal{F}_{max} is still given by Eq. (3.31). The sensor defined by Eq. (3.34) captures well the reaction zone but not the surrounding diffusion

zone where large gradients of density exist. To take into account this diffusion zone in the sensor, a fictive species $\psi(\mathbf{x}, t)$ is introduced and transported through the following equation [175]:

$$\frac{\partial \rho \psi}{\partial t} + \nabla \cdot (\rho \mathbf{u} \psi) = \nabla \cdot (\rho D_\psi \nabla \psi) + \rho \dot{\omega}_\psi \quad (3.35)$$

where D_ψ is the diffusion coefficient of ψ and $\dot{\omega}_\psi$ is a source term defined by:

$$\begin{aligned} \dot{\omega}_\psi(\mathbf{x}, t) &= \frac{\psi_0 - \psi(\mathbf{x}, t)}{\tau_0} \text{ if } \mathcal{S}(\mathbf{x}, t) > 0.8 \\ \dot{\omega}_\psi(\mathbf{x}, t) &= \frac{0 - \psi(\mathbf{x}, t)}{\tau_1} \text{ if } \mathcal{S}(\mathbf{x}, t) < 0.05 \\ \dot{\omega}_\psi(\mathbf{x}, t) &= 0 \text{ otherwise} \end{aligned} \quad (3.36)$$

where τ_0 and τ_1 are two characteristic times such that $\tau_0 \ll \tau_1$ and ψ_0 is a constant larger than one. The source term is built so as to relax ψ to zero in non-reacting zones and to ψ_0 in the flame with a spatial transition controlled by D_ψ . The way to set the four parameters ψ_0 , τ_0 , τ_1 and D_ψ of this equation is described in Section A.6. One important thing to notice is that thickening is also applied to ψ . Consequently, D_ψ is multiplied by \mathcal{F} and $\dot{\omega}_\psi$ is divided by \mathcal{F} but only in the zone where $\mathcal{S}(\mathbf{x}, t) < 0.05$ [175]. The fictive species ψ is then used to compute a filtered sensor:

$$\hat{\mathcal{S}}(\mathbf{x}, t) = \max[\min(\psi(\mathbf{x}, t), 1), \mathcal{S}(\mathbf{x}, t)] \quad (3.37)$$

which is then used in the dynamic thickening factor:

$$\mathcal{F}(\mathbf{x}, t) = (1 - \hat{\mathcal{S}}(\mathbf{x}, t)) + \hat{\mathcal{S}}(\mathbf{x}, t) \mathcal{F}_{max} \quad (3.38)$$

The relaxation sensor procedure described in this paragraph is the one that was implemented in JAGUAR during this work. Its implementation in a SD context is detailed in Section 8.1.

3.3.2.3. Generic sensor of Rochette *et al.*

More recently, a new sensor based on a geometrical analysis of the flame front was developed with the objective to be independent of the flame properties, avoiding the computation of reference flames to set sensor parameters. This sensor was developed during the thesis of Bastien Rochette and has been applied in a variety of test cases, from simple 1D laminar flames to 3D turbulent combustion cases [176]. However, as this sensor still needs further consolidation, it was decided to not implement it during this work. One important implementation issue is that this sensor requires to compute the distance to the flame front from any point of the mesh. Rochette *et al.* used the Lagrangian formalism already implemented in the AVBP solver to efficiently compute this distance but such feature does not yet exist in JAGUAR.

3.3.3. Subgrid-scale modeling in the Thickened Flame model

If the flame front thickening solves the issue of flame front resolution on the grid, it does not take into account the subgrid-scale interaction of the flame with the turbulence. Moreover, the thickening by a factor \mathcal{F} modifies the resolved flame-turbulence interactions: vortices with size smaller than $\mathcal{F}\delta_L^0$ do not wrinkle the thickened flame front whereas they are able to wrinkle the non-thickened flame front. This phenomenon was studied by Angelberger *et al.* [177] and Colin *et al.* [163] using DNS of flame/vortex and flame/turbulence interactions with a non-thickened and a thickened flame. Some of their results are illustrated in Figure 3.2 where it can be seen that the thickened flame is less wrinkled than the non-thickened flame leading to a loss of flame surface area for the thickened flame.

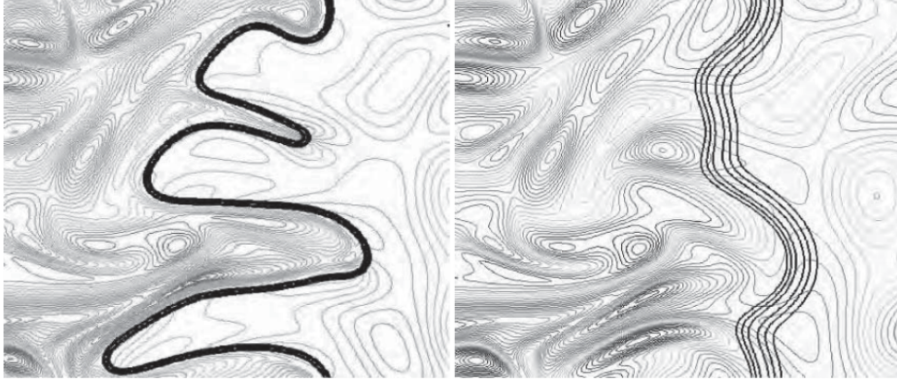


Figure 3.2. – Results of flame/turbulence interactions with non-thickened and thickened flames simulated in [163, 177]. Left: non-thickened flame. Right: thickened flame.

The modeling of these missing flame-turbulence interactions is done by introducing an efficiency function defined as the wrinkling ratio between the non-thickened flame of thickness δ_L^0 in non-filtered turbulence, and the thickened flame of thickness $\delta_L^1 = \mathcal{F}\delta_L^0$ in filtered turbulence [178]:

$$\mathcal{E} \equiv \frac{\Xi(\delta_L^0)}{\Xi(\delta_L^1)} \quad (3.39)$$

where Ξ is called the wrinkling factor of the flame surface, and represents the ratio of the SGS turbulent flame speed at the flame LES filter size $S_{T\Delta}$ and the laminar flame speed S_L^0 [178]:

$$\Xi \equiv \frac{S_{T\Delta}}{S_L^0} \quad (3.40)$$

The efficiency function \mathcal{E} is then applied as follows:

$$D \rightarrow \mathcal{E}\mathcal{F}D \quad \text{and} \quad \bar{\omega} \rightarrow \mathcal{E}\frac{\bar{\omega}}{\mathcal{F}} \quad (3.41)$$

The efficiency combined with the flame thickening constitute the TF model for LES denoted by TFLES. It is implemented in a CFD solver as follows, at each mesh point and time step:

- λ and all the D_k are multiplied by $\mathcal{E}\mathcal{F}$.
- All the $\dot{\omega}_k$ are divided by \mathcal{F} and multiplied by \mathcal{E} .

Therefore, the filtered reaction rates are finally modeled following Eq. (3.42):

$$\bar{\omega}_k = \mathcal{E} \frac{\dot{\omega}_k(\tilde{T}, \tilde{Y}_k)}{\mathcal{F}} \quad \text{for } k = 1, N_s \quad (3.42)$$

and the filtered heat release rate is deduced from the $\bar{\omega}_k$:

$$\bar{\omega}_T = \sum_{k=1}^{N_s} \Delta h_{f,k}^0 \bar{\omega}_k \quad (3.43)$$

3.3.4. Models for the wrinkling factor

Several models for the wrinkling factor can be used such as the ones proposed in Colin *et al.* [179] or Charlette *et al.* [178, 180] for the most famous. Since the Charlette model appears to be the most used in the combustion community and needs less parameters than the Colin model, it was chosen in the present work and is detailed below.

3.3.4.1. Original formulation of Charlette *et al.*

Assuming that production and destruction of the subgrid-scale flame-surface density are in equilibrium and also that the turbulent spectrum is homogeneous and isotropic at the sub-filter scale, Charlette *et al.* [178, 180] proposed the following formulation for the wrinkling factor, extracted from DNS of flame/vortex interactions:

$$\Xi(\delta_L^0) = \left(1 + \min \left[\frac{u'_\Delta}{S_L^0} \Gamma \left(\frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right), \frac{\Delta}{\delta_L^0} \right] \right)^\beta \quad (3.44)$$

where β is a model parameter whose estimation is detailed in paragraph 3.3.4.3 and Γ is a function which has the following form:

$$\Gamma \left(\frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right) = \left[\left((f_u^{-a} + f_\Delta^{-a})^{-1/a} \right)^{-b} + f_{Re}^{-b} \right]^{-1/b} \quad (3.45)$$

with f_u , f_Δ , f_{Re} , a and b given by:

$$f_u \equiv f_u \left(\frac{u'_\Delta}{S_L^0} \right) = 4 \left(\frac{27C_k}{110} \right)^{1/2} \left(\frac{18C_k}{55} \right) \left(\frac{u'_\Delta}{S_L^0} \right)^2 \approx 1.19 \left(\frac{u'_\Delta}{S_L^0} \right)^2, \quad C_k \approx 1.5 \quad (3.46)$$

$$f_\Delta \equiv f_\Delta \left(\frac{\Delta}{\delta_L^0} \right) = \left[\frac{27C_k\pi^{4/3}}{110} \times \left(\left(\frac{\Delta}{\delta_L^0} \right)^{4/3} - 1 \right) \right]^{1/2} \approx 1.30 \left(\left(\frac{\Delta}{\delta_L^0} \right)^{4/3} - 1 \right)^{1/2} \quad (3.47)$$

$$f_{Re} \equiv f_{Re}(Re_\Delta) = \left[\frac{9}{55} \exp \left(-\frac{3}{2} C_k \pi^{4/3} Re_\Delta^{-1} \right) \right]^{1/2} \times Re_\Delta^{1/2} \approx 0.40 \left[\exp \left(-10.35 Re_\Delta^{-1} \right) \right]^{1/2} \times Re_\Delta^{1/2} \quad (3.48)$$

$$a \equiv a \left(\frac{u'_\Delta}{S_L^0}, \frac{\Delta}{\delta_L^0} \right) = 0.60 + 0.20 \exp \left(-0.1 \frac{u'_\Delta}{S_L^0} \right) - 0.20 \exp \left(-0.01 \frac{\Delta}{\delta_L^0} \right) \quad (3.49)$$

$$b = 1.4 \quad (3.50)$$

In these equations:

- Δ is the LES filter size for the flame so that $\Delta = \mathcal{F}\delta_L^0 = \delta_L^1$ as in [178, 180, 181]. Consequently, it can be shown that the wrinkling factor of the thickened flame $\Xi(\delta_L^1)$ is actually equal to unity:

$$f_\Delta \left(\frac{\Delta}{\delta_L^1} \right) = f_\Delta(1) = 0 \Rightarrow \Gamma \left(\frac{\Delta}{\delta_L^1}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right) = 0 \Rightarrow \Xi(\delta_L^1) = 1 \quad (3.51)$$

- Re_Δ is the SGS Reynolds number at size Δ and is computed as [163, 178]:

$$Re_\Delta = 4 \left(\frac{\Delta}{\delta_L^0} \right) \left(\frac{u'_\Delta}{S_L^0} \right) \quad (3.52)$$

- u'_Δ is the SGS turbulent velocity up to the scale Δ estimated using [163, 182]:

$$u'_\Delta = c_2 \Delta_x^3 \|\nabla^2 (\nabla \times \tilde{\mathbf{u}})\|_2 \left(\frac{\Delta}{10\Delta_x} \right)^{1/3} \quad (3.53)$$

where $c_2 = 1/0.6 \approx 1.667$ is a model constant, Δ_x is the local mesh size and $\nabla^2 f$ and $\nabla \times f$ are respectively the Laplacian and rotational operators applied to a variable f . The computation of u'_Δ , and especially of these two operators, in the SD context will be validated in paragraph 8.1.5.

Because of Eq. (3.51), the efficiency function introduced in Eq. (3.39) for Charlette model becomes:

$$\mathcal{E} = \frac{\Xi \left(\frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right)}{1} = \left(1 + \min \left[\frac{u'_\Delta}{S_L^0} \Gamma \left(\frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right), \frac{\Delta}{\delta_L^0} \right] \right)^\beta \quad (3.54)$$

3.3.4.2. Wang *et al.* improved formulation

Later on, Wang *et al.* [181] have replaced Δ/δ_L^0 by $\Delta/\delta_L^0 - 1$ in Eq. (3.54) to maximize the wrinkling factor around Δ/δ_L^0 when u'_Δ is large, leading to:

$$\mathcal{E} = \left(1 + \max \left[0, \min \left[\frac{u'_\Delta}{S_L^0} \Gamma \left(\frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right), \frac{\Delta}{\delta_L^0} - 1 \right] \right] \right)^\beta \quad (3.55)$$

Eq. (3.55) is now the common practice when using Charlette model and is the one implemented in JAGUAR during this work. Eq. (3.55) has the following three important properties:

1. If the LES filter size for the flame becomes smaller than δ_L^0 , that is $\Delta < \delta_L^0$, no efficiency is needed anymore and $\mathcal{E} \rightarrow 1$.
2. If u'_Δ is well higher than S_L^0 , then \mathcal{E} degenerates towards a fractal-like power-law [183, 184]:

$$\mathcal{E} \rightarrow \left(\frac{\Delta}{\delta_L^0} \right)^\beta \quad (3.56)$$

3. If u'_Δ goes to zero, no efficiency is needed and $\mathcal{E} \rightarrow 1$ like in the case $\Delta < \delta_L^0$.

3.3.4.3. Estimation of β

As stated in paragraph 3.3.4.1, the value of β is not known a priori and needs to be specified. It can be seen as the parameter of the fractal-like power-law introduced in Eq. (3.56) which has actually a physical meaning [183, 184]. In the static β formulation, β is a constant for which multiple values have been proposed depending on the type of flame and flow considered. In their original paper, Charlette *et al.* [178] advised to use $\beta = 0.5$ but recent DNS of lean turbulent flames by Moureau *et al.* [185] for instance, have shown that $\beta = 0.337$ gives the closest results to the DNS. Therefore there is no consensus today on the value of β , which is probably case dependent and influenced by local flame and flow characteristics.

To overcome this issue, the dynamic β formulation was introduced in [180, 181], where β is evaluated on the fly during the computation from the filtered LES solution. The dynamic approach was applied to multiple combustion cases [186–188] leading to more realistic results than with the static β formulation. However, this formulation was not considered in this work because its implementation and validation require more time than the static formulation. It was also not considered necessary for the main purpose of this work which is to demonstrate 3D turbulent combustion simulations using the SD method.

3.3.4.4. Saturated version of the wrinkling factor

In paragraph 3.3.4.1, a formula for the SGS turbulent velocity u'_Δ was introduced through Eq. (3.53). The computation of this term can be very costly because of the third-order derivatives of the resolved velocity field that are computed at each grid point and time step. However, recent DNS of Veynante *et al.* [186] have shown that, for sufficiently turbulent flames and when β is dynamically adjusted, Eq. (3.55) very often tends to:

$$\mathcal{E} = \left(\frac{\Delta}{\delta_L^0} \right)^\beta \quad (3.57)$$

Eq. (3.57) is referred to as the saturated Charlette model. Since it avoids the computation of u'_Δ , it decreases the cost of \mathcal{E} computation with limited impact on the solution in many flame and flow configurations [186].

3.3.5. Synthesis on the use of the TFLES model

The TFLES model is composed of multiple bricks with several possible formulations. In a turbulent combustion case these are :

- The type of thickening: constant or dynamic. In the case of dynamic thickening, the corresponding flame sensor must be selected among the original, relaxation or generic sensors.
- The type of efficiency function: Colin, Charlette or any other one. In the case of Charlette model, three additional precisions have to be made:
 - ★ the type of formulation: the original one or the one of Wang.
 - ★ the β formulation: static or dynamic.
 - ★ if u'_Δ is computed or not: non-saturated or saturated versions of Charlette model. In the non-saturated case, the formula used for u'_Δ has to be specified since Eq. (3.53) is not a unique choice and other possibilities exist in the literature which can strongly impact the results [182].

For most cases considered in this work, the dynamic TFLES approach using the relaxation sensor associated to the Charlette model, in its saturated version and with the static β formulation for the efficiency function, were employed.

3.4. Summary of this chapter

In this chapter, the characteristics of turbulent premixed flows are introduced and their resolution using LES is assessed. Firstly, the concepts and usual descriptions of turbulent flows are briefly explained leading to the three main approaches commonly used in CFD for simulating turbulent flows namely RANS, LES and DNS.

LES have become very popular over the last two decades for simulating turbulent combustion due to its more realistic flow resolution compared to RANS approach and its cheaper cost compared to DNS. LES are actually able to capture most of the physics of turbulent combustion phenomena since the main flow characteristics are contained in the large scales of turbulence. However, the small scales need to be modeled to take into account, in the simulations, the phenomena happening at scales smaller than the LES filter size.

Thus, both non-reactive and reactive models are introduced with a special focus on the TFLES model which was the reactive model implemented in this work whereas implicit turbulence subgrid modeling was considered for the non-reactive unclosed terms. The purpose of the TFLES model is to artificially thicken the flame front so that the LES mesh will be able to resolve it with enough mesh points. However, a sensor is needed to apply the thickening only in the near flame region to not change mixing properties in regions outside of the flame. Moreover, an efficiency function has to be used to take into account the loss of flame surface on the thickened flame because some of the small vortices cannot wrinkle it anymore. A particular focus is made on the relaxation sensor to know where to apply the TFLES model and on Charlette's model for the efficiency function.

Part II.

The Spectral Difference method and its application to a mono-species and non-reacting gas mixture

The Spectral Difference Method

Overview

| | |
|--|-----------|
| 4.1. General principle in the one-dimensional case | 52 |
| 4.1.1. Isoparametric transformation in 1D | 52 |
| 4.1.2. Definitions of solution points and flux points | 54 |
| 4.1.3. Interpolation between SP and FP | 55 |
| 4.1.4. General algorithm for a hyperbolic 1D-equation | 56 |
| 4.1.5. Conservativity of the method | 57 |
| 4.1.6. Gradient computation in 1D for a diffusion equation: original formulation . . | 58 |
| 4.1.7. Gradient computation in 1D for a diffusion equation: SDLIFT formulation . . | 59 |
| 4.2. Formulation for hexahedral elements | 64 |
| 4.2.1. Isoparametric transformation in 3D | 65 |
| 4.2.2. Solution points and flux points in 3D | 68 |
| 4.2.3. General principle in 3D | 68 |
| 4.2.4. Practical implementation | 70 |
| 4.3. Riemann solvers for convective fluxes at interfaces | 71 |
| 4.3.1. The Riemann problem and its solvers using a 1D equation | 71 |
| 4.3.2. The HLLC Riemann solver | 72 |
| 4.4. Gradient computation in 3D for diffusive fluxes | 75 |
| 4.4.1. Original formulation | 75 |
| 4.4.2. SDLIFT formulation | 76 |
| 4.5. Local polynomial degree adaptation | 79 |
| 4.5.1. Sensors | 79 |
| 4.5.2. Load-balancing | 79 |
| 4.5.3. Methodology | 79 |
| 4.5.4. The mortar element method | 80 |
| 4.6. Temporal discretization | 81 |
| 4.6.1. Runge-Kutta schemes | 81 |
| 4.6.2. Stability conditions | 83 |
| 4.7. Summary of this chapter | 84 |

This chapter is dedicated to the discretization of the governing equations using the SD method. It starts with the one-dimensional formulation to give the basics of the method. Then the three-dimensional discretization on hexahedral elements is introduced. As the method is discontinuous across elements, Riemann solvers for convective fluxes and diffusion schemes for diffusive fluxes at interfaces are detailed along with the gradient computation within the SD framework. The local polynomial adaptation strategy using the SD method is provided and finally the temporal integration of the equations is discussed.

4.1. General principle in the one-dimensional case

The objective of this section is to describe the SD discretization process in one-dimensional configurations, and to introduce notations that will be useful in later chapters.

4.1.1. Isoparametric transformation in 1D

To introduce the basics of the SD method, let's consider a hyperbolic 1D-equation written in divergent form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} = \mathbf{0}, \text{ for } (x, t) \in \Omega \times [0, t_f] \quad (4.1)$$

where $\mathbf{U} = \mathbf{U}(x, t)$ is the vector of conservative variables (also called the solution vector), $\mathbf{E} = \mathbf{E}(\mathbf{U})$ is the flux vector of \mathbf{U} , t is the time, t_f is the final time of the computation, Ω is the physical domain of study and x represents the 1D spatial coordinate in Ω . Eq. (4.1) is a general form for 1D conservation laws with N_{eq} equations such as:

- The 1D scalar advection equation ($N_{eq} = 1$) where $\mathbf{U} = u$ and $\mathbf{E} = cu$ with c the advection speed.
- The 1D Euler equations ($N_{eq} = 3$) where $\mathbf{U} = (\rho, \rho u, \rho E)^T$ and $\mathbf{E} = (\rho u, P + \rho u^2, u(P + \rho E))^T$. They are a simplification of the NSE summarized in paragraph 2.4.7 for a mono-species gas evolving in a 1D inviscid and non-reacting flow.

To numerically solve Eq. (4.1), Ω is first divided into N_e non-overlapping and possibly non-uniform elements Ω_e :

$$\Omega = \bigcup_{e=1}^{N_e} \Omega_e \quad (4.2)$$

where $\Omega_e = \{x \in \Omega / x \in [a, b]\}$ with $h_e = b - a$ its length as shown in Figures 4.1b and 4.2b. The principle of the SD technique is to not work with element Ω_e but with a standard line $\mathcal{L} = \{\xi, 0 \leq \xi \leq 1\}$ into which each Ω_e is transformed. Thus, there is no index e for \mathcal{L} since it is the same standard line for all Ω_e . This transformation is usually written from the isoparametric or standard domain \mathcal{L} to the physical domain Ω for each element Ω_e :

$$x(\xi) = \sum_{i=1}^{N_v} M_i(\xi) x_i^e \quad (4.3)$$

with $\xi \in [0, 1]$ the 1D spatial coordinate in \mathcal{L} , x_i^e the physical coordinate of vertex i of Ω_e , N_v the number of vertices of Ω_e and M_i the shape functions. For instance, in the case of 1D linear elements composing Ω , each element Ω_e is defined by $N_v = 2$ vertices and the following shape functions [116]:

$$M_1(\xi) = 1 - \xi \text{ and } M_2(\xi) = \xi \quad (4.4)$$

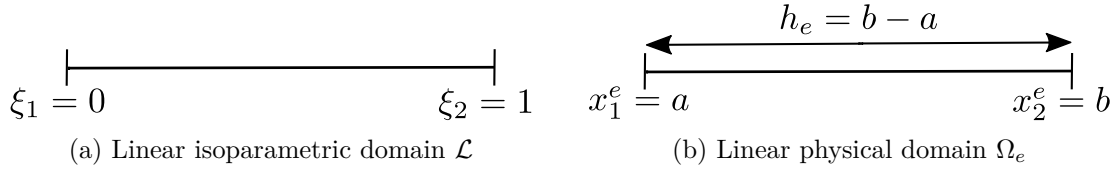


Figure 4.1. – Linear isoparametric and physical domains in 1D.

These shape functions are used to go from the linear isoparametric domain shown in Figure 4.1a to the linear physical domain illustrated in Figure 4.1b. For the 1D linear isoparametric element, shape functions M_1 and M_2 are respectively the 1D Lagrange polynomials built at ξ_1 and ξ_2 since:

$$M_1(\xi) = \frac{\xi - \xi_2}{\xi_1 - \xi_2} = \frac{\xi - 1}{0 - 1} = 1 - \xi \equiv l_1^{VT}(\xi) \quad \text{and} \quad M_2(\xi) = \frac{\xi - \xi_1}{\xi_2 - \xi_1} = \frac{\xi - 0}{1 - 0} = \xi \equiv l_2^{VT}(\xi) \quad (4.5)$$

where l_i^{VT} (VT stands for vertex) is the i -th 1D Lagrange polynomial built at vertex i using the coordinates of the remaining vertices $s \neq i$ in the isoparametric domain:

$$l_i^{VT}(\xi) = \prod_{s=1, s \neq i}^{N_v} \frac{\xi - \xi_s}{\xi_i - \xi_s}, \quad \text{for } i \in \llbracket 1, N_v \rrbracket \quad (4.6)$$

The same reasoning can be applied for quadratic elements (and higher order types of elements) which are defined with $N_v = 3$ vertices as represented in Figures 4.2a and 4.2b respectively in the isoparametric and physical domain. In that case, there are three shape functions [116]:

$$M_1(\xi) = 1 - 3\xi + 2\xi^2, \quad M_2(\xi) = 2\xi^2 - \xi \quad \text{and} \quad M_3(\xi) = 4(\xi - \xi^2) \quad (4.7)$$

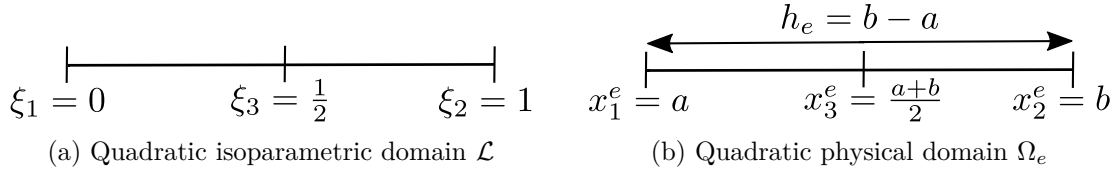


Figure 4.2. – Quadratic isoparametric and physical domains in 1D.

The isoparametric transformation is characterized by a non-singular Jacobian matrix J , which is a scalar in 1D, given by:

$$J = \frac{\partial x}{\partial \xi} \equiv x_\xi = \sum_{i=1}^{N_v} \frac{\partial M_i}{\partial \xi} x_i^e \quad (4.8)$$

The inverse transformation from the physical domain to the isoparametric domain is then related to the inverse of J defined as:

$$J^{-1} = \frac{\partial \xi}{\partial x} \equiv \xi_x \quad (4.9)$$

Consequently, Eq. (4.1) is solved in the isoparametric domain using the derivation chain rule for $\partial \mathbf{E} / \partial x$ and multiplying by the determinant of J noted $|J|$:

$$\frac{\partial \mathbf{U}}{\partial t} + J^{-1} \frac{\partial \mathbf{E}}{\partial \xi} = \mathbf{0} \Rightarrow \frac{\partial (|J| \mathbf{U})}{\partial t} + \frac{\partial (|J| J^{-1} \mathbf{E})}{\partial \xi} - \underbrace{\mathbf{E} \frac{\partial (|J| J^{-1})}{\partial \xi}}_{=0} = \mathbf{0} \quad (4.10)$$

because $|J|J^{-1} = 1$ in 1D. Variables defined in the standard domain are introduced so that Eq. (4.10) becomes:

$$\frac{\partial \hat{\mathbf{U}}}{\partial t} + \frac{\partial \hat{\mathbf{E}}}{\partial \xi} = \mathbf{0}, \text{ for } (\xi, t) \in \mathcal{L} \times [0, t_f] \quad (4.11)$$

where

$$\hat{\mathbf{U}} = |J|\mathbf{U} \quad (4.12)$$

$$\hat{\mathbf{E}} = |J|J^{-1}\mathbf{E} \quad (4.13)$$

$\hat{\mathbf{U}}$ and $\hat{\mathbf{E}}$ are respectively the solution vector and the flux vector in the isoparametric domain. The main objective of the SD discretization is the computation of the term $\partial \hat{\mathbf{E}} / \partial \xi$ corresponding to the divergence of the flux in the isoparametric domain. The formulation shown in Eq. (4.11) allows to treat each element Ω_e with exactly the same methodology to compute this flux divergence. It is worth mentioning that in Eq. (4.10), $|J|$ is assumed to be independent of time. It implies that the mesh cannot change during the computation which was the case during this work.

4.1.2. Definitions of solution points and flux points

In order to have a $(p + 1)$ -th-order of accuracy inside each element Ω_e when solving Eq. (4.11), the SD principle assumes that vector $\hat{\mathbf{U}}$ varies as a polynomial of degree p inside \mathcal{L} . It means that $\hat{\mathbf{U}}$ must be known on $p + 1$ points in \mathcal{L} called solution points (SP). Moreover, due to Eq. (4.11), the flux divergence should be a polynomial of degree p too so that $\hat{\mathbf{E}}$ has to be a polynomial of degree $p + 1$. Then, $p + 2$ points called flux points (FP) are also required inside \mathcal{L} to build a polynomial of degree $p + 1$ for $\hat{\mathbf{E}}$. The evaluation of fluxes at FP requires to have values of the solution at FP: it was shown that interpolating \mathbf{U} instead of $\hat{\mathbf{U}}$ from SP to FP is more robust especially on curved elements [30]. Consequently, it is a common practice in the SD community to build the solution polynomial using values of \mathbf{U} at SP as it will be shown in Eq. (4.18).

4.1.2.1. Solution points and degrees of freedom

In the SD community [5, 42, 49], the location of SP inside \mathcal{L} is usually taken as the Gauss-Chebyshev quadrature points of the first kind on $[0, 1]$ defined as:

$$\xi_j = \frac{1}{2} \left[1 - \cos \left(\frac{2j - 1}{2N_{SP}^{1d}} \pi \right) \right], \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.14)$$

with $N_{SP}^{1d} = p + 1$ the number of SP in 1D inside a standard element of degree p . Figures 4.3 and 4.4 illustrate the SP positions inside \mathcal{L} for $p = 2$ and $p = 3$ cases represented by red triangles. Since any element Ω_e is transformed into \mathcal{L} , SP coordinates introduced in Eq. (4.14) are the same for all elements assuming that they have the same value of p . In that case, all elements are said to be of degree p . If p -adapted simulations are considered, p can be different from one element to another and then SP coordinates will not be the same for all elements inside \mathcal{L} . Another set of SP is also employed by other authors which is the $(p + 1)$ Gauss-Legendre quadrature points [189–191]. More details on Gauss-Legendre quadrature points and Legendre polynomials are given in Appendix C. Using Gauss-Legendre quadrature points for SP should provide minimal aliasing errors [61, 159, 166]. In this work, only Gauss-Chebyshev quadrature points of the first kind were employed giving satisfying results in all cases.

The total number of SP inside the physical domain Ω is called the degrees of freedom (DOF) of the solution. More generally for any numerical method, DOF are the points where the numerical solution is stored and advanced in time. For instance, in a classical FV method, DOF corresponds to cell centers. For a fair comparison between two numerical methods in terms of accuracy and computational time, the same DOF value for both methods must be used.

4.1.2.2. Flux points

In this work, the location of FP inside \mathcal{L} are given by the Gauss-Legendre quadrature points [189–191] for the p interior FP and the two remaining FP are placed at the boundaries of \mathcal{L} namely $\xi = 0$ and $\xi = 1$. Figures 4.3 and 4.4 show the distribution of the $N_{FP}^{1d} = p + 2$ FP for $p = 2$ and $p = 3$ cases represented by blue squares. It should be mentioned that the stability of the SD method depends on the location of FP whereas the location of SP has no influence on stability. Indeed, it was found that the use of Gauss-Legendre quadrature points for the p interior FP is stable for any value of p [49, 50]. As for SP when p is the same in all elements, FP coordinates are the same in the reference element \mathcal{L} but will be different in case of p -adapted simulations.

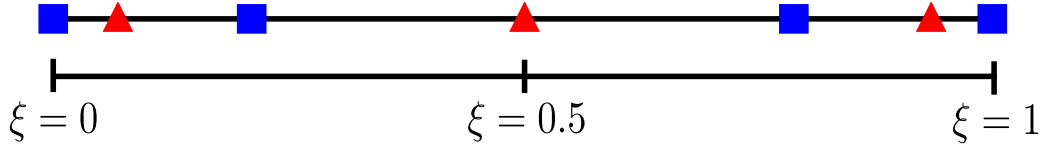


Figure 4.3. – Locations of SP (▲) and FP (■) for $p = 2$ in isoparametric domain \mathcal{L}

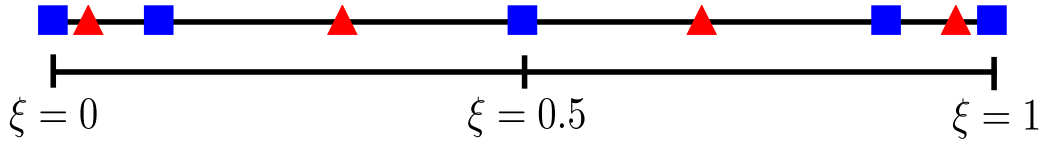


Figure 4.4. – Locations of SP (▲) and FP (■) for $p = 3$ in isoparametric domain \mathcal{L}

At this point, the two sets of points that are used in the SD method have been introduced. In this work, for a given polynomial order p , SP and FP locations in 1D will always be as:

- Gauss-Chebyshev quadrature points of the first kind on $[0, 1]$ defined by Eq. (4.14) for SP.
- Gauss-Legendre quadrature points for the p interior FP and the two ending points of \mathcal{L} for FP.

The interpolation and extrapolation processes between these sets of points can now be explained.

4.1.3. Interpolation between SP and FP

Within the SD framework, the interpolation process uses the Lagrange interpolation principle. The Lagrange polynomial basis of degree p built at SP, needed to approximate $\hat{\mathbf{U}}$ as a polynomial of degree p inside \mathcal{L} , writes:

$$l_j^{SP}(\xi) = \prod_{s=1, s \neq j}^{N_{SP}^{1d}} \frac{\xi - \xi_s}{\xi_j - \xi_s}, \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.15)$$

l_j^{SP} is then the j -th 1D polynomial of Lagrange basis built at SP ξ_j using all the other SP ξ_s with $s \neq j$. Similarly, using the values of $\hat{\mathbf{E}}$ at N_{FP}^{1d} FP, a $(p + 1)$ -degree polynomial can be built using the following Lagrange polynomial basis of degree $p + 1$:

$$l_k^{FP}(\xi) = \prod_{s=1, s \neq k}^{N_{FP}^{1d}} \frac{\xi - \xi_s}{\xi_k - \xi_s}, \text{ for } k \in \llbracket 1, N_{FP}^{1d} \rrbracket \quad (4.16)$$

l_k^{FP} is then the k -th 1D polynomial of Lagrange basis built at FP ξ_k using all the other FP ξ_s with $s \neq k$. Actually, in the SD process, the derivative of Eq. (4.16) is used when $\hat{\mathbf{E}}$ is differentiated at SP

along ξ direction. That is why its analytic formula is recalled in Eq. (4.17):

$$\frac{\partial l_k^{FP}}{\partial \xi}(\xi) = \frac{\sum_{s=1, s \neq k}^{N_{FP}^{1d}} \left[\prod_{m=1, m \neq s}^{N_{FP}^{1d}} \xi - \xi_m \right]}{\prod_{s=1, s \neq k}^{N_{FP}^{1d}} \xi_k - \xi_s}, \text{ for } k \in \llbracket 1, N_{FP}^{1d} \rrbracket \quad (4.17)$$

As l_j^{SP} , l_k^{FP} and $(\partial l_k^{FP} / \partial \xi)$ are built from SP and FP locations, their values are also independent of the mesh element in the constant p case but are element-dependent for p -adapted simulations.

4.1.4. General algorithm for a hyperbolic 1D-equation

Since the equations are solved in the reference domain, the SD discretization process can be explained inside the standard element \mathcal{L} without loss of generality. At a given time instant, the solution vector $\hat{\mathbf{U}}$ contains the conservative variables in the isoparametric domain stored at each SP ξ_j inside \mathcal{L} noted $\hat{\mathbf{U}}_j$. These values at SP are used to construct a p -degree polynomial representation of a continuous solution \mathbf{U}_h across \mathcal{L} using the Lagrange polynomial basis defined in Eq. (4.15):

$$\mathbf{U}_h(\xi) = \sum_{j=1}^{N_{SP}^{1d}} \frac{\hat{\mathbf{U}}_j}{|J|_j} l_j^{SP}(\xi), \text{ for } \xi \in [0, 1] \quad (4.18)$$

Using Eq. (4.18), conservative variables are interpolated at internal FP ($2 \leq k \leq N_{FP}^{1d} - 1$) and extrapolated at interface FP ($k = 1$ and $k = N_{FP}^{1d}$). Conservative variables at a FP k in the physical domain are denoted by \mathbf{U}_k . At an internal FP, \mathbf{E}_k is directly obtained from \mathbf{U}_k and the flux in the isoparametric domain $\hat{\mathbf{E}}_k$ is deduced from Eq. (4.13). However at an interface FP, \mathbf{U}_k is not uniquely defined since interface FP are shared by two elements. In order to have a continuous flux on the whole domain to ensure conservativity, a Riemann solver [192] at interface FP is employed to compute an interface flux noted $\hat{\mathbf{E}}_k^I$ (I stands for interface). It should be mentioned that $\hat{\mathbf{E}}_k^I$ is referred as a numerical flux since it is computed based on a numerical treatment, which is the Riemann solver, and not based on the direct link between \mathbf{E} and \mathbf{U} . Riemann solvers will be introduced in more details in Section 4.3.

Once $\hat{\mathbf{E}}_k$ at internal FP and $\hat{\mathbf{E}}_k^I$ at interface FP have been computed, a $(p+1)$ -degree polynomial representation of a continuous flux $\hat{\mathbf{E}}_h(\xi)$ across \mathcal{L} can be constructed from flux values at FP:

$$\hat{\mathbf{E}}_h(\xi) = \sum_{k=1}^{N_{FP}^{1d}} \hat{\mathbf{E}}_k l_k^{FP}(\xi), \text{ for } \xi \in [0, 1] \quad (4.19)$$

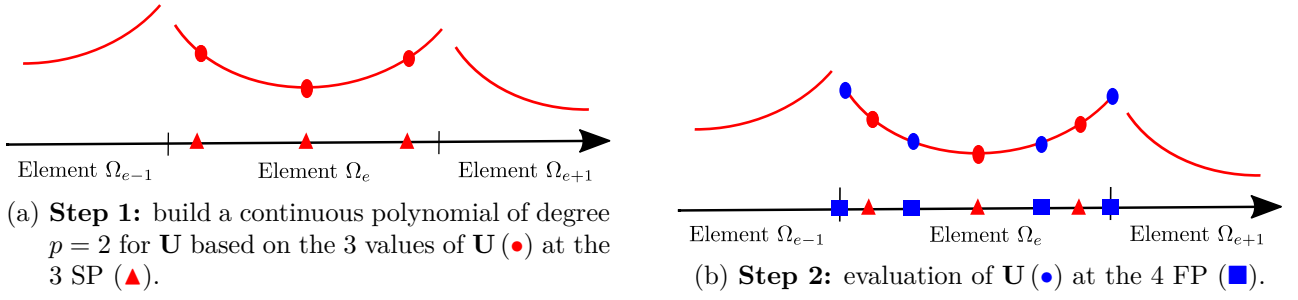
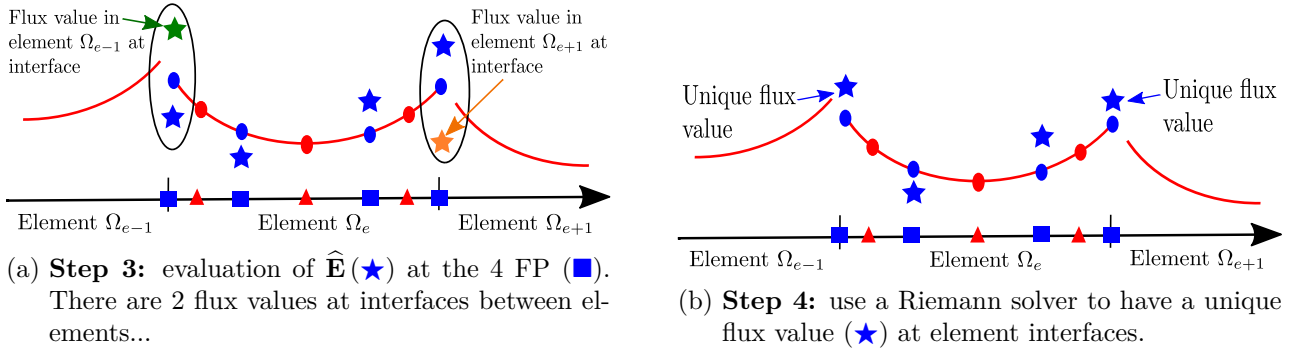
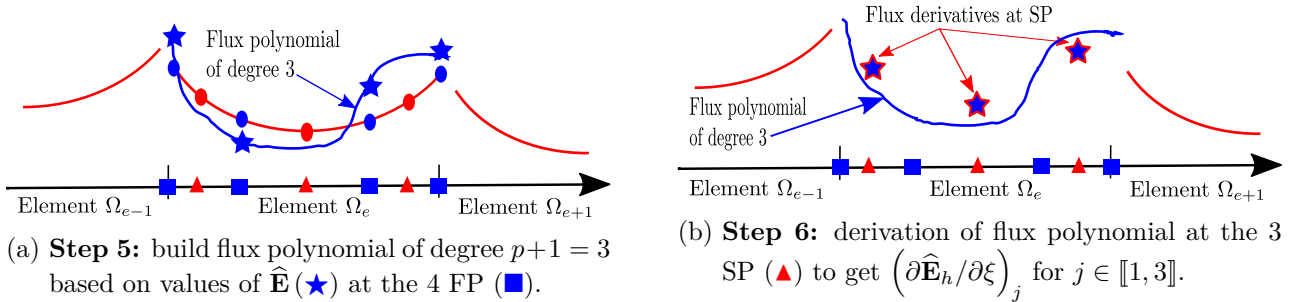
where $\hat{\mathbf{E}}_1 = \hat{\mathbf{E}}_1^I$ and $\hat{\mathbf{E}}_{N_{FP}^{1d}} = \hat{\mathbf{E}}_{N_{FP}^{1d}}^I$. The flux polynomial defined by Eq. (4.19) is differentiated along the ξ direction and evaluated at each SP ξ_j to obtain:

$$\frac{\partial \hat{\mathbf{E}}_h}{\partial \xi}(\xi_j) = \sum_{k=1}^{N_{FP}^{1d}} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j), \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.20)$$

Finally, Eq. (4.11) can be marched in time using any explicit temporal scheme at each SP inside \mathcal{L} :

$$\frac{d\hat{\mathbf{U}}}{dt}(\xi_j) = -\frac{\partial \hat{\mathbf{E}}_h}{\partial \xi}(\xi_j) = -\sum_{k=1}^{N_{FP}^{1d}} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j), \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.21)$$

The SD discretization described in this paragraph for a hyperbolic 1D-equation is summed up into six different steps that are shown from Figure 4.5a to Figure 4.7b for a $p = 2$ case. It is a situation shown for a given element Ω_e and its left (respectively right) neighbor Ω_{e-1} (respectively Ω_{e+1}) which has been transformed into the standard element \mathcal{L} for applying the SD discretization process.


 Figure 4.5. – 1st and 2nd steps of the SD process in 1D for a hyperbolic equation.

 Figure 4.6. – 3rd and 4th steps of the SD process in 1D for a hyperbolic equation.

 Figure 4.7. – 5th and 6th steps of the SD process in 1D for a hyperbolic equation.

4.1.5. Conservativity of the method

For any numerical method, it is important to verify that the discretization process is conservative: the temporal derivative of the mean value of $\hat{\mathbf{U}}$ inside \mathcal{L} must be equal to the flux difference between its left ($k = 1$) and right ($k = N_{FP}^{1d}$) faces. Since $\hat{\mathbf{U}}$ is defined at N_{SP}^{1d} Gauss-Chebyshev quadrature points, a quadrature rule can be defined on these points that integrates exactly a polynomial of degree p at most. This quadrature rule is slightly different from the original Gauss-Chebyshev quadrature and is detailed in Section A.1. Consequently, each Eq. (4.21) is multiplied by ω_j (obtained with Eq. (A.4)) and summing all these equations gives:

$$\frac{d}{dt} \sum_{j=1}^{N_{SP}^{1d}} \omega_j \hat{\mathbf{U}}(\xi_j) + \sum_{k=1}^{N_{FP}^{1d}} \hat{\mathbf{E}}_k \sum_{j=1}^{N_{SP}^{1d}} \omega_j \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (4.22)$$

Each component of $\widehat{\mathbf{U}}_j$ and each $\partial l_k^{FP}/\partial \xi$, for $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$, are polynomials of degree p so that:

$$\sum_{j=1}^{N_{SP}^{1d}} \omega_j \widehat{\mathbf{U}}(\xi_j) = \int_0^1 \widehat{\mathbf{U}} d\xi \quad (4.23)$$

$$\sum_{j=1}^{N_{SP}^{1d}} \omega_j \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = \int_0^1 \frac{\partial l_k^{FP}}{\partial \xi} d\xi = l_k^{FP}(1) - l_k^{FP}(0) = \begin{cases} 0 & \text{for } k = 2, N_{FP}^{1d} - 1 \\ 1 & \text{for } k = N_{FP}^{1d} \\ -1 & \text{for } k = 1 \end{cases} \quad (4.24)$$

Therefore, Eq. (4.22) becomes:

$$\frac{d}{dt} \int_0^1 \widehat{\mathbf{U}} d\xi = \widehat{\mathbf{E}}_1 - \widehat{\mathbf{E}}_{N_{FP}^{1d}} \quad (4.25)$$

demonstrating the conservativity of the SD method.

4.1.6. Gradient computation in 1D for a diffusion equation: original formulation

The discretization process described in paragraph 4.1.4 is sufficient if the equation solved is hyperbolic that is if \mathbf{E} is only a function of \mathbf{U} . However if the equation is not hyperbolic but is parabolic, \mathbf{E} is a function of either or both \mathbf{U} and $\nabla \mathbf{U}$. This kind of equation is often associated with diffusion processes so that it is also denoted by the term diffusion equation. For instance, the 1D heat diffusion equation:

$$\frac{\partial \mathbf{U}}{\partial t} = \frac{\partial}{\partial x} \left(\lambda \frac{\partial \mathbf{U}}{\partial x} \right) \quad (4.26)$$

is similar to Eq. (4.1) with $\mathbf{E} = -\lambda(\partial \mathbf{U}/\partial x)$ so that \mathbf{E} is a function of $(\partial \mathbf{U}/\partial x)$. The NSE introduced in Section 2.4.7 also have a parabolic part due to their viscous terms that it is why the discretization of parabolic equations using the SD method is described here. Thus if \mathbf{E} depends on $(\partial \mathbf{U}/\partial x)$, it means that $(\partial \mathbf{U}/\partial x)$ is needed at FP to compute \mathbf{E} at these points. The situation considered is represented in Figure 4.8 where, for clarity purposes, an element Ω_e is shown with its two interfaces between respectively Ω_{e-1} on its left and Ω_{e+1} on its right. Firstly, as for the hyperbolic 1D-equation,

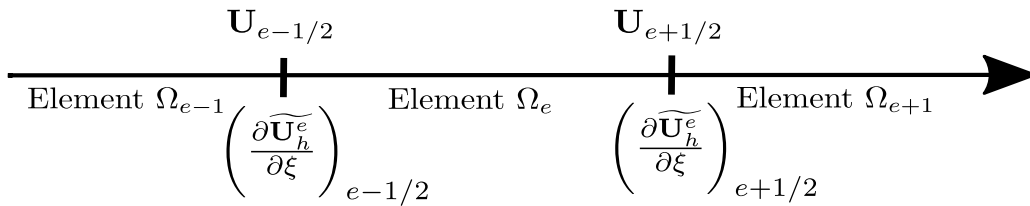


Figure 4.8. – Defining common values for an element Ω_e .

Eq. (4.18) is employed to evaluate \mathbf{U} at FP. At element interfaces $\xi = 0$ and $\xi = 1$, \mathbf{U} is discontinuous across elements and then interface (or common) values must be defined at these points. Usually, they are taken as the arithmetic average between the two states available at each interface:

$$\mathbf{U}_{e-1/2} = \frac{\mathbf{U}_h^{e-1}(1) + \mathbf{U}_h^e(0)}{2} \text{ and } \mathbf{U}_{e+1/2} = \frac{\mathbf{U}_h^e(1) + \mathbf{U}_h^{e+1}(0)}{2} \quad (4.27)$$

where superscripts $e-1$, e and $e+1$ were added here for clarity. Once common values have been computed at interface FP, \mathbf{U} is continuous across all elements and its values at FP are denoted by \mathbf{U}_k^C for $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$. Consequently, as for the flux polynomial defined by Eq. (4.19), a $(p+1)$ -degree

polynomial representation of a continuous solution vector can be constructed. This polynomial is differentiated along ξ direction and evaluated at SP to obtain:

$$\frac{\partial \mathbf{U}_h^e}{\partial \xi}(\xi_j) = \sum_{k=1}^{N_{FP}^{1d}} \mathbf{U}_k^C \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j), \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.28)$$

which is the gradient of \mathbf{U} along ξ evaluated at SP. The same process is repeated but for these gradient at SP:

1. A p -degree polynomial representation of the gradient is built from the values defined in Eq. (4.28) and is evaluated at FP:

$$\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi_k) = \sum_{j=1}^{N_{SP}^{1d}} \left(\frac{\partial \mathbf{U}_h^e}{\partial \xi} \right)_j l_j^{SP}(\xi_k), \text{ for } k \in \llbracket 1, N_{FP}^{1d} \rrbracket \quad (4.29)$$

2. Values of $\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)(\xi_k)$ are discontinuous at element interfaces. In the SD community, the usual approach is to take the arithmetic average, as in Eq. (4.27) for states, to define common gradient values at element interfaces:

$$\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} = \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^{e-1}}{\partial \xi}(1) + \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(0) \right] \text{ and } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} = \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(1) + \frac{\partial \widetilde{\mathbf{U}}_h^{e+1}}{\partial \xi}(0) \right] \quad (4.30)$$

3. Once common gradients have been computed, a $(p+1)$ -degree polynomial for the gradient of \mathbf{U} along ξ direction, continuous across all elements, is obtained denoted by $\left(\frac{\partial \widetilde{\widetilde{\mathbf{U}}}_h^e}{\partial \xi} \right)_k$.
4. Values of $\left(\frac{\partial \widetilde{\widetilde{\mathbf{U}}}_h^e}{\partial \xi} \right)_k$ are scaled by ξ_x to get the gradient of \mathbf{U} along x direction:

$$\left(\frac{\partial \mathbf{U}}{\partial x} \right)_k = \xi_x \left(\frac{\partial \widetilde{\widetilde{\mathbf{U}}}_h^e}{\partial \xi} \right)_k, \text{ for } k \in \llbracket 1, N_{FP}^{1d} \rrbracket \quad (4.31)$$

Finally, values of $(\partial \mathbf{U} / \partial x)_k$ are used to compute \mathbf{E}_k and Eq. (4.13) ends the process to obtain $\widehat{\mathbf{E}}_k$ at each FP $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$.

The methodology introduced in this paragraph to compute the gradient at FP is the original one proposed by Kopriva [193] and later used by Sun [47]. It is refer to as the average approach due to the averaging of both states (with Eq. (4.27)) and gradients (with Eq. (4.30)) at element interfaces. In paragraph 4.1.7, a new approach developed during this work will be introduced and is based on FR like scheme for gradient evaluation.

4.1.7. Gradient computation in 1D for a diffusion equation: SDLIFT formulation

The average approach described in paragraph 4.1.6 does not offer flexibility for the computation of common state and gradient at element interfaces and appeared to be very unstable when the mesh is coarse if a multi-species reacting gas is considered. Another methodology can be considered using principles from the FR discretization introduced by Huynh in 2009 in his paper for solving the diffusion equation with a FR scheme [194]. Thus, the reasoning shown here is strongly inspired from what is presented in this paper. The situation considered is still the one represented in Figure 4.8 where Eq. (4.26) has to be solved using a SD discretization.

Starting from Eq. (4.18), a $(p-1)$ -degree polynomial can be built for the gradient of \mathbf{U} along ξ direction:

$$\frac{\partial \mathbf{U}_h^e}{\partial \xi}(\xi) = \sum_{j=1}^{N_{SP}^{1d}} \frac{\hat{\mathbf{U}}_j^e}{|J|_j^e} \frac{\partial l_j^{SP}}{\partial \xi}(\xi), \text{ for } \xi \in [0, 1] \quad (4.32)$$

If $\mathbf{U}_{e-1/2}$ and $\mathbf{U}_{e+1/2}$ are the common state values (computed for instance with Eq. (4.27)), then $(\partial \mathbf{U}_h^e / \partial \xi)$ can be corrected into a p -degree polynomial as in the FR approach:

$$\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi) = \frac{\partial \mathbf{U}_h^e}{\partial \xi}(\xi) + (\mathbf{U}_{e-1/2} - \mathbf{U}_h^e(0)) \frac{\partial g_e^R}{\partial \xi}(\xi) + (\mathbf{U}_{e+1/2} - \mathbf{U}_h^e(1)) \frac{\partial g_e^L}{\partial \xi}(\xi) \quad (4.33)$$

where g_e^L (respectively g_e^R) is the left (respectively right) correction function at the right (respectively left) interface of Ω_e . The superscripts L and R refer to the sides of the interface and not on the element side as it was done in Huynh's original papers [57, 194]. For instance, g_e^L has superscript L because it is used for the left side of the interface $e+1/2$ and g_e^R is defined equivalently for right side of the interface in $e-1/2$. These correction functions are polynomials of degree $p+1$ that approximate zero in some sense [57, 194] and have the following constraints:

$$g_e^L(1) = 1 \text{ and } g_e^L(0) = 0 \quad (4.34)$$

$$g_e^R(1) = 0 \text{ and } g_e^R(0) = 1 \quad (4.35)$$

Thanks to these correction functions, $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)$ is of degree p but is discontinuous across elements. Consequently, a new correction step is done by introducing common gradient values at element interfaces (computed for instance with Eq. (4.30)) to have:

$$\frac{\partial \widetilde{\widetilde{\mathbf{U}}}_h^e}{\partial \xi}(\xi) = \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi) + \left[\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} - \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(0) \right] g_e^R(\xi) + \left[\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} - \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(1) \right] g_e^L(\xi) \quad (4.36)$$

Eq. (4.36) defines a $(p+1)$ -degree polynomial for the gradient of \mathbf{U} along ξ direction which is continuous across the computational domain and is evaluated at all FP to get $(\partial \mathbf{U} / \partial \xi)_k^C$ for $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$. Finally from the values of $(\partial \mathbf{U} / \partial \xi)_k^C$, Eq. (4.31) is employed to compute the gradient along x direction used for the fluxes at FP.

This new methodology is named the SDLIFT approach in this work since it uses the SD method coupled with lifting/correction functions as in the FR formalism. It offers the possibility to make different choices for the common states and gradients at element interfaces. Each of these choices is considered as a numerical diffusion scheme since it is a user choice on the way to set these common states and gradients. Three different diffusion schemes are considered in this work and are detailed respectively in paragraphs 4.1.7.2, 4.1.7.3 and 4.1.7.4. Before going through these diffusion schemes, some precision about correction functions employed in this work is provided.

4.1.7.1. Correction functions employed in this work

Correction functions have been introduced for the FR theory by Huynh [57, 194] where he proposed three different correction functions. In this work, two of these three correction functions were employed. There are defined on $\xi \in [0, 1]$ whereas Huynh worked with $\xi \in [-1, 1]$. Consequently, since in Huynh's framework g_e^L and g_e^R are symmetric about 0 for $\xi \in [-1, 1]$ [57, 194], in $\xi \in [0, 1]$ these two functions are symmetric about 0.5 meaning that:

$$g_e^R(\xi) = g_e^L(1 - \xi) \text{ for } \xi \in [0, 1] \quad (4.37)$$

Thus, it is possible to focus only on building g_e^R as a polynomial degree of $p + 1$ satisfying Eq. (4.34) and Eq. (4.35) and then use Eq. (4.37) to find g_e^L from g_e^R . That is why, for clarity in this paragraph, g_e^R will be denoted by g .

The first correction function used in this work is the Right Radau polynomial of degree $p + 1$, noted $R_{R,p+1}$, defined in Eq. (C.14) for an arbitrary degree $k \geq 1$. It turns out that with such correction function, the FR scheme is equivalent to the DG method [57] so that Huynh named it as g_{DG} :

$$g_{\text{DG}} = R_{R,p+1} \quad (4.38)$$

By construction, g_{DG} is of degree $p + 1$ and satisfies Eq. (4.35) because of Eq. (C.15).

The second correction function which was also investigated in this work is the one written as g_{Ga} by Huynh and defined as [57, 194]:

$$g_{\text{Ga}} = \frac{p+1}{2p+1} R_{R,p+1} + \frac{p}{2p+1} R_{R,p} \quad (4.39)$$

Thanks to the properties of Right Radau polynomials, g_{Ga} also verifies Eq. (4.35) and is of degree $p + 1$ since $R_{R,p+1}$ is of degree $p + 1$. It is named like this because its remaining p zeros (g_{Ga} is already zero at $\xi = 1$) on $[0, 1]$ are the p Gauss-Legendre points (see appendix C for more details) on $[0, 1]$. This can be demonstrated by expressing g_{Ga} as a function of P_p which is the Legendre polynomial of degree p (again see appendix C for more precision) on $[0, 1]$ here. Starting from Eq. (4.39), g_{Ga} becomes:

$$\begin{aligned} g_{\text{Ga}} &= \frac{1}{2p+1} \left[(p+1) \frac{(-1)^{p+1}}{2} (P_{p+1} - P_p) + p \frac{(-1)^p}{2} (P_p - P_{p-1}) \right] \\ \implies g_{\text{Ga}} &= \frac{(-1)^{p+1}}{2(2p+1)} [(p+1)(P_{p+1} - P_p) - p(P_p - P_{p-1})] \\ \implies g_{\text{Ga}} &= \frac{(-1)^{p+1}}{2(2p+1)} [(p+1)P_{p+1} - (2p+1)P_p + pP_{p-1}] \end{aligned} \quad (4.40)$$

According to the recurrence formula of Shifted (ξ is in $[0, 1]$ here) Legendre polynomials, shown in Eq. (C.11), for $k = p$:

$$\begin{aligned} (p+1)P_{p+1} &= (2p+1)(2\xi-1)P_p - pP_{p-1} \\ \implies (2p+1)(2\xi-1)P_p &= (p+1)P_{p+1} + pP_{p-1} \\ \implies (2p+1)(2\xi-2)P_p &= (p+1)P_{p+1} - (2p+1)P_p + pP_{p-1} \end{aligned} \quad (4.41)$$

by subtracting $(2p+1)P_p$ on both sides. Injecting Eq. (4.41) into Eq. (4.40) implies that:

$$g_{\text{Ga}} = \frac{(-1)^{p+1}}{2(2p+1)} (2p+1)(2\xi-2)P_p = (-1)^{p+1}(\xi-1)P_p \quad (4.42)$$

Therefore, Eq. (4.42) shows that g_{Ga} and $(\xi-1)P_p$ have the same zeros which are as expected $\xi = 1$ and the p roots of P_p corresponding to the p Gauss-Legendre points on $[0, 1]$. This result also highlights the fact that g_{Ga} vanishes at the p interior FP of the SD scheme since they correspond to the p Gauss-Legendre points on $[0, 1]$. This correction function has then an impact only at boundary FP $\xi = 0$ and $\xi = 1$.

The representations of g_{DG} and g_{Ga} for the cases $p = 2$ (degree 3 for correction functions) and $p = 4$ (degree 5 for correction functions) are shown in Figures 4.9a and 4.9b. The function g_{DG} is steeper than g_{Ga} . Huynh has shown that generally, a steeper correction function results in a scheme with smaller timesteps but is more accurate [57, 194]. The choice of the correction function is a user-defined parameter along with the diffusion scheme that will be detailed in the following

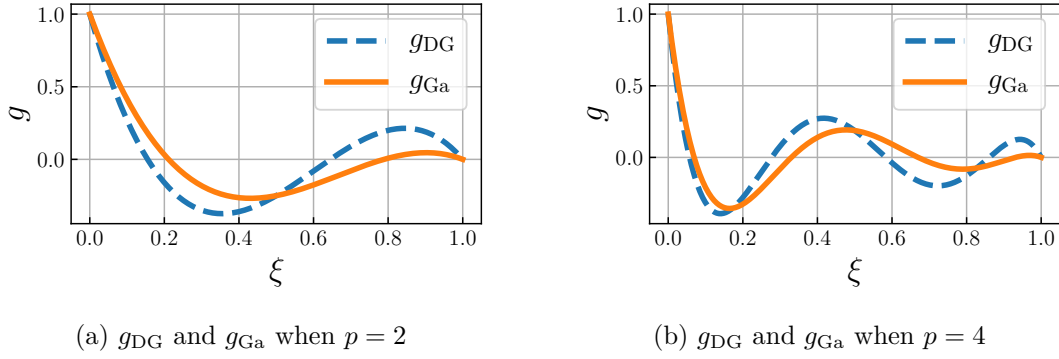


Figure 4.9. – Correction functions used in this work for the SDLIFT approach shown at two different polynomial degrees.

paragraphs for three different cases. In his article on diffusion schemes for the FR method, Huynh gave some recommendations about which correction function to use with a given diffusion scheme [194]. However, these recommendations were based on empirical results from the resolution of a simple diffusion equation so that it is difficult to generalize them when applied to the 3D NSE. Vincent *et al.* [59] have developed a methodology to find energy-stable FR scheme and have found some of stable Huynh's schemes with their method. They were able to identify mathematically the stability and the accuracy of a FR method for a given correction function. Nevertheless, there was no conclusion on which correction function to use: the more accurate ones have the lower timestep limits whereas the more stable dissipates too much and accuracy is lost. This trade-off between accuracy and stability is to be determined by the user.

4.1.7.2. Bassi and Rebay 1 diffusion scheme

The Bassi and Rebay 1 (BR1) scheme originally comes from DG formulation [195, 196], where common states and gradients in a given element Ω_e are set according respectively to Eq. (4.27) and Eq. (4.30). Thus, Eq. (4.36) using the common gradients defined in Eq. (4.30) has a five elements stencil as illustrated on Figure 4.10 since:

- For the left interface gradient in $e - 1/2$:

$$\begin{aligned} \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi) &\text{ depends on } \mathbf{U}_{e-1/2} \text{ and } \mathbf{U}_{e+1/2} \text{ then } \Omega_{e-1} \text{ and } \Omega_{e+1} \text{ are involved} \\ \Rightarrow \frac{\partial \widetilde{\mathbf{U}}_h^{e-1}}{\partial \xi}(\xi) &\text{ involves } \Omega_{e-2} \text{ and } \Omega_e \Rightarrow \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} \text{ involves } \Omega_{e-2}, \Omega_{e-1} \text{ and } \Omega_e. \end{aligned}$$

- For the right interface gradient in $e + 1/2$:

$$\text{Same reasoning ends up with } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} \text{ involving } \Omega_e, \Omega_{e+1} \text{ and } \Omega_{e+2}.$$

- **Conclusion:** $\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi)$ involves $\Omega_{e-2}, \Omega_{e-1}, \Omega_e, \Omega_{e+1}$ and Ω_{e+2} for the BR1 diffusion scheme.

Moreover, if g_e^L and g_e^R are zero at Gauss-Legendre points (g_{Ga} correction function is employed), this BR1 formulation is completely equivalent to the average approach described in paragraph 4.1.6.

4.1.7.3. Bassi and Rebay 2 diffusion scheme

As for the BR1 diffusion scheme, the Bassi and Rebay 2 (BR2) scheme also comes from DG formulation [195, 196], where common states are again computed with Eq. (4.27) but common gradients are

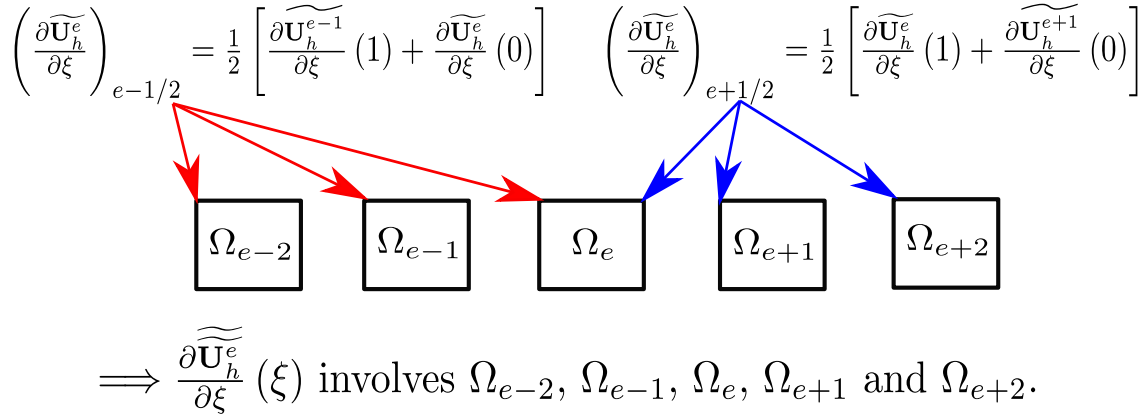


Figure 4.10. – Illustration of the stencil of the BR1 diffusion scheme.

set as:

$$\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} = \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^{e-1,R}}{\partial \xi}(1) + \frac{\partial \widetilde{\mathbf{U}}_h^{e,L}}{\partial \xi}(0) \right] \text{ and } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} = \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^{e,R}}{\partial \xi}(1) + \frac{\partial \widetilde{\mathbf{U}}_h^{e+1,L}}{\partial \xi}(0) \right] \quad (4.43)$$

where

$$\frac{\partial \widetilde{\mathbf{U}}_h^{e,L}}{\partial \xi}(0) \equiv \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(0) - (\mathbf{U}_{e+1/2} - \mathbf{U}_h^e(1)) \frac{\partial g_e^L}{\partial \xi}(0) = \frac{\partial \mathbf{U}_h^e}{\partial \xi}(0) + (\mathbf{U}_{e-1/2} - \mathbf{U}_h^e(0)) \frac{\partial g_e^R}{\partial \xi}(0) \quad (4.44)$$

$$\frac{\partial \widetilde{\mathbf{U}}_h^{e,R}}{\partial \xi}(1) \equiv \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(1) - (\mathbf{U}_{e-1/2} - \mathbf{U}_h^e(0)) \frac{\partial g_e^R}{\partial \xi}(1) = \frac{\partial \mathbf{U}_h^e}{\partial \xi}(1) + (\mathbf{U}_{e+1/2} - \mathbf{U}_h^e(1)) \frac{\partial g_e^L}{\partial \xi}(1) \quad (4.45)$$

In Eq. (4.44) and Eq. (4.45), the second equality is obtained from Eq. (4.33). Common states are then computed similarly to the BR1 scheme but thanks to Eq. (4.44) and Eq. (4.45), Eq. (4.36) using the common gradients defined in Eq. (4.43) has a three elements stencil as shown on Figure 4.11 because:

- For the left interface gradient in $e - 1/2$:

$$\frac{\partial \widetilde{\mathbf{U}}_h^{e-1,R}}{\partial \xi}(\xi) \text{ and } \frac{\partial \widetilde{\mathbf{U}}_h^{e,L}}{\partial \xi}(\xi) \text{ depend on } \mathbf{U}_{e-1/2} \text{ then } \Omega_{e-1} \text{ and } \Omega_e \text{ are involved}$$

$$\Rightarrow \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} \text{ involves only } \Omega_{e-1} \text{ and } \Omega_e.$$

- For the right interface gradient in $e + 1/2$:

$$\text{Same reasoning ends up with } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} \text{ involving only } \Omega_e \text{ and } \Omega_{e+1}.$$

- **Conclusion:** $\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi)$ involves Ω_{e-1}, Ω_e and Ω_{e+1} for the BR2 diffusion scheme.

$$\begin{aligned}
 \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} &= \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^{e-1,R}}{\partial \xi} (1) + \frac{\partial \widetilde{\mathbf{U}}_h^{e,L}}{\partial \xi} (0) \right] \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} = \frac{1}{2} \left[\frac{\partial \widetilde{\mathbf{U}}_h^{e,R}}{\partial \xi} (1) + \frac{\partial \widetilde{\mathbf{U}}_h^{e+1,L}}{\partial \xi} (0) \right] \\
 &\quad \begin{array}{c} \text{Red arrows from } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} \text{ to } \Omega_{e-1} \text{ and } \Omega_e \\ \text{Blue arrows from } \left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} \text{ to } \Omega_e \text{ and } \Omega_{e+1} \end{array} \\
 &\quad \begin{array}{c} \boxed{\Omega_{e-2}} \quad \boxed{\Omega_{e-1}} \quad \boxed{\Omega_e} \quad \boxed{\Omega_{e+1}} \quad \boxed{\Omega_{e+2}} \end{array} \\
 &\implies \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} (\xi) \text{ involves } \Omega_{e-1}, \Omega_e \text{ and } \Omega_{e+1}.
 \end{aligned}$$

Figure 4.11. – Illustration of the stencil of the BR2 diffusion scheme.

4.1.7.4. I-continuous diffusion scheme

The I-continuous (IC) scheme was introduced by Huynh [194] in the FR context. It computes the common state value at an element interface, for instance between Ω_e and Ω_{e+1} , such that:

$$\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} = \frac{\partial \widetilde{\mathbf{U}}_h^{e,R}}{\partial \xi} (1) = \frac{\partial \widetilde{\mathbf{U}}_h^{e+1,L}}{\partial \xi} (0) \quad (4.46)$$

$$\begin{aligned}
 &\iff \frac{\partial \mathbf{U}_h^e}{\partial \xi} (1) + (\mathbf{U}_{e+1/2} - \mathbf{U}_h^e (1)) \frac{\partial g_e^L}{\partial \xi} (1) = \frac{\partial \mathbf{U}_h^{e+1}}{\partial \xi} (0) + (\mathbf{U}_{e+1/2} - \mathbf{U}_h^{e+1} (0)) \frac{\partial g_{e+1}^R}{\partial \xi} (0) \\
 &\iff \mathbf{U}_{e+1/2} = \frac{\frac{\partial \mathbf{U}_h^{e+1}}{\partial \xi} (0) - \frac{\partial \mathbf{U}_h^e}{\partial \xi} (1) - \mathbf{U}_h^{e+1} (0) \frac{\partial g_{e+1}^R}{\partial \xi} (0) + \mathbf{U}_h^e (1) \frac{\partial g_e^L}{\partial \xi} (1)}{\frac{\partial g_e^L}{\partial \xi} (1) - \frac{\partial g_{e+1}^R}{\partial \xi} (0)} \quad (4.47)
 \end{aligned}$$

The same calculation holds for the interface between Ω_{e-1} and Ω_e . In practice, only g_e^L or g_e^R is stored in a CFD code since from Eq. (4.37):

$$\frac{\partial g_e^R}{\partial \xi} (\xi) = -\frac{\partial g_e^L}{\partial \xi} (1 - \xi) \quad (4.48)$$

Therefore if only right correction functions are stored, Eq. (4.47) can be recast into:

$$\mathbf{U}_{e+1/2} = \frac{\frac{\partial \mathbf{U}_h^e}{\partial \xi} (1) - \frac{\partial \mathbf{U}_h^{e+1}}{\partial \xi} (0) + \mathbf{U}_h^{e+1} (0) \frac{\partial g_{e+1}^R}{\partial \xi} (0) + \mathbf{U}_h^e (1) \frac{\partial g_e^R}{\partial \xi} (0)}{\frac{\partial g_e^R}{\partial \xi} (0) + \frac{\partial g_{e+1}^R}{\partial \xi} (0)} \quad (4.49)$$

Once common state has been computed with Eq. (4.49), common gradient is computed based on Eq. (4.46). As for the BR2 scheme, IC scheme has also a three elements stencil because common gradients are computed using values defined by Eq. (4.44) and Eq. (4.45). If the computation is done at constant polynomial order p , $g_e^R = g_{e+1}^R$ and Eq. (4.49) can be further simplified.

4.2. Formulation for hexahedral elements

The SD method for hexahedral elements, also called tensor-product elements, is an extension of the 1D formulation presented in Section 4.1. The objective of this section is to explain how the isoparametric transformation is applied when the equations considered are fully three-dimensional and also how SP and FP are located inside a standard hexahedral element. The physical and isoparametric

coordinates are now respectively denoted by $\mathbf{x} = (x, y, z)^T$ and $\boldsymbol{\xi} = (\xi, \eta, \zeta)^T$. It should be highlighted that the discretization of the SD method on triangles and tetrahedral elements, also called simplex elements, is not explained here since these elements were not considered in this work. Actually, simplex elements are treated differently than tensor-product ones. Indeed, the 1D formulation of Section 4.1 cannot be considered direction per direction on simplex elements, as it is the case for tensor-product ones which will be shown in the next paragraphs.

4.2.1. Isoparametric transformation in 3D

The NSE summarized in paragraph 2.4.7, can be recast into the following general form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{G}}{\partial z} = \mathbf{S}, \text{ for } (\mathbf{x}, t) \in \Omega \times [0, t_f] \quad (4.50)$$

where $\mathbf{U} = (\rho, \rho u, \rho v, \rho w, \rho E, \rho Y_1, \dots, \rho Y_{N_s})^T$ is the vector of conservative variables, $\mathbf{E} = \mathbf{E}_c + \mathbf{E}_d$, $\mathbf{F} = \mathbf{F}_c + \mathbf{F}_d$ and $\mathbf{G} = \mathbf{G}_c + \mathbf{G}_d$ are respectively the sum of convective and diffusive fluxes of \mathbf{U} along x , y and z directions and \mathbf{S} is a source term vector. They read as:

$$\begin{aligned} \mathbf{E}_c &= (\rho u, \quad \rho u^2 + P, \quad \rho v u, \quad \rho w u, \quad u(P + \rho E), \quad \rho u Y_1, \quad \dots, \quad \rho u Y_{N_s})^T \\ \mathbf{F}_c &= (\rho v, \quad \rho u v, \quad \rho v^2 + P, \quad \rho w v, \quad v(P + \rho E), \quad \rho v Y_1, \quad \dots, \quad \rho v Y_{N_s})^T \\ \mathbf{G}_c &= (\rho w, \quad \rho u w, \quad \rho v w, \quad \rho w^2 + P, \quad w(P + \rho E), \quad \rho w Y_1, \quad \dots, \quad \rho w Y_{N_s})^T \\ \mathbf{E}_d &= (0, \quad -\tau_{11}, \quad -\tau_{21}, \quad -\tau_{31}, \quad \partial_x q_x - u\tau_{11} - v\tau_{12} - w\tau_{13}, \quad -\partial_x M_{1x}, \quad \dots, \quad -\partial_x M_{N_s x})^T \\ \mathbf{F}_d &= (0, \quad -\tau_{12}, \quad -\tau_{22}, \quad -\tau_{32}, \quad \partial_y q_y - u\tau_{21} - v\tau_{22} - w\tau_{23}, \quad -\partial_y M_{1y}, \quad \dots, \quad -\partial_y M_{N_s y})^T \\ \mathbf{G}_d &= (0, \quad -\tau_{13}, \quad -\tau_{23}, \quad -\tau_{33}, \quad \partial_z q_z - u\tau_{31} - v\tau_{32} - w\tau_{33}, \quad -\partial_z M_{1z}, \quad \dots, \quad -\partial_z M_{N_s z})^T \\ \mathbf{S} &= (0, \quad 0, \quad 0, \quad 0, \quad \dot{\omega}_T, \quad \dot{\omega}_1, \quad \dots, \quad \dot{\omega}_{N_s})^T \end{aligned} \quad (4.51)$$

The 3D computational domain Ω is now divided into N_e non-overlapping hexahedral elements inside which Eq. (4.50) is to be solved. Each element Ω_e of Ω will be transformed into a standard hexahedron $\mathcal{H} = \{(\xi, \eta, \zeta), 0 \leq \xi, \eta, \zeta \leq 1\}$ following the inverse of the transformation defined in Eq. (4.52) [47]:

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{i=1}^{N_v} M_i(\boldsymbol{\xi}) \mathbf{x}_i^e \quad (4.52)$$

where $\mathbf{x}_i^e = (x_i^e, y_i^e, z_i^e)$ are the Cartesian coordinates in the physical domain of vertex i in Ω_e and $M_i(\boldsymbol{\xi})$ are still the shape functions. As in the 1D case, Ω_e can be linear ($N_v = 8$), quadratic ($N_v = 27$) or a even higher order hexahedral element. The transformation and its inverse from a linear standard hexahedron in the isoparametric domain into a linear hexahedron in the physical domain is represented in Figure 4.12 where $\boldsymbol{\xi}_i = (\xi_i, \eta_i, \zeta_i)$ are the coordinates of vertex i in the standard hexahedron. The

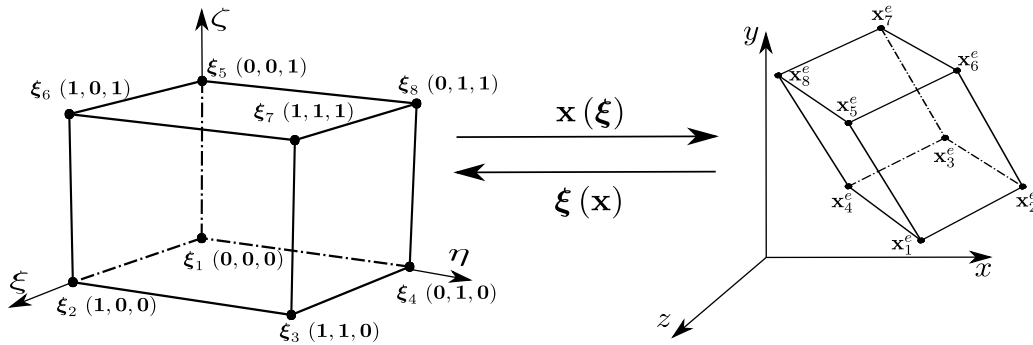


Figure 4.12. – Isoparametric transformation between a linear standard hexahedron and a linear hexahedron.

advantage of using high-order elements comes when complex geometries are considered since they will better describe curved boundaries or mesh singularities. In 3D for a hexahedral element, the shape functions are the product of the one-dimensional shape functions:

$$M_i(\xi, \eta, \zeta) = l_i^{VT}(\xi) l_i^{VT}(\eta) l_i^{VT}(\zeta), \text{ for } i \in \llbracket 1, N_v \rrbracket \quad (4.53)$$

where l_i^{VT} is the 1D Lagrange polynomial defined in Eq. (4.6). This isoparametric transformation from the reference domain ξ to the physical domain \mathbf{x} is characterized by a Jacobian matrix J along with its inverse (assuming a non-singular transformation) representing the reverse transformation:

$$J = \begin{pmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{pmatrix} \quad \text{and} \quad J^{-1} = \begin{pmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{pmatrix} \quad (4.54)$$

J can be determined explicitly thanks to Eq. (4.52) which can be derived with respect to ξ, η and ζ :

$$\frac{\partial x}{\partial(\xi, \eta, \zeta)} \equiv (x_\xi, x_\eta, x_\zeta)^T = \sum_{i=1}^{N_v} \frac{\partial M_i}{\partial(\xi, \eta, \zeta)} x_i^e \quad (4.55)$$

$$\frac{\partial y}{\partial(\xi, \eta, \zeta)} \equiv (y_\xi, y_\eta, y_\zeta)^T = \sum_{i=1}^{N_v} \frac{\partial M_i}{\partial(\xi, \eta, \zeta)} y_i^e \quad (4.56)$$

$$\frac{\partial z}{\partial(\xi, \eta, \zeta)} \equiv (z_\xi, z_\eta, z_\zeta)^T = \sum_{i=1}^{N_v} \frac{\partial M_i}{\partial(\xi, \eta, \zeta)} z_i^e \quad (4.57)$$

Thanks to Eq. (4.53), derivatives of shape functions with respect to reference coordinates can be analytically expressed:

$$\frac{\partial M_i}{\partial \xi}(\xi, \eta, \zeta) = \frac{\partial l_i^{VT}}{\partial \xi}(\xi) l_i^{VT}(\eta) l_i^{VT}(\zeta), \text{ for } i \in \llbracket 1, N_v \rrbracket \quad (4.58)$$

$$\frac{\partial M_i}{\partial \eta}(\xi, \eta, \zeta) = l_i^{VT}(\xi) \frac{\partial l_i^{VT}}{\partial \eta}(\eta) l_i^{VT}(\zeta), \text{ for } i \in \llbracket 1, N_v \rrbracket \quad (4.59)$$

$$\frac{\partial M_i}{\partial \zeta}(\xi, \eta, \zeta) = l_i^{VT}(\xi) l_i^{VT}(\eta) \frac{\partial l_i^{VT}}{\partial \zeta}(\zeta), \text{ for } i \in \llbracket 1, N_v \rrbracket \quad (4.60)$$

Therefore, J^{-1} and $|J^{-1}|$ will be given by [4]:

$$J^{-1} = \frac{[com(J)]^T}{|J|} = \frac{1}{|J|} \begin{pmatrix} y_\eta z_\zeta - z_\eta y_\zeta & z_\eta x_\zeta - x_\eta z_\zeta & x_\eta y_\zeta - y_\eta x_\zeta \\ z_\xi y_\zeta - y_\xi z_\zeta & x_\xi z_\zeta - z_\xi x_\zeta & y_\xi x_\zeta - x_\xi y_\zeta \\ y_\xi z_\eta - z_\xi y_\eta & z_\xi x_\eta - x_\xi z_\eta & x_\xi y_\eta - y_\xi x_\eta \end{pmatrix} \quad (4.61)$$

$$|J^{-1}| = \frac{1}{|J|} = \frac{1}{x_\xi(y_\eta z_\zeta - z_\eta y_\zeta) - y_\xi(x_\eta z_\zeta - x_\zeta z_\eta) + z_\xi(x_\eta y_\zeta - x_\zeta y_\eta)} \quad (4.62)$$

where $com(J)$ is the comatrix of J . Combining Eq. (4.61) with Eq. (4.54) gives the expressions of J^{-1} components as a function of J components:

$$\begin{aligned} \xi_x |J| &= (y_\eta z_\zeta - z_\eta y_\zeta) \\ \xi_y |J| &= (z_\eta x_\zeta - x_\eta z_\zeta) \\ \xi_z |J| &= (x_\eta y_\zeta - y_\eta x_\zeta) \\ \eta_x |J| &= (z_\xi y_\zeta - y_\xi z_\zeta) \\ \eta_y |J| &= (x_\xi z_\zeta - z_\xi x_\zeta) \\ \eta_z |J| &= (y_\xi x_\zeta - x_\xi y_\zeta) \\ \zeta_x |J| &= (y_\xi z_\eta - z_\xi y_\eta) \\ \zeta_y |J| &= (z_\xi x_\eta - x_\xi z_\eta) \\ \zeta_z |J| &= (x_\xi y_\eta - y_\xi x_\eta) \end{aligned} \quad (4.63)$$

Jacobian matrices, and then their determinants, depend on the location of the points where they are computed due to the interpolation process on geometry using shape functions. Therefore, there are Jacobian matrices for SP and for FP.

As in the 1D case, Eq. (4.50) is solved in the isoparametric domain using the derivation chain rule on $\partial \mathbf{E}/\partial x$, $\partial \mathbf{F}/\partial y$ and $\partial \mathbf{G}/\partial z$:

$$\frac{\partial \mathbf{U}}{\partial t} + \left(\xi_x \frac{\partial \mathbf{E}}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}}{\partial \xi} \right) + \left(\eta_x \frac{\partial \mathbf{E}}{\partial \eta} + \eta_y \frac{\partial \mathbf{F}}{\partial \eta} + \eta_z \frac{\partial \mathbf{G}}{\partial \eta} \right) + \left(\zeta_x \frac{\partial \mathbf{E}}{\partial \zeta} + \zeta_y \frac{\partial \mathbf{F}}{\partial \zeta} + \zeta_z \frac{\partial \mathbf{G}}{\partial \zeta} \right) = \mathbf{S} \quad (4.64)$$

Eq. (4.64) is not in divergent form which is not something useful in a CFD context. However, Vinokur [197] and Viviand [198] have shown that any equations of the form of Eq. (4.50), to which a transformation was applied, can be put back into a conservative form in the new transformed domain [4]. To do so, Eq. (4.64) is first multiplied by $|J|$ and is then rearranged into a conservative form in (ξ, η, ζ) space by adding and subtracting terms:

$$\begin{aligned} |J| \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial}{\partial \xi} [|J| (\xi_x \mathbf{E} + \xi_y \mathbf{F} + \xi_z \mathbf{G})] + \frac{\partial}{\partial \eta} [|J| (\eta_x \mathbf{E} + \eta_y \mathbf{F} + \eta_z \mathbf{G})] + \frac{\partial}{\partial \zeta} [|J| (\zeta_x \mathbf{E} + \zeta_y \mathbf{F} + \zeta_z \mathbf{G})] \\ - \mathbf{E} \left[\frac{\partial}{\partial \xi} (\xi_x |J|) + \frac{\partial}{\partial \eta} (\eta_x |J|) + \frac{\partial}{\partial \zeta} (\zeta_x |J|) \right] \\ - \mathbf{F} \left[\frac{\partial}{\partial \xi} (\xi_y |J|) + \frac{\partial}{\partial \eta} (\eta_y |J|) + \frac{\partial}{\partial \zeta} (\zeta_y |J|) \right] \\ - \mathbf{G} \left[\frac{\partial}{\partial \xi} (\xi_z |J|) + \frac{\partial}{\partial \eta} (\eta_z |J|) + \frac{\partial}{\partial \zeta} (\zeta_z |J|) \right] = |J| \mathbf{S} \end{aligned} \quad (4.65)$$

The last three terms in brackets are equal to zero and can be dropped. This can be checked by substituting the expressions of J components given by Eq. (4.63) into these three terms and using Schwarz's theorem to permute partial derivatives all the terms canceled. Thus, introducing the flux vectors in the isoparametric domain:

$$\hat{\mathbf{E}} = |J| (\xi_x \mathbf{E} + \xi_y \mathbf{F} + \xi_z \mathbf{G}), \quad \hat{\mathbf{F}} = |J| (\eta_x \mathbf{E} + \eta_y \mathbf{F} + \eta_z \mathbf{G}), \quad \hat{\mathbf{G}} = |J| (\zeta_x \mathbf{E} + \zeta_y \mathbf{F} + \zeta_z \mathbf{G}) \quad (4.66)$$

and substitute them into Eq. (4.65), the equation in conservative form in the isoparametric space is:

$$\frac{\partial \hat{\mathbf{U}}}{\partial t} + \frac{\partial \hat{\mathbf{E}}}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}}{\partial \zeta} = \hat{\mathbf{S}}, \quad \text{for } (\xi, t) \in \mathcal{H} \times [0, t_f] \quad (4.67)$$

with still $\hat{\mathbf{U}}$ given by Eq. (4.12) and $\hat{\mathbf{S}} = |J| \mathbf{S}$. Again as in the 1D case, $|J|$ is assumed to be independent of time. Note that Eq. (4.66) can be written in matrix vector product using $|J|$ and J^{-1} :

$$\hat{\mathcal{F}} = |J| J^{-1} \mathcal{F} \quad \text{where} \quad \hat{\mathcal{F}} = \begin{pmatrix} \hat{\mathbf{E}} \\ \hat{\mathbf{F}} \\ \hat{\mathbf{G}} \end{pmatrix} \quad \text{and} \quad \mathcal{F} = \begin{pmatrix} \mathbf{E} \\ \mathbf{F} \\ \mathbf{G} \end{pmatrix} \quad (4.68)$$

which allows to write Eq. (4.50) and Eq. (4.67) as:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{G}}{\partial z} = \mathbf{S} \iff \frac{\partial \mathbf{U}}{\partial t} + \nabla_{\mathbf{x}} \cdot \mathcal{F} = \mathbf{S} \quad (4.69)$$

$$\frac{\partial \hat{\mathbf{U}}}{\partial t} + \frac{\partial \hat{\mathbf{E}}}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}}{\partial \zeta} = \hat{\mathbf{S}} \iff \frac{\partial \hat{\mathbf{U}}}{\partial t} + \nabla_{\xi} \cdot \hat{\mathcal{F}} = \hat{\mathbf{S}} \quad (4.70)$$

where $\nabla_{\mathbf{x}}$ and ∇_{ξ} are the divergence operators respectively in \mathbf{x} and ξ spaces. Variables \mathcal{F} and $\hat{\mathcal{F}}$ are the flux tensors respectively in the physical and isoparametric domains. Since, $|J| \times \text{Eq. (4.64)}$ and Eq. (4.65) are equal, it can be written that:

$$\nabla_{\xi} \cdot \hat{\mathcal{F}} = |J| \times \nabla_{\mathbf{x}} \cdot \mathcal{F} \quad (4.71)$$

4.2.2. Solution points and flux points in 3D

4.2.2.1. Solutions points in 3D

For 3D hexahedral grids, the SP are set direction per direction by repeating the same 1D process described in Section 4.1. It means that for an element of degree p , the number of SP inside this element is:

$$N_{SP}^d = (p+1)^d \quad (4.72)$$

where d stands for the space dimension ($d = 3$ for hexahedral elements). A SP $j \in \llbracket 1, N_{SP}^d \rrbracket$ has then three coordinates: $\xi_j = (\xi_j, \eta_j, \zeta_j)^T$. Each of these coordinates are still given by the Gauss-Chebyshev points in each direction in the isoparametric space:

$$(\xi_j, \eta_j, \zeta_j)^T = (\xi_{j_1}, \xi_{j_2}, \xi_{j_3})^T = \xi_{j_1, j_2, j_3}, \quad (j_1, j_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket^d \quad (4.73)$$

where ξ_{j_1} , ξ_{j_2} and ξ_{j_3} are obtained using Eq. (4.14). If all elements are at the same degree p , the number of DOF inside the computational domain is given by:

$$\text{DOF} = N_e \times N_{SP}^d \quad (4.74)$$

4.2.2.2. Flux points in 3D

FP locations in 3D hexahedral grids are linked to the fact that the flux polynomial must be of degree $p+1$ along each direction ξ , η and ζ . To do so, three sets of FP, one per direction ξ , η and ζ , are introduced. Coordinates $\xi_k = (\xi_k, \eta_k, \zeta_k)^T$ of a given FP $k \in \llbracket 1, N_{FP}^d \rrbracket$ (N_{FP}^d is defined in Eq. (4.76)) depends on which direction the FP is:

- **For ξ -FP:** $(\xi_k, \eta_k, \zeta_k)^T = (\xi_{k_1}, \eta_{j_2}, \zeta_{j_3})^T$, $(k_1, j_2, j_3) \in \llbracket 1, N_{FP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket$
- **For η -FP:** $(\xi_k, \eta_k, \zeta_k)^T = (\xi_{j_1}, \eta_{k_2}, \zeta_{j_3})^T$, $(j_1, k_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{FP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket$
- **For ζ -FP:** $(\xi_k, \eta_k, \zeta_k)^T = (\xi_{j_1}, \eta_{j_2}, \zeta_{k_3})^T$, $(j_1, j_2, k_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{FP}^{1d} \rrbracket$

where ξ_{k_1} (respectively η_{k_2} and ζ_{k_3}) are the Gauss-Legendre points along ξ (respectively η and ζ) direction plus the two ending points. Consequently in space dimension d , the number of FP along a given direction noted $N_{FP}^{dir, d}$ and the total number of FP noted N_{FP}^d inside a hexahedral elements are given by:

$$N_{FP}^{dir, d} = (p+2)(p+1)^{d-1} \quad (4.75)$$

$$N_{FP}^d = d \times N_{FP}^{dir, d} = d(p+2)(p+1)^{d-1} \quad (4.76)$$

SP and FP in the isoparametric domain \mathcal{H} are represented in Figure 4.13 for the $p = 1$ case.

4.2.3. General principle in 3D

The extension of the algorithm presented in paragraph 4.1.4 for a 1D hyperbolic equation is done here for 3D conservative laws written in the isoparametric domain such as Eq. (4.67). At a given time instant, the solution vector $\hat{\mathbf{U}}$ contains the conservative variables in the isoparametric domain stored at each SP $(\xi_{j_1}, \xi_{j_2}, \xi_{j_3})^T$ inside \mathcal{H} noted $\hat{\mathbf{U}}_{(j_1, j_2, j_3)}$. These values at SP are used to construct, in each direction ξ , η and ζ , a p -degree polynomial representation of a continuous solution \mathbf{U}_h across \mathcal{H} as the tensor product of the one-dimensional Lagrange polynomials of degree p along ξ , η and ζ directions:

$$\mathbf{U}_h(\xi, \eta, \zeta) = \sum_{j_1=1}^{N_{SP}^{1d}} \sum_{j_2=1}^{N_{SP}^{1d}} \sum_{j_3=1}^{N_{SP}^{1d}} \frac{\hat{\mathbf{U}}_{(j_1, j_2, j_3)}}{|J|_{(j_1, j_2, j_3)}} l_{j_1}^{SP}(\xi) l_{j_2}^{SP}(\eta) l_{j_3}^{SP}(\zeta), \quad \text{for } (\xi, \eta, \zeta) \in [0, 1]^3 \quad (4.77)$$

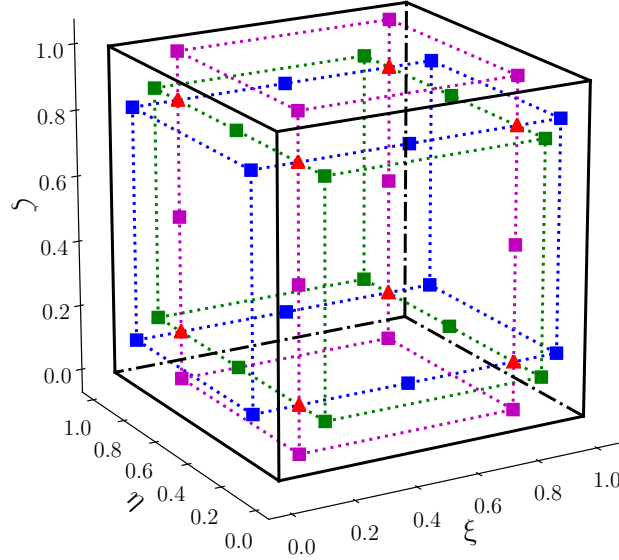


Figure 4.13. – Locations of SP (\blacktriangle), ξ -FP (\blacksquare), η -FP (\blacksquare) and ζ -FP (\blacksquare) for $p = 1$ in isoparametric domain \mathcal{H} .

where $l_{j_1}^{SP}$ (respectively $l_{j_2}^{SP}$ and $l_{j_3}^{SP}$) is the 1D Lagrange polynomials built at 1D SP ξ_{j_1} (respectively ξ_{j_2} and ξ_{j_3}). Eq. (4.77) is used to evaluate conservative variables at FP along ξ , η and ζ directions respectively noted $\mathbf{U}_{(k_1, j_2, j_3)}$, $\mathbf{U}_{(j_1, k_2, j_3)}$ and $\mathbf{U}_{(j_1, j_2, k_3)}$. The flux evaluation at FP depends on whether the FP is an internal or an interface FP and also if the flux to compute is a convective or a diffusive flux. Consequently, there are four possible cases:

- **Convective fluxes at internal FP:** values of \mathbf{E}_c , \mathbf{F}_c and \mathbf{G}_c are obtained at ξ -FP (respectively η -FP and ζ -FP) from the values of $\mathbf{U}_{(k_1, j_2, j_3)}$ (respectively $\mathbf{U}_{(j_1, k_2, j_3)}$ and $\mathbf{U}_{(j_1, j_2, k_3)}$) since convective fluxes are only a function of \mathbf{U} .
- **Diffusive fluxes at internal FP:** as introduced in paragraphs 4.1.6 and 4.1.7, the diffusive fluxes can be both a function of \mathbf{U} and $\nabla \mathbf{U}$. In that case, gradients at FP along ξ , η and ζ directions, respectively denoted by $\nabla \mathbf{U}_{(k_1, j_2, j_3)}$, $\nabla \mathbf{U}_{(j_1, k_2, j_3)}$ and $\nabla \mathbf{U}_{(j_1, j_2, k_3)}$, are computed at FP. Then, values of \mathbf{E}_d , \mathbf{F}_d and \mathbf{G}_d are obtained at ξ -FP (respectively η -FP and ζ -FP) from the values of $(\mathbf{U}, \nabla \mathbf{U})_{(k_1, j_2, j_3)}$ (respectively $(\mathbf{U}, \nabla \mathbf{U})_{(j_1, k_2, j_3)}$ and $(\mathbf{U}, \nabla \mathbf{U})_{(j_1, j_2, k_3)}$). The methodology to compute the gradients used in this work is detailed in Section 4.4.
- **Convective fluxes at interface FP:** as in 1D, conservative variables are discontinuous at these FP leading to a discontinuity of convective fluxes. A Riemann solver is employed to compute interface fluxes in the isoparametric domain namely $\hat{\mathbf{E}}_c^I$, $\hat{\mathbf{F}}_c^I$ and $\hat{\mathbf{G}}_c^I$. A description of the Riemann solvers and the one used in this work is explained in Section 4.3.
- **Diffusive fluxes at interface FP:** they are also discontinuous at interface FP since both conservative variables and their gradients are discontinuous at these locations. A diffusion scheme is considered to get unique diffusive flux values in the isoparametric domain at interface FP namely $\hat{\mathbf{E}}_d^I$, $\hat{\mathbf{F}}_d^I$ and $\hat{\mathbf{G}}_d^I$. A description of the diffusion schemes employed in this work is also found in Section 4.4.

At internal FP along ξ , η and ζ directions, convective and diffusive fluxes in the physical domain are added together to get \mathbf{E} , \mathbf{F} and \mathbf{G} on these locations. The fluxes in the isoparametric domain $\hat{\mathbf{E}}_{(k_1, j_2, j_3)}$, $\hat{\mathbf{F}}_{(j_1, k_2, j_3)}$ and $\hat{\mathbf{G}}_{(j_1, j_2, k_3)}$ are then deduced from Eq. (4.68). Once internal and interface fluxes have been computed, continuous fluxes $\hat{\mathbf{E}}_h$, $\hat{\mathbf{F}}_h$ and $\hat{\mathbf{G}}_h$ across \mathcal{H} of degree $(p+1)$ respectively along ξ , η and ζ directions can be constructed from flux values at FP and 1D Lagrange polynomials:

$$\hat{\mathbf{E}}_h(\xi, \eta, \zeta) = \sum_{k_1=1}^{N_{FP}^{1d}} \sum_{j_2=1}^{N_{SP}^{1d}} \sum_{j_3=1}^{N_{SP}^{1d}} \hat{\mathbf{E}}_{(k_1, j_2, j_3)} l_{k_1}^{FP}(\xi) l_{j_2}^{SP}(\eta) l_{j_3}^{SP}(\zeta), \text{ for } (\xi, \eta, \zeta) \in [0, 1]^3 \quad (4.78)$$

$$\hat{\mathbf{F}}_h(\xi, \eta, \zeta) = \sum_{j_1=1}^{N_{SP}^{1d}} \sum_{k_2=1}^{N_{FP}^{1d}} \sum_{j_3=1}^{N_{SP}^{1d}} \hat{\mathbf{F}}_{(j_1, k_2, j_3)} l_{j_1}^{SP}(\xi) l_{k_2}^{FP}(\eta) l_{j_3}^{SP}(\zeta), \text{ for } (\xi, \eta, \zeta) \in [0, 1]^3 \quad (4.79)$$

$$\hat{\mathbf{G}}_h(\xi, \eta, \zeta) = \sum_{j_1=1}^{N_{SP}^{1d}} \sum_{j_2=1}^{N_{SP}^{1d}} \sum_{k_3=1}^{N_{FP}^{1d}} \hat{\mathbf{G}}_{(j_1, j_2, k_3)} l_{j_1}^{SP}(\xi) l_{j_2}^{SP}(\eta) l_{k_3}^{FP}(\zeta), \text{ for } (\xi, \eta, \zeta) \in [0, 1]^3 \quad (4.80)$$

Eq. (4.78), Eq. (4.79) and Eq. (4.80) are then differentiated respectively along ξ , η and ζ directions and evaluated at SP to get:

$$\left(\frac{\partial \hat{\mathbf{E}}_h}{\partial \xi} \right)_{(j_1, j_2, j_3)} = \sum_{k_1=1}^{N_{FP}^{1d}} \hat{\mathbf{E}}_{(k_1, j_2, j_3)} \frac{\partial l_{k_1}^{FP}}{\partial \xi}(\xi_{j_1}), \quad (j_1, j_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket^d \quad (4.81)$$

$$\left(\frac{\partial \hat{\mathbf{F}}_h}{\partial \eta} \right)_{(j_1, j_2, j_3)} = \sum_{k_2=1}^{N_{FP}^{1d}} \hat{\mathbf{F}}_{(j_1, k_2, j_3)} \frac{\partial l_{k_2}^{FP}}{\partial \eta}(\eta_{j_2}), \quad (j_1, j_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket^d \quad (4.82)$$

$$\left(\frac{\partial \hat{\mathbf{G}}_h}{\partial \zeta} \right)_{(j_1, j_2, j_3)} = \sum_{k_3=1}^{N_{FP}^{1d}} \hat{\mathbf{G}}_{(j_1, j_2, k_3)} \frac{\partial l_{k_3}^{FP}}{\partial \zeta}(\zeta_{j_3}), \quad (j_1, j_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket^d \quad (4.83)$$

where $l_{jm}^{SP}(\xi_{jn}) = \delta_{mn}$, with δ_{mn} being the Kronecker symbol between m and n , was abundantly used to get rid of the triple summations in Eq. (4.78), Eq. (4.79) and Eq. (4.80). Finally, the semi-discrete (because time has not been discretized yet) formulation of Eq. (4.67) can be written and integrated in time using any explicit temporal scheme:

$$\left(\frac{d\hat{\mathbf{U}}}{dt} \right)_{(j_1, j_2, j_3)} = - \left[\left(\frac{\partial \hat{\mathbf{E}}_h}{\partial \xi} \right)_{(j_1, j_2, j_3)} + \left(\frac{\partial \hat{\mathbf{F}}_h}{\partial \eta} \right)_{(j_1, j_2, j_3)} + \left(\frac{\partial \hat{\mathbf{G}}_h}{\partial \zeta} \right)_{(j_1, j_2, j_3)} \right] + \hat{\mathbf{S}}_{(j_1, j_2, j_3)}, \quad (j_1, j_2, j_3) \in \llbracket 1, N_{SP}^{1d} \rrbracket^d \quad (4.84)$$

The temporal discretization of Eq. (4.84) will be detailed in Section 4.6.

4.2.4. Practical implementation

According to Eq. (4.81), Eq. (4.82) and Eq. (4.83), values of $\hat{\mathbf{E}}$, $\hat{\mathbf{F}}$ and $\hat{\mathbf{G}}$ are only needed respectively at ξ -FP, η -FP and ζ -FP. For instance, there is no need to compute $\hat{\mathbf{F}}$ and $\hat{\mathbf{G}}$ for FP along ξ -direction since they will not be used to derive $\hat{\mathbf{E}}$ along ξ -direction with Eq. (4.81). Consequently for ξ -FP, only $|J|\xi_x$, $|J|\xi_y$ and $|J|\xi_z$ are stored at these points to compute $\hat{\mathbf{E}}$ from \mathbf{E} , \mathbf{F} and \mathbf{G} using the first component of Eq. (4.68). This is exactly what it is done for internal FP but for interface FP some precision should be made. At interface FP, a Riemann solver is employed for convective fluxes (subscript c) and a diffusion scheme is used for diffusive fluxes (subscript d). The Riemann solver and the diffusion scheme give respectively the physical convective and diffusive flux values in the physical normal direction of the interface FP defined as:

$$\mathcal{F}_{c/d, n} = \mathcal{F}_{c/d} \cdot \mathbf{n}_{FP}^u \quad (4.85)$$

where \mathbf{n}_{FP}^u is the unit physical normal vector at a given FP introduced in Section A.2. However, for instance at interface FP along ξ -direction, it is the value of $\hat{\mathbf{E}}_{c/d}^I$ that is actually needed. It shows that a link between $\mathcal{F}_{c/d} \cdot \mathbf{n}_{FP}^u$ and $\hat{\mathbf{E}}_{c/d}^I$ has to be found. By definition, $\hat{\mathbf{E}}_{c/d}^I$ can be viewed as:

$$\hat{\mathbf{E}}_{c/d}^I \equiv \hat{\mathcal{F}}_{c/d} \cdot \hat{\mathbf{n}}_{FP} = |J|_{FP} \times \mathcal{F}_{c/d} \cdot \mathbf{n}_{FP} = |J|_{FP} \times \mathcal{F}_{c/d} \cdot \mathbf{n}_{FP}^u \times A_{FP} \quad (4.86)$$

with $\hat{\mathbf{n}}_{FP} = (1, 0, 0)^T$ and \mathbf{n}_{FP} , \mathbf{n}_{FP}^u and A_{FP} given in paragraph A.2.2 for an interface FP along ξ -direction. Using these expressions of \mathbf{n}_{FP}^u and A_{FP} , Eq. (4.86) becomes:

$$\hat{\mathbf{E}}_{c/d}^I = |J|_{FP} \times \mathcal{F}_{c/d} \cdot \frac{1}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}} \begin{pmatrix} \xi_x \\ \xi_y \\ \xi_z \end{pmatrix} \times \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} = |J| \left(\xi_x \mathbf{E}_{c/d} + \xi_y \mathbf{F}_{c/d} + \xi_z \mathbf{G}_{c/d} \right) \quad (4.87)$$

Eq. (4.87) is fully consistent with the first component of Eq. (4.68) used for internal FP along ξ -direction. The same reasoning holds for interface FP along η and ζ directions.

4.3. Riemann solvers for convective fluxes at interfaces

As mentioned in paragraphs 4.1.4 and 4.2.3, during the SD discretization process a Riemann solver is needed to determine flux values at FP interfaces between two elements. Actually, the Riemann solver gives the term $\mathcal{F}_c \cdot \mathbf{n}_{FP}^u$ introduced in Eq. (4.86). The objective of this section is to give a brief explanation on what is actually a Riemann solver and show an example of one used during this work.

4.3.1. The Riemann problem and its solvers using a 1D equation

In this paragraph as only the 1D case is considered, $\mathcal{F}_c = \mathbf{E}_c$ and $\mathbf{n}_{FP}^u = n_{FP}^u = 1$ so that the physical normal flux at an interface is simply:

$$\mathcal{F}_c \cdot \mathbf{n}_{FP}^u = \mathbf{E}_c \quad (4.88)$$

4.3.1.1. Presentation of the problem

The Riemann problem can be illustrated by considering a 1D hyperbolic equation as Eq. (4.1) for a convective flux $\mathbf{E} = \mathbf{E}_c$ with the following initial condition at an interface $x = 0$ on a 1D domain:

$$\mathbf{U}(x, 0) = \mathbf{U}^L (x < 0) \text{ and } \mathbf{U}(x, 0) = \mathbf{U}^R (x > 0) \quad (4.89)$$

This situation is represented in Figure 4.14 where obviously $\mathbf{U}^L \neq \mathbf{U}^R$. It is the kind of situation that occurs at each interface FP when using the SD method. The approach used to solve a Riemann problem depends on the system of equations that is considered and especially on whether \mathbf{U} is scalar or a vector and on whether \mathbf{E}_c is a linear or a nonlinear function of \mathbf{U} . In this work, only system of equations coming from conservation laws where \mathbf{U} is a vector and \mathbf{E}_c is a nonlinear function of \mathbf{U} will be considered. Lax [199] gave a review of the results for the general Riemann problem using conservation laws. However, efficiently solving a Riemann problem applied to conservation laws actually depends on the system of conservation laws from which Eq. (4.1) comes from [192]. As CFD simulations are considered in this work, the Riemann solvers will be introduced in the context of the unsteady Euler equations for which an expression in the 1D case was already shown in paragraph 4.1.1.

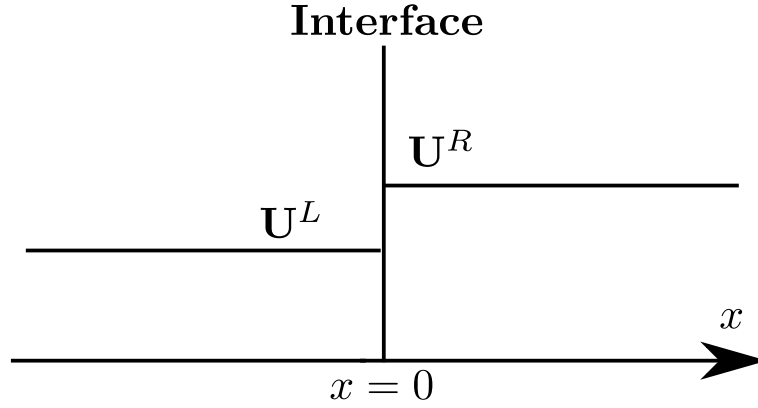


Figure 4.14. – Illustration of the Riemann problem in a 1D general case.

4.3.1.2. Exact and approximate solutions of the Riemann problem

The first one to propose a methodology to solve the Riemann problem for the unsteady 1D Euler equations for a calorically perfect gas was Godunov [200]. In his method, the Riemann problem is solved exactly but the computational cost is quite high. That is why, the development of Riemann solvers less costly, but with properties closed to the exact Riemann solver of Godunov, were on-trend. These Riemann solvers are usually refer to as approximate Riemann solvers. During this work, only the Harten Lax and van Leer Contact (HLLC) [201] Riemann solver was employed and is the topic of paragraph 4.3.2. A complete review of approximate Riemann solvers can be found in Toro's book [192].

4.3.2. The HLLC Riemann solver

4.3.2.1. General algorithm

The HLLC Riemann solver is usually presented using the three-dimensional Euler equations written for the 3D flux projected on an interface of unit normal $\mathbf{n}^u = [n_x^u, n_y^u, n_z^u]^T$ separating two states \mathbf{U}^L and \mathbf{U}^R as in Figure 4.14:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ \rho Y_k \end{pmatrix}, \quad \mathcal{F}_c(\mathbf{U}) \cdot \mathbf{n}^u \equiv \mathcal{F}_{c,n}(\mathbf{U}) = \begin{pmatrix} \rho u_n \\ \rho u u_n + P n_x^u \\ \rho v u_n + P n_y^u \\ \rho w u_n + P n_z^u \\ u_n (P + \rho E) \\ \rho u_n Y_k \end{pmatrix} \quad (4.90)$$

where $u_n = u n_x^u + v n_y^u + w n_z^u$ is the component of velocity acting in direction \mathbf{n}^u . This scheme assumes a three-wave model for the solution of the Riemann problem as illustrated in Figure 4.15 for $\mathbf{n}^u = [1, 0, 0]^T$. It consists of two acoustic waves with the smallest and largest speeds denoted respectively by S_L and S_R and a contact wave of speed S_* which must lie between S_L and S_R . The idea of the HLLC solver, is to compute the interface numerical flux based on the signs of S_L , S_* and S_R [192]:

$$\mathcal{F}_{c,n}^I = \begin{cases} \mathcal{F}_{c,n}^L & \text{if } S_L > 0 \\ \mathcal{F}_{c,n}^{*L} & \text{if } S_L \leq 0 < S_* \\ \mathcal{F}_{c,n}^{*R} & \text{if } S_* \leq 0 \leq S_R \\ \mathcal{F}_{c,n}^R & \text{if } S_R < 0 \end{cases} \quad (4.91)$$

where $\mathcal{F}_{c,n}^L = \mathcal{F}_{c,n}(\mathbf{U}^L)$, $\mathcal{F}_{c,n}^R = \mathcal{F}_{c,n}(\mathbf{U}^R)$. In Eq. (4.91), $\mathcal{F}_{c,n}^{*L}$ and $\mathcal{F}_{c,n}^{*R}$ are the intermediate fluxes in the star region whose expressions are determined using Rankine-Hugoniot conditions across each

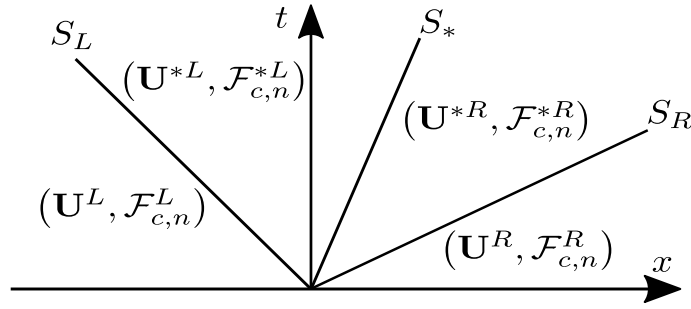


Figure 4.15. – Situation of the Riemann problem solved with the HLLC solver when $\mathbf{n}^u = [1, 0, 0]^T$. The star region consists of two constant states separated by a contact wave of speed S_* .

of the three waves:

$$\mathcal{F}_{c,n}^{*L} = \mathcal{F}_{c,n}^L + S_L (\mathbf{U}^{*L} - \mathbf{U}^L) \quad (4.92)$$

$$\mathcal{F}_{c,n}^{*R} = \mathcal{F}_{c,n}^{*L} + S_* (\mathbf{U}^{*R} - \mathbf{U}^{*L}) \quad (4.93)$$

$$\mathcal{F}_{c,n}^{*R} = \mathcal{F}_{c,n}^R + S_R (\mathbf{U}^{*R} - \mathbf{U}^R) \quad (4.94)$$

with \mathbf{U}^{*L} and \mathbf{U}^{*R} respectively the left and right states in the star region. They are also unknowns of the problem. Consequently, there are four unknowns $\mathcal{F}_{c,n}^{*L}$, $\mathcal{F}_{c,n}^{*R}$, \mathbf{U}^{*L} and \mathbf{U}^{*R} and three equations Eq. (4.92), Eq. (4.93) and Eq. (4.94) so that the problem is underdetermined. One condition remains to be found and a judicious choice is to impose a condition satisfied by the exact solution of the Riemann problem. From this exact solution for the Euler equations, it can be assumed that [192]:

$$P^{*L} = P^{*R} \equiv P^* \quad (\text{Continuity of pressure in the star region}) \quad (4.95)$$

$$u_n^{*L} = u_n^{*R} \equiv S^* \quad (\text{Continuity of normal velocity in the star region}) \quad (4.96)$$

The objective is then to find expressions for each variables in the star region. Assuming that S_L and S_R are known, the first components of Eq. (4.92) and Eq. (4.94) gives the densities in the star region:

$$\rho^{*L} = \rho^L \frac{S_L - u_n^L}{S_L - S_*} \quad \text{and} \quad \rho^{*R} = \rho^R \frac{S_R - u_n^R}{S_R - S_*} \quad (4.97)$$

Multiplying second, third and fourth components of Eq. (4.92) and Eq. (4.94) by n_x^u , n_y^u and n_z^u respectively and then summing gives:

$$P^{*L} = P^L + \rho^{*L} (S_L - S_*) S_* - \rho^L u_n^L (S_L - u_n^L) \quad \text{and} \quad P^{*R} = P^R + \rho^{*R} (S_R - S_*) S_* - \rho^R u_n^R (S_L - u_n^R) \quad (4.98)$$

Then using Eq. (4.97), pressures in the star region have the following expressions:

$$P^{*L} = P^L + \rho^L (u_n^L - S_L) (u_n^L - S_*) \quad \text{and} \quad P^{*R} = P^R + \rho^R (u_n^R - S_R) (u_n^R - S_*) \quad (4.99)$$

Because of Eq. (4.95) and using expressions in Eq. (4.99), the contact wave speed is obtained:

$$S_* = \frac{P^R - P^L - \rho^L u_n^L (u_n^L - S_L) + \rho^R u_n^R (u_n^R - S_R)}{\rho^R (u_n^R - S_R) - \rho^L (u_n^L - S_L)} \quad (4.100)$$

Now that densities $\rho^{*L/R}$, pressure $P^{*L/R} = P^*$ and normal velocity S^* have been computed, they can be employed to obtain the remaining components of \mathbf{U}^{*L} and \mathbf{U}^{*R} thanks to Eq. (4.92) and Eq. (4.94):

$$(\rho u)^{*L/R} = \frac{(P^* - P^{L/R}) n_x^u + \rho^{L/R} u^{L/R} (S_{L/R} - u_n^{L/R})}{S_{L/R} - S_*} \quad (4.101)$$

$$(\rho v)^{*L/R} = \frac{(P^* - P^{L/R}) n_y^u + \rho^{L/R} v^{L/R} (S_{L/R} - u_n^{L/R})}{S_{L/R} - S_*} \quad (4.102)$$

$$(\rho w)^{*L/R} = \frac{(P^* - P^{L/R}) n_z^u + \rho^{L/R} w^{L/R} (S_{L/R} - u_n^{L/R})}{S_{L/R} - S_*} \quad (4.103)$$

$$(\rho E)^{*L/R} = \frac{P^* S_* - P^{L/R} u_n^{L/R} + (\rho E)^{L/R} (S_{L/R} - u_n^{L/R})}{S_{L/R} - S_*} \quad (4.104)$$

$$(\rho Y_k)^{*L/R} = (\rho Y_k)^{L/R} \frac{S_{L/R} - u_n^{L/R}}{S_{L/R} - S_*} \quad (4.105)$$

Finally, $\mathcal{F}_{c,n}^{*L}$ and $\mathcal{F}_{c,n}^{*R}$ can be computed and the final choice for the interface numerical flux is obtained through Eq. (4.91). It should be pointed out that $\mathbf{U}^{*L/R}$ is never employed to compute $\mathcal{F}_{c,n}^{*L/R}$ using $\mathcal{F}_{c,n}^{*L/R} = \mathcal{F}_{c,n}(\mathbf{U}^{*L/R})$ as opposed to what is done in approximate-state Riemann solvers. The methodology described here is valid for any EOS once S_L and S_R are available. The EOS impacts only the values of S_L and S_R because the sound speed expression is different for each EOS.

4.3.2.2. Wave speed estimates in the HLLC Riemann solver

All the previous computations were done assuming that values of S_L and S_R were known. Several choices are available in the literature and the one that was done in this work is:

$$S_L = \min(u_n^L - c^L, \tilde{u}_n - \tilde{c}, u_n^R - c^R) \quad \text{and} \quad S_R = \max(u_n^L + c^L, \tilde{u}_n + \tilde{c}, u_n^R + c^R) \quad (4.106)$$

where c^L and c^R are the sound speeds respectively in the left and right state. Since ideal multi-species gases are employed in this work, these sound speeds are given by:

$$c^{L/R} = \sqrt{\frac{\gamma^{L/R} P^{L/R}}{\rho^{L/R}}} \quad (4.107)$$

In Eq. (4.107), γ^L and γ^R are the heat capacity ratios respectively in the left and right states. In Eq. (4.106), variables with a symbol $\tilde{(\cdot)}$ are Roe's averages introduced in [202] and are recalled here for completeness:

$$\tilde{r} = \sqrt{\frac{\rho^R}{\rho^L}} \quad (4.108)$$

$$\tilde{u}_n = \frac{\tilde{r} u_n^R + u_n^L}{1 + \tilde{r}} \quad (4.109)$$

$$\tilde{H} = \frac{\tilde{r} H^R + H^L}{1 + \tilde{r}} \quad (4.110)$$

$$\tilde{\gamma} = \frac{\tilde{r} \gamma^R + \gamma^L}{1 + \tilde{r}} \quad (4.111)$$

$$\tilde{c}^2 = (\tilde{\gamma} - 1) \left(\tilde{H} - \frac{\tilde{u}_n^2}{2} \right) \quad (4.112)$$

where $H = E + P/\rho$ stands for the total enthalpy. The wave speed estimates used in this work are actually a combination of what is done in [85] completed by the Roe's averages terms which return the exact solution for an isolated shock and satisfies entropy inequality [201, 203]. However, it should be mentioned that technically, Eq. (4.112) is valid only for a calorically mono-species gas ($\tilde{\gamma} = \gamma$ is constant) as presented by Roe in its original paper [202]. For a multi-species thermally perfect gas, another expressions have been derived such as in the recent work of Lecointre *et al.* [204] who have developed a methodology to compute \tilde{c} for a general EOS with a multi-species gas. It could be interesting to try their algorithm in the future although no apparent issues were seen in this work concerning the computation of \tilde{c} using Eq. (4.112). Finally, there is another methodology, based on pressure estimates, to compute S_L and S_R as proposed in the original HLLC paper [205]. It has the advantage to not use the Roe's averages avoiding questions on their validity in the multi-species case. Nevertheless, the entropic character of the wave speed estimates is lost. Still, this approach is widely employed in the DG community that simulates multi-species reacting flows [25, 27, 92].

4.4. Gradient computation in 3D for diffusive fluxes

In paragraph 4.2.3, the need to compute the gradients of the conservative variables was highlighted in order to get the diffusive fluxes. The objectives of this section are two folds:

- Explain the original methodology of Sun *et al.* [47], already explained in a simpler 1D case in paragraph 4.1.6, in 3D to compute the gradient needed for the evaluation of diffusive fluxes.
- Extend the SDLIFT formulation, originally introduced in 1D (see paragraph 4.1.7), in 3D.

4.4.1. Original formulation

As in the 1D case, the algorithm starts with the conservative variables in the physical space at FP evaluated using Eq. (4.77). At interface FP, \mathbf{U} is discontinuous and the solution at the interface is taken as the average between the left and right states following Eq. (4.27). Once all common values have been computed for all interface FP along ξ , η and ζ directions, \mathbf{U} is continuous across elements and is denoted by \mathbf{U}_{FP}^C . Values of \mathbf{U}_{FP}^C along ξ -direction (respectively η -direction and ζ -direction) are used to build a $(p+1)$ -degree polynomial representation of \mathbf{U}_{FP}^C in ξ -direction (respectively η -direction and ζ -direction) as it is done for fluxes in Eq. (4.78), Eq. (4.79) and Eq. (4.80). Then to compute the gradient of any component U_i^C ($i \in \llbracket 1, N_{eq} \rrbracket$) of \mathbf{U}_{FP}^C at SP, two options are available depending on what is stored by the CFD code employed. Starting from the gradient definition in Cartesian coordinates:

$$\begin{aligned}
 \nabla_{\mathbf{x}} U_i^C &\equiv \frac{\partial U_i^C}{\partial x} \mathbf{e}_x + \frac{\partial U_i^C}{\partial y} \mathbf{e}_y + \frac{\partial U_i^C}{\partial z} \mathbf{e}_z \\
 \Rightarrow \nabla_{\mathbf{x}} U_i^C &= \left(\xi_x \frac{\partial U_i^C}{\partial \xi} + \eta_x \frac{\partial U_i^C}{\partial \eta} + \zeta_x \frac{\partial U_i^C}{\partial \zeta} \right) \mathbf{e}_x + \left(\xi_y \frac{\partial U_i^C}{\partial \xi} + \eta_y \frac{\partial U_i^C}{\partial \eta} + \zeta_y \frac{\partial U_i^C}{\partial \zeta} \right) \mathbf{e}_y \\
 &\quad + \left(\xi_z \frac{\partial U_i^C}{\partial \xi} + \eta_z \frac{\partial U_i^C}{\partial \eta} + \zeta_z \frac{\partial U_i^C}{\partial \zeta} \right) \mathbf{e}_z \\
 \Rightarrow \nabla_{\mathbf{x}} U_i^C &= [J^{-1}]^T \cdot \nabla_{\xi} U_i^C
 \end{aligned} \tag{4.113}$$

where $\mathbf{e}_x = (1, 0, 0)^T$, $\mathbf{e}_y = (0, 1, 0)^T$ and $\mathbf{e}_z = (0, 0, 1)^T$ are the unit vectors of the Cartesian basis. Thus, by deriving at SP the $(p+1)$ -degree polynomial created using \mathbf{U}_{FP}^C values and storing J^{-1} components at SP also, Eq. (4.113) is the first option to obtain the gradients at SP. The second option starts from Eq. (4.113), multiplied it by $|J|$ and use the following tensor identities for the right hand

side:

$$|J| \left[J^{-1} \right]^T \cdot \nabla_{\xi} U_i^C = \nabla_{\xi} \cdot \left(|J| U_i^C \left[J^{-1} \right]^T \right) - U_i^C \nabla_{\xi} \cdot \left(|J| \left[J^{-1} \right]^T \right) \quad (4.114)$$

In Eq. (4.114), the term $U_i^C \nabla_{\xi} \cdot \left(|J| \left[J^{-1} \right]^T \right)$ is zero for the same reasons mentioned in Eq. (4.65), then gradient in Cartesian coordinates can also be evaluated following Eq. (4.115):

$$\nabla_{\mathbf{x}} U_i^C = \frac{1}{|J|} \nabla_{\xi} \cdot \left(|J| U_i^C \left[J^{-1} \right]^T \right) \quad (4.115)$$

Therefore, Eq. (4.115) actually computes the gradient in Cartesian coordinates at SP by computing the divergence of the matrix $|J| U_i^C \left[J^{-1} \right]^T$, with respect to isoparametric coordinates, at SP. To do so, it needs to have the values of $|J| \left[J^{-1} \right]^T$ at FP before computing the divergence. However, compared to the first option, J^{-1} is not needed at SP in this approach. In Sun *et al.* [47], the second option is employed since they store only $|J| \left[J^{-1} \right]^T$ at FP and $|J|$ at SP.

Computing all these gradients for each conservative variable gives $\left(\nabla_{\mathbf{x}} \mathbf{U}^C \right)_{SP}$ which contains the gradients of \mathbf{U} with respect to Cartesian coordinates evaluated at SP. These gradients are then interpolated at FP along ξ , η and ζ directions using the same interpolation procedure than in Eq. (4.77) to obtain $\left(\nabla_{\mathbf{x}} \mathbf{U}^C \right)_{FP}$. Again, at interface FP, the gradient is discontinuous and in the original approach of Sun *et al.* [47], it is taken as the average between the left and right gradients following Eq. (4.30). but this time with derivatives along physical coordinates. Once the gradients are continuous at all FP, the diffusive flux tensor $\mathcal{F}_d = \mathcal{F}_d(\mathbf{U}, \nabla \mathbf{U})$ is computed at these FP. Finally, depending on whether the FP is internal or on an interface the following operations are done to get the flux tensor in the isoparametric domain:

- **Internal FP:** \mathcal{F}_d is added to \mathcal{F}_c and Eq. (4.68) is used.
- **Interface FP:** Eq. (4.86) is used with the \mathbf{n}_{FP} value associated to the correct interface FP direction.

4.4.2. SDLIFT formulation

The SDLIFT formulation in a general 3D framework is very similar to what is described in paragraph 4.1.7 for the 1D case except that derivatives in the tangential directions in isoparametric coordinates are also needed. Without loss of generality, the FP along ξ -direction in \mathcal{H} are considered but the reasoning is equivalent for the remaining FP along η and ζ directions. Starting from Eq. (4.77), a $(p-1)$ -degree polynomial with respect to ξ but of degree p in η and ζ can be built for the derivative of \mathbf{U} with respect to ξ :

$$\frac{\partial \mathbf{U}_h^e}{\partial \xi}(\xi, \eta, \zeta) = \sum_{j_1=1}^{N_{SP}^{1d}} \sum_{j_2=1}^{N_{SP}^{1d}} \sum_{j_3=1}^{N_{SP}^{1d}} \frac{\widehat{\mathbf{U}}_{(j_1, j_2, j_3)}^e}{|J|_{(j_1, j_2, j_3)}^e} \frac{\partial l_{j_1}^{SP}}{\partial \xi}(\xi) l_{j_2}^{SP}(\eta) l_{j_3}^{SP}(\zeta), \text{ for } (\xi, \eta, \zeta) \in [0, 1]^3 \quad (4.116)$$

If $\mathbf{U}_{e-1/2}$ and $\mathbf{U}_{e+1/2}$ are the common states values defined at interface FP along ξ -direction, $(\partial \mathbf{U}_h^e / \partial \xi)$ can be corrected into a p -degree polynomial as in the 1D case using correction functions of degree $p+1$:

$$\frac{\partial \widehat{\mathbf{U}}_h^e}{\partial \xi}(\xi, \eta, \zeta) = \frac{\partial \mathbf{U}_h^e}{\partial \xi}(\xi, \eta, \zeta) + \left(\mathbf{U}_{e-1/2} - \mathbf{U}_h^e(0, \eta, \zeta) \right) \frac{\partial g_e^R}{\partial \xi}(\xi) + \left(\mathbf{U}_{e+1/2} - \mathbf{U}_h^e(1, \eta, \zeta) \right) \frac{\partial g_e^L}{\partial \xi}(\xi) \quad (4.117)$$

Eq. (4.117) defines a p -degree polynomial along ξ , η and ζ directions but is discontinuous at interface FP along ξ -direction since it is almost sure that:

$$\frac{\partial \widetilde{\mathbf{U}}_h^{e-1}}{\partial \xi}(1, \eta, \zeta) \neq \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(0, \eta, \zeta) \quad \text{and} \quad \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(1, \eta, \zeta) \neq \frac{\partial \widetilde{\mathbf{U}}_h^{e+1}}{\partial \xi}(0, \eta, \zeta) \quad (4.118)$$

Thus, by defining common normal derivatives at these interface FP, respectively $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)_{e-1/2}$ and $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)_{e+1/2}$, a $(p+1)$ -degree polynomial in ξ and a p -degree polynomial in η and ζ for $\partial \mathbf{U} / \partial \xi$ at ξ -FP, continuous across elements, can be built:

$$\begin{aligned} \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi, \eta, \zeta) &= \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(\xi, \eta, \zeta) + \left[\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e-1/2} - \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(0, \eta, \zeta) \right] g_e^R(\xi) \\ &\quad + \left[\left(\frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi} \right)_{e+1/2} - \frac{\partial \widetilde{\mathbf{U}}_h^e}{\partial \xi}(1, \eta, \zeta) \right] g_e^L(\xi) \end{aligned} \quad (4.119)$$

Therefore, the use of Eq. (4.119) at FP along ξ -direction, gives the derivative of \mathbf{U} in ξ -direction for these FP denoted by $(\partial \mathbf{U} / \partial \xi)_{(k_1, j_2, j_3)}^C$. However, to compute the viscous fluxes at FP, derivatives of \mathbf{U} with respect to x , y and z are actually needed and not the derivatives with respect to ξ , η and ζ . The formula shown in Eq. (4.113) should be used to obtain the gradient in Cartesian coordinates from the derivatives with respect to ξ , η and ζ . The SDLIFT treatment already provides derivative along ξ with $(\partial \mathbf{U} / \partial \xi)_{(k_1, j_2, j_3)}^C$ so that it remains to find values for $(\partial \mathbf{U} / \partial \eta)_{(k_1, j_2, j_3)}^C$ and $(\partial \mathbf{U} / \partial \zeta)_{(k_1, j_2, j_3)}^C$. Actually, there are two possibilities to compute these tangential derivatives.

The first one starts by writing the corrected continuous $(p+1)$ -degree polynomials of \mathbf{U} respectively along η and ζ directions:

$$\widetilde{\mathbf{U}}_h^{e, \eta}(\xi, \eta, \zeta) = \mathbf{U}_h^e(\xi, \eta, \zeta) + \left(\mathbf{U}_{e-1/2}^\eta - \mathbf{U}_h^e(\xi, 0, \zeta) \right) g_e^R(\eta) + \left(\mathbf{U}_{e+1/2}^\eta - \mathbf{U}_h^e(\xi, 1, \zeta) \right) g_e^L(\eta) \quad (4.120)$$

$$\widetilde{\mathbf{U}}_h^{e, \zeta}(\xi, \eta, \zeta) = \mathbf{U}_h^e(\xi, \eta, \zeta) + \left(\mathbf{U}_{e-1/2}^\zeta - \mathbf{U}_h^e(\xi, \eta, 0) \right) g_e^R(\zeta) + \left(\mathbf{U}_{e+1/2}^\zeta - \mathbf{U}_h^e(\xi, \eta, 1) \right) g_e^L(\zeta) \quad (4.121)$$

where superscripts η and ζ have been added to the notations of the corrected polynomial $\widetilde{\mathbf{U}}_h^e$ and interface values $\mathbf{U}_{e-1/2}$ and $\mathbf{U}_{e+1/2}$ in these directions for clarity. Then, $\widetilde{\mathbf{U}}_h^{e, \eta}$ and $\widetilde{\mathbf{U}}_h^{e, \zeta}$ are derived respectively in η and ζ directions and evaluated at SP to get $(\partial \widetilde{\mathbf{U}}_h^{e, \eta} / \partial \eta)_{(j_1, j_2, j_3)}$ and $(\partial \widetilde{\mathbf{U}}_h^{e, \zeta} / \partial \zeta)_{(j_1, j_2, j_3)}$.

Finally, these derivatives at SP will be extrapolated at FP along ξ -direction which give discontinuous tangential derivatives at ξ -FP written as: $(\partial \mathbf{U} / \partial \eta)_{(k_1, j_2, j_3)}^D$ and $(\partial \mathbf{U} / \partial \zeta)_{(k_1, j_2, j_3)}^D$. At this point, tangential derivatives of \mathbf{U} in η and ζ directions evaluated at ξ -FP have been obtained but they are not continuous at interface FP as opposed to the normal derivatives that are built continuous thanks to the SDLIFT process. There are two choices available:

1. Keep the discontinuity for tangential derivatives in η and ζ and use Eq. (4.113) to obtain the gradient in Cartesian coordinates. This is probably what is done in Castonguay *et al.* [206] and in Sheshadri *et al.* [207] in the FR context although this point is not clearly specified.
2. Do the average of the tangential derivatives at each interface FP to have fully continuous derivatives in ξ , η and ζ . However, this choice will increase the stencil for the gradient computation and thus will change the compactness of diffusion schemes as the BR2 or the IC introduced in paragraphs 4.1.7.3 and 4.1.7.4 in the 1D case.

In this work, for this first choice for computing tangential derivatives, it was chosen to keep the discontinuity for tangential derivatives. This first possibility for obtaining tangential derivatives is named the classical approach.

The second possibility to obtain tangential derivatives of \mathbf{U} at FP is the methodology described by Huynh [194]. It starts from the corrected continuous $(p+1)$ -degree polynomial in ξ , and of degree p in η and ζ , for \mathbf{U} :

$$\widetilde{\mathbf{U}}_h^e(\xi, \eta, \zeta) = \mathbf{U}_h^e(\xi, \eta, \zeta) + \left(\mathbf{U}_{e-1/2} - \mathbf{U}_h^e(0, \eta, \zeta) \right) g_e^R(\xi) + \left(\mathbf{U}_{e+1/2} - \mathbf{U}_h^e(1, \eta, \zeta) \right) g_e^L(\xi) \quad (4.122)$$

Eq. (4.122) evaluated at ξ -FP gives $\widetilde{\mathbf{U}}_{h(k_1, j_2, j_3)}^e$ which are used to compute tangential derivatives at ξ -FP:

$$\left(\frac{\partial \mathbf{U}}{\partial \eta} \right)_{(k_1, j_2, j_3)}^C = \sum_{j_2=1}^{N_{SP}^{1d}} \widetilde{\mathbf{U}}_{h(k_1, j_2, j_3)}^e \frac{\partial l_{j_2}^{SP}}{\partial \eta}(\eta_{j_2}), \quad (k_1, j_2, j_3) \in \llbracket 1, N_{FP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.123)$$

$$\left(\frac{\partial \mathbf{U}}{\partial \zeta} \right)_{(k_1, j_2, j_3)}^C = \sum_{j_3=1}^{N_{SP}^{1d}} \widetilde{\mathbf{U}}_{h(k_1, j_2, j_3)}^e \frac{\partial l_{j_3}^{SP}}{\partial \zeta}(\zeta_{j_3}), \quad (k_1, j_2, j_3) \in \llbracket 1, N_{FP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \times \llbracket 1, N_{SP}^{1d} \rrbracket \quad (4.124)$$

Therefore, derivatives of \mathbf{U} along η and ζ directions for ξ -FP are also obtained with Huynh's approach. There are continuous (superscript C) at interface FP since $\widetilde{\mathbf{U}}_{h(1, j_2, j_3)}^e$ and $\widetilde{\mathbf{U}}_{h(N_{FP}^{1d}, j_2, j_3)}^e$ are the common values at interface FP along ξ direction. However, tangential derivatives introduced in Eq. (4.123) and Eq. (4.124) define $(p-1)$ -degree polynomials respectively in η and ζ and not of degree p as in the original approach. This second possibility for computing tangential derivatives is named the Huynh's approach.

Once tangential derivatives along η and ζ have been obtained, either with classical or Huynh's approach, Eq. (4.113) is employed to get the gradient in Cartesian coordinates at ξ -FP. At interface FP, these Cartesian gradients are still discontinuous for two reasons:

1. If the classical approach is considered, $(\partial \mathbf{U} / \partial \eta)_{(k_1, j_2, j_3)}^D$ and $(\partial \mathbf{U} / \partial \zeta)_{(k_1, j_2, j_3)}^D$ are discontinuous at interface FP.
2. The terms $\eta_x, \eta_y, \eta_z, \zeta_x, \zeta_y$ and ζ_z of J^{-1} are discontinuous at ξ interface FP. The three remaining terms of J^{-1} , ξ_x, ξ_y and ξ_z are continuous at ξ interface FP since they define the normal vectors at these FP which is unique and thus independent on which side of the interface it is computed.

This is independent of the approach used for computing tangential derivatives.

Consequently, Eq. (4.30) is employed to define unique Cartesian gradients at interface FP. Finally, the evaluation of diffusive flux tensor \mathcal{F}_d at ξ -FP is done as in the original formulation. The way to compute common states and common normal derivatives is exactly the same as explained in the 1D case for the three diffusion schemes used in this work: BR1, BR2 and IC introduced respectively in paragraphs 4.1.7.2, 4.1.7.3 and 4.1.7.4. The main steps to compute the gradients in Cartesian coordinates of conservatives variables at ξ -FP using the SDLIFT formulation are summed up in Algorithm 3 of Section B.2. The reasoning is exactly the same for η and ζ directions.

4.5. Local polynomial degree adaptation

One of the most interesting feature of HO polynomial methods is that they can locally adapt the polynomial degree to have high polynomial degrees only in regions of interest. This methodology is often referred to as p -adaptation or p -refinement and, similarly to mesh refinement, is an excellent way to reduce the total number of DOF, which surely saves computational time, without losing accuracy. It will be employed in this work in 2D and 3D cases.

4.5.1. Sensors

The polynomial degree within each element can be set by hand by the user or automated using a user-defined sensor. Multiple sensor definitions exist depending on the simulation considered. The sensors used for mesh refinement could be used. For instance, sensors can be based on error estimates [95, 208] or on time-averaged of target quantities [209]. In this work, time-averaged of some quantities like the norm of the density or temperature gradients or of the HRR have been considered. For instance, in the case of the norm of the density gradient, the following sensor value θ_e is computed in each element Ω_e :

$$\theta_e = \frac{1}{t_{\text{avg}}|\Omega_e|} \int_t^{t+t_{\text{avg}}} \int_{\Omega_e} \|\nabla \rho\|_2 d\Omega_e dt \text{ and } \tilde{\theta}_e = \frac{\theta_e}{\max_{\Omega}(\theta_e)} \quad (4.125)$$

where $|\Omega_e|$ is the volume of Ω_e and t_{avg} is the time averaging duration. In Eq. (4.125), $\tilde{\theta}_e$ is the scaled sensor value of θ_e to keep it between 0 and 1 in the whole domain. Using $\tilde{\theta}_e$, the polynomial degree p_e within each element is set according to:

$$p_e = p_{\min} + \text{INT} \left[\tanh \left(\alpha \tilde{\theta}_e \right) (p_{\max} - p_{\min}) \right] \quad (4.126)$$

where p_{\min} and p_{\max} are respectively the minimum and maximum polynomial degrees set by the user, INT stands for the integer function and α is a smoothing parameter used to avoid jumps of more than one degree between adjacent elements as it is done in the DG framework [96].

4.5.2. Load-balancing

For parallel computations, having a variable value of p in the whole domain entails the use of load-balancing when the mesh is cut and distributed to processors in order to have more processors on zones where the polynomial degree is higher. In JAGUAR, it is done using the ParMETIS MPI-based parallel library [210] which takes a weight of $(p_e + 1)^d$ (number of DOF per element in dimension d) associated to each element and re-assigns elements to processors that had only elements with low weight values after the first partitioning of the mesh (which was done before p_e was set).

4.5.3. Methodology

The polynomial distribution can be set once at the beginning of the computation based on $\tilde{\theta}_e$ values obtained from a previous computation, it is called then static p -adaptation, or it can evolve during the computation, it is then named dynamic p -adaptation. Static p -adaptation requires a first computation, usually done at a small constant polynomial degree in all elements, but is easier to use than dynamic p -adaptation which needs to reallocate arrays and adapt load balancing during the computation. However, dynamic p -adaptation has the advantage to adjust to the solution during the simulation. In this work, only static p -adaptation was considered.

4.5.4. The mortar element method

When using p -adaptation, the polynomial degree of each element, noted p_e , can differ from one element to another. Interfaces between two elements of different degrees have FP placed at different locations. Figure 4.16 illustrates this situation for an interface between a left element with $p_L = 1$ and a right element with $p_R = 2$. Kopriva [43] proposed to solve this problem by using a mortar element

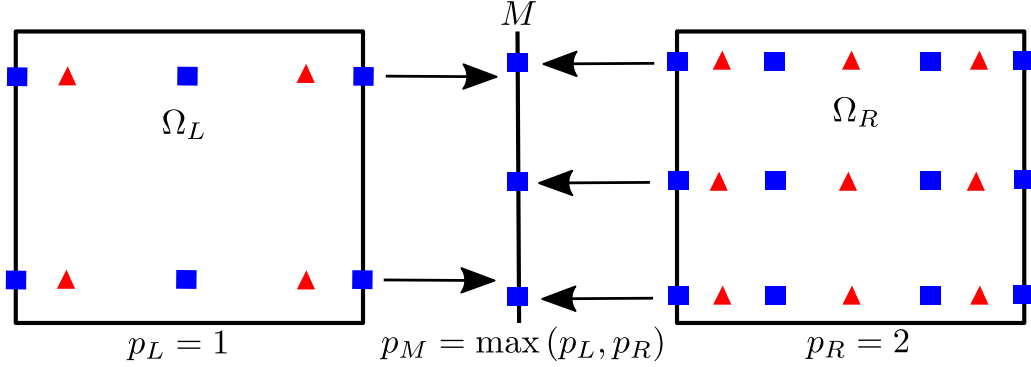


Figure 4.16. – Illustration of the mortar element method at an interface between a left element with $p_L = 1$ and a right element with $p_R = 2$. Red triangles represent SP and blue squares represent FP in the ξ direction.

method [211] where a fictive interface M , called mortar, is introduced to compute interface fluxes. In the general case, the mortar degree p_M is equal to $\max(p_L, p_R)$, thus $p_M = 2$ in the example of Figure 4.16. The mortar element treatment is different for the convective and diffusive fluxes since the first one is only a function of \mathbf{U} whereas the second one is a function of both \mathbf{U} and $\nabla \mathbf{U}$.

For the convective fluxes at an interface where $p_L \neq p_R$, the mortar element method is done in three steps [96]:

1. \mathbf{U}^L is projected on M using an unweighted L^2 -projection to get $\mathbf{U}^{M,L}$ and $\mathbf{U}^{M,R} = \mathbf{U}^R$ since $p_M = p_R$.
2. A Riemann problem is solved between states $\mathbf{U}^{M,L}$ and $\mathbf{U}^{M,R}$ to get convective fluxes on Mortar FP noted $\mathcal{F}_{c,n}^M$.
3. $\mathcal{F}_{c,n}^M$ is projected back on Ω_L to get $\mathcal{F}_{c,n}^{\Omega_L}$ and $\mathcal{F}_{c,n}^{\Omega_R} = \mathcal{F}_{c,n}^M$ since $p_M = p_R$.

$\mathcal{F}_{c,n}^{\Omega_L}$ and $\mathcal{F}_{c,n}^{\Omega_R}$ are the convective fluxes values at interface FP respectively in Ω_L and Ω_R that will be used to compute the flux divergence at SP in these elements. The projections mentioned in steps 1 and 3 are explained in Appendix D for both 2D and 3D cases.

For the diffusive fluxes at an interface where $p_L \neq p_R$, the mortar element method is done this time in eight steps [96]:

1. Same step as step 1 of the convective flux treatment on mortar to obtain $\mathbf{U}^{M,L}$ and $\mathbf{U}^{M,R}$.
2. Common interface values on mortar FP, noted $\mathbf{U}^{M,I}$, are computed from $\mathbf{U}^{M,L}$ and $\mathbf{U}^{M,R}$ based on the diffusion scheme used.
3. $\mathbf{U}^{M,I}$ is projected back on Ω_L to get $\mathbf{U}^{\Omega_L,I}$ and for interface FP on Ω_R : $\mathbf{U}^{\Omega_R,I} = \mathbf{U}^{M,I}$ since $p_M = p_R$.
4. Gradients of \mathbf{U} in Ω_L and Ω_R are computed with $\mathbf{U}^{\Omega_L,I}$ and $\mathbf{U}^{\Omega_R,I}$ at interface FP.
5. Viscous fluxes $\mathcal{F}_{d,n}^{\Omega_L}$ and $\mathcal{F}_{d,n}^{\Omega_R}$ are evaluated at interface FP of Ω_L and Ω_R , independently.
6. $\mathcal{F}_{d,n}^{\Omega_L}$ is projected on M using an unweighted L^2 -projection to get $\mathcal{F}_{d,n}^{M,L}$ and $\mathcal{F}_{d,n}^{M,R} = \mathcal{F}_{d,n}^{\Omega_R}$ since $p_M = p_R$.
7. Viscous fluxes on M are averaged to get $\mathcal{F}_{d,n}^M = (\mathcal{F}_{d,n}^{M,L} + \mathcal{F}_{d,n}^{M,R}) / 2$.

8. $\mathcal{F}_{d,n}^M$ is projected back on element interfaces as in step 3 of the convective flux treatment.

Except the number of steps, the main difference with convective fluxes treatment is that the viscous fluxes are computed at the interface of each element prior to being projected onto the mortar element. Note that in practice, $\mathcal{F}_{d,n}^M$ is added to $\mathcal{F}_{c,n}^M$ and then the back projection from mortar to element interfaces is done on $\mathcal{F}_{c,n}^M + \mathcal{F}_{d,n}^M$ to only have one projection operation of the fluxes instead of two. Finally, it should be highlighted that Kopriva demonstrated that this mortar element method is conservative [43]. This treatment is the only difference between a computation at constant degree p and a computation using p -adaptation.

4.6. Temporal discretization

During the previous sections of this chapter, the way to spatially discretize Eq. (4.67) using the SD method was explained ending up with Eq. (4.84) which can be put under the generic form:

$$\frac{d\hat{\mathbf{U}}}{dt} + \nabla_{\xi} \cdot \hat{\mathcal{F}} = \hat{\mathbf{S}} \iff \frac{d\hat{\mathbf{U}}}{dt} = \hat{\mathbf{S}} - \nabla_{\xi} \cdot \hat{\mathcal{F}} \equiv \mathcal{R} \quad (4.127)$$

where $\mathcal{R} = \hat{\mathbf{S}} - \nabla_{\xi} \cdot \hat{\mathcal{F}}$ is called the Residual. It corresponds to the discretization of the source terms and of the flux divergence. Following this definition and as an example, the simplest way to discretize Eq. (4.127) in order to advance $\hat{\mathbf{U}}$ in time from t to $t + \Delta t$, $\Delta t(t)$ being the timestep computed at time t , is by doing an explicit Euler integration:

$$\hat{\mathbf{U}}(t + \Delta t(t)) = \hat{\mathbf{U}}(t) + \Delta t(t) \mathcal{R}(t) \text{ or also } \hat{\mathbf{U}}^{n+1} = \hat{\mathbf{U}}^n + (\Delta t)^n \mathcal{R}^n \quad (4.128)$$

with n the considered time iteration. The issue with such explicit Euler integration is that it is only first-order accurate in time. More complex time integration procedure have been developed with the objective to have higher order of accuracy in time. One of the most common choice is to use Runge-Kutta (RK) schemes which were employed in this work.

4.6.1. Runge-Kutta schemes

4.6.1.1. Characteristics of a Runge-Kutta scheme

A RK scheme has four main characteristics which are:

1. Its number of stages, noted Q here.
2. Its order of accuracy, noted S here.
3. Its stability: A-stable [212] or L-stable [213]. L-stability is a special case of A-stability which is better for solving nonlinear stiff equations.
4. Its type of integration: explicit or implicit.

RK methods are denoted "nameRK(Q,S)" which allows the reader to directly know the order of accuracy and the number of stages.

4.6.1.2. General formula of a Runge-Kutta scheme

The RK integration process is the sum of two tasks:

- Task 1: the Q stage values $\hat{\mathbf{U}}^q$ are computed using a stage equation:

$$\hat{\mathbf{U}}^q = \hat{\mathbf{U}}^n + (\Delta t)^n \sum_{j=1}^Q a_{qj} \mathcal{R}^j \text{ for } 1 \leq q \leq Q \quad (4.129)$$

with $\mathcal{R}^1 = \mathcal{R}^n$.

- Task 2: the solution at time instant $n + 1$ is computed using an update equation:

$$\hat{\mathbf{U}}^{n+1} = \hat{\mathbf{U}}^n + (\Delta t)^n \sum_{q=1}^Q b_q \mathcal{R}^q \quad (4.130)$$

where $\mathcal{R}^j = \mathcal{R}(\hat{\mathbf{U}}^j)$, the matrix $\mathbf{A} = [a_{qj}]_{1 \leq q, j \leq Q}$ and the vector $\mathbf{b} = [b_q]_{1 \leq q \leq Q}^T$ are given by the RK method. They are typically gathered in what is called a Butcher tableau [214] as represented in Table 4.1 in the general case.

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array}$$

Table 4.1. – General form of a Butcher tableau.

These coefficients are found by solving a system of equations which depends on the order of accuracy and also on dissipation and dispersion properties that are investigated to obtain a given scheme. An example of such system can be seen in [215]. Note that since \mathcal{R} does not depend on time in this work, the vector \mathbf{c} in the RK method is not used in Eq. (4.129) and Eq. (4.130). However, \mathbf{c} is employed to get the current time at a given RK stage (see Eq. (6.81)). Additionally, implicit RK schemes have not been considered throughout this work and only explicit RK schemes are discussed in the next paragraph.

4.6.1.3. Explicit Runge-Kutta schemes

For explicit RK schemes, $a_{qj} = 0$ for $j \geq q$ meaning that matrix \mathbf{A} is a strictly lower triangular matrix. Thus, each stage $\hat{\mathbf{U}}^q$ is "explicitly" known thanks to the values of the previous stages and Eq. (4.129) can be recast as:

$$\hat{\mathbf{U}}^q = \hat{\mathbf{U}}^n + (\Delta t)^n \sum_{j=1}^{q-1} a_{qj} \mathcal{R}^j \text{ for } 1 \leq q \leq Q \quad (4.131)$$

Explicit RK schemes are typically named ERK(Q,S) and Butcher's tableau for forward Euler, ERK(2,2) and ERK(4,4) methods can be found in [216] for illustration. These ERK schemes work well to integrate any equation but more complex schemes have been derived based on specific design criterion linked to special applications. For instance to reduce the memory cost of the RK method, low-storage RK schemes have been developed where $\hat{\mathbf{U}}^q$ can be computed using only the values of \mathcal{R}^{q-1} :

$$\hat{\mathbf{U}}^q = \hat{\mathbf{U}}^n + (\Delta t)^n \gamma_q \mathcal{R}^{q-1} \text{ for } 1 \leq q \leq Q \quad (4.132)$$

with $\mathcal{R}^0 = \mathcal{R}^n$ and $\boldsymbol{\gamma} = [\gamma_q]_{1 \leq q \leq Q}^T$ the input RK coefficients of the low-order RK scheme. Consequently, for low-storage RK schemes the solution at time iteration $n + 1$ is:

$$\hat{\mathbf{U}}^{n+1} \equiv \hat{\mathbf{U}}^Q = \hat{\mathbf{U}}^n + (\Delta t)^n \gamma_Q \mathcal{R}^{Q-1} \quad (4.133)$$

A very popular low-storage RK scheme is the explicit RK of second-order and six stages with low-dissipation and low-dispersion designed by Bogey and Bailly for aeroacoustic applications [217] noted ERK(6,2)LDLD here. More recently, Vanharen *et al.* [218] have also developed a fourth-order with six stages low-storage RK scheme designed for SD with a higher stability than the ERK(6,2)LDLD where the last two coefficients, γ_5 and γ_6 , depend on polynomial degree p . More details about low-storage RK schemes can be found in [219] especially on how Butcher's coefficients can be manipulated to obtain only one coefficient γ_q to update the solution at each stage. However, it should be mentioned that not all the RK schemes can be recast into low-storage RK processes [220, 221].

Another class of ERK schemes have been developed for shock applications where strong discontinuities appear in the computational domain. They were originally named as Total Variation Diminishing (TVD) RK schemes since they were designed so that the total variation (TV) of each component \hat{U}^n of the numerical solution vector $\hat{\mathbf{U}}^n$ [222]:

$$TV(\hat{U}^n) = \sum_{i=1}^{\text{DOF}} |\hat{U}_i^n - \hat{U}_{i-1}^n| \quad (4.134)$$

does not increase with time meaning that [222]:

$$TV(\hat{U}^{n+1}) \leq TV(\hat{U}^n) \quad (4.135)$$

The solution at each stage is usually presented under the form of Eq. (4.136) introduced by Shu and Osher [9]:

$$\begin{aligned} \hat{\mathbf{U}}^q &= \sum_{j=0}^{q-1} \left(\alpha_{qj} \hat{\mathbf{U}}^j + (\Delta t)^n \beta_{qj} \mathcal{R}^j \right) \text{ for } 1 \leq q \leq Q \\ \hat{\mathbf{U}}^0 &= \hat{\mathbf{U}}^n, \quad \hat{\mathbf{U}}^Q = \hat{\mathbf{U}}^{n+1} \end{aligned} \quad (4.136)$$

where coefficients α_{qj} and β_{qj} are given with the TVD-RK scheme. For consistency, $\sum_{j=0}^{q-1} \alpha_{qj} = 1$ for $q \in \llbracket 1, Q \rrbracket$. The formulation of Eq. (4.136) is used because it is easier to employ for building TVD-RK schemes than the one of Butcher showed in Eq. (4.129) and Eq. (4.130). However, every RK scheme written under Shu-Osher representation has an equivalent Butcher tableau and vice versa. The link between coefficients α , β and Butcher matrices can be found in [223, 224]. There is also a Python module with many RK methods and associated functions to compute Shu-Osher or Butcher representations [225]. In this work the three-stages third-order in time TVD-RK scheme of Gottlieb and Shu [222] and the five-stages fourth-order in time Strong Stability Preserving (SSP) RK scheme of Spiteri and Ruuth [226] were employed. A SSP scheme is actually the new name of a TVD scheme given by Gottlieb and Shu who claim that it is more suitable for the properties of such schemes [87]. Consequently, these two schemes used in this work will be named as SSP-ERK(3,3) and SSP-ERK(5,4). Their coefficients α and β and their equivalent Butcher tableau are summed up in Appendix E. Both schemes are widely employed in the HO community especially when simulating shocks or combustion applications [24–27, 85, 90]. That is why they have been retained in this work.

Note that no special time integration treatment is done for the combustion source terms such as Strang splitting [73] as in [24–27, 85], sub-cycling, or exponential integration like in [74]. This was not needed for the present work since the chemical schemes employed were not sufficiently stiff to have an impact on stability.

4.6.2. Stability conditions

Explicit time integration is associated with stability criteria limiting the timestep to a maximum value. These stability criteria are based on Courant-Friedrichs-Lewy (CFL) and Fourier (Fo) conditions linked to the spatial discretization of respectively the convective and diffusive fluxes. Firstly, the convective timestep computed at each SP, is obtained from the CFL number:

$$\text{CFL} = \frac{(\|\mathbf{u}\|_2 + c) \Delta t^{\text{conv}}}{\Delta_{SP}} \iff \Delta t^{\text{conv}} = \frac{\text{CFL} \times \Delta_{SP}}{(\|\mathbf{u}\|_2 + c)} \quad (4.137)$$

where $\|\mathbf{u}\|_2 + c$ and Δ_{SP} are respectively the maximum eigenvalue of the convective flux Jacobian matrix and the characteristic size evaluated at the given SP given by:

$$\Delta_{SP} = \frac{|J|_{SP}^{1/d}}{p+1} \quad (4.138)$$

Note that Eq. (4.138) assumes that SP are equally spaced which is actually untrue for any value of p if Eq. (4.14) is employed for their location in the isoparametric domain. Secondly, there is also a diffusive timestep involving viscosity, thermal and species diffusion effects through the Fo number:

$$\text{Fo} = \frac{\Delta t^{diff}}{\Delta_{SP}^2} \max\left(\nu, \frac{\nu}{Pr}, \frac{\nu}{Sc_k}\right) \iff \Delta t^{diff} = \frac{\text{Fo} \times \Delta_{SP}^2}{\max\left(\nu, \frac{\nu}{Pr}, \frac{\nu}{Sc_k}\right)} \quad (4.139)$$

The timestep at a given SP is then taken as the minimum between the convective and the diffusive timesteps:

$$\Delta t = \min(\Delta t^{\text{conv}}, \Delta t^{\text{diff}}) \quad (4.140)$$

The CFL and Fo numbers are input parameters and depend on the temporal scheme that is used and also on the value of the polynomial degree p . Generally, for a given time integration scheme, CFL and Fo values have to be decreased when p increases. Finally, a chemical timestep can be computed based on the smallest chemical timescale of the species $\tau_k = \rho Y_k / |\dot{\omega}_k|$:

$$\Delta t^{\text{chem}} = \min_k(\tau_k) = \min_k(\rho Y_k / |\dot{\omega}_k|) \quad (4.141)$$

In this work, chemical schemes employed were not stiff enough, and it was always verified that $\Delta t^{\text{chem}} > \Delta t^{\text{conv}}$ and $\Delta t^{\text{chem}} > \Delta t^{\text{diff}}$.

4.7. Summary of this chapter

In this chapter, the methodology to discretize conservation equations using the SD method first on 1D elements for illustration and then on 3D hexahedral elements has been explained. The SD method performs an isoparametric transformation of the computational domain transforming each mesh element into a standard reference element where governing equations are numerically solved. Inside the reference element, the SD method uses two sets of points:

- Solution points (SP) where the p -degree continuous polynomial of the solution vector is built using a Lagrange interpolation principle.
- Flux points (FP) where the $(p+1)$ -degree continuous polynomial of the flux vector is built still using a Lagrange interpolation principle.

Before being differentiated at SP, the flux polynomial is made continuous at element interfaces through the use of a Riemann solver for convective fluxes and of a diffusion scheme for diffusive fluxes. A special focus has been made on the HLLC Riemann solver [201] which was the one employed during this work. Concerning gradient computation and diffusion schemes, the classical approach of Sun *et al.* [47] and a completely new approach called SDLIFT formulation were introduced. The latter was designed following the FR framework and aims at reducing the stencil of gradient computation in order to gain in stability for stiff problems.

One of the best feature of the SD method to reduce its computational cost, without losing accuracy, is local polynomial adaptation. The polynomial distribution can be set by hand or using a suitable sensor based on physical quantities. At an interface between two elements with different degrees, FP do not coincide and the mortar element method [211], firstly introduced by Kopriva [43] in the SD

4.7 Summary of this chapter

context, is used to evaluate interface fluxes. Finally, when the SD discretization is completed, the equations are integrated in time using explicit SSP RK schemes either of third or fourth order in time. Due to this explicit time-integration procedure, the timestep cannot exceed a maximum value given by CFL and Fourier stability conditions.

Boundary Conditions

Overview

| | |
|--|-----|
| 5.1. Navier-Stokes Characteristic Boundary Conditions for a multi-species non-reacting gas | 88 |
| 5.1.1. Imposing derivatives at FP in SD | 88 |
| 5.1.2. Generalized characteristic form of the mono-species and non-reacting Navier-Stokes equations | 89 |
| 5.1.3. Compact formulation of the characteristic boundary conditions for a mono-species non-reacting gas | 91 |
| 5.1.4. Types of NSCBC | 93 |
| 5.2. Symmetry and no-slip wall boundary conditions for a multi-species non-reacting gas | 96 |
| 5.2.1. General case | 96 |
| 5.2.2. How to satisfy a normal gradient condition | 97 |
| 5.2.3. Practical implementation | 98 |
| 5.3. Summary of this chapter | 100 |

This chapter is dedicated to characteristic and wall boundary conditions for the general case of a multi-species gas for which a mono-species gas is only a particular case. The gas is also assumed to be non-reacting meaning that boundary conditions are supposed to be far from flames. The way to impose characteristic boundary conditions in the context of the SD method is described and some examples are shown on subsonic inflow and outflow conditions. Then, the methodology to reproduce symmetry, adiabatic and isothermal no-slip walls and especially the different options available at a boundary FP are discussed. Additionally, a formula to impose a normal gradient of a variable at a given FP is provided.

5.1. Navier-Stokes Characteristic Boundary Conditions for a multi-species non-reacting gas

The way to impose boundary conditions when solving the compressible form of the NSE is essential to avoid reflections of acoustic waves at the boundaries. Indeed, Dirichlet boundary conditions, which impose conservative variables directly at boundaries, make the problem ill-posed leading to unwanted behaviors or even instabilities. This is avoided by characteristic boundary conditions, widely employed in the context of FD [227] and FV [228], and also with more recent numerical methods such as DG [229], SD [117] and LBM [110]. Originally introduced by Thompson for hyperbolic systems of equations [230], they were extended to the non-reacting Navier-Stokes equations by Poinso and Lele [231] who named them as Navier-Stokes Characteristic Boundary Conditions (NSCBC). Later on, Baum *et al.* [232] and Sutherland *et al.* [233] developed NSCBC for multi-species and reacting gases. All these developments were written for low-order FD or FV schemes where no geometrical transformation of the physical domain is done since the equations are solved in a cartesian coordinate system. That is why Kim and Lee [234, 235] extended the NSCBC approach, in the mono-species gas case, to HO FD schemes where the equations are solved in a generalized coordinate system as in Eq. (4.67). Recently, based on the work of Kim and Lee, Fievet *et al.* [117] have developed a compact formulation of the NSCBC methodology, still in the mono-species case, for a generalized coordinate system and applied it in the SD context on academic 1D and 2D test cases.

The objective of this section is to present characteristic boundary conditions in the context of the SD method for multi-species mixtures. Firstly, the formulation of Kim and Lee is recalled and then the formulation of Fievet *et al.*, that is used in this work, is introduced. In particular, the extension of Fievet *et al.* algorithm to a multi-species non-reacting thermally perfect gas, which was needed for this work, is explained.

5.1.1. Imposing derivatives at FP in SD

In the SD algorithm, boundary conditions are set at FP since SP are always strictly inside the computational domain. It will be shown in the next parts that for particular boundary conditions, derivatives of some scalars or vectors have to be imposed at boundary FP. Let's denote by f any scalar (pressure, temperature or mass fraction) or vector (flux vector) function. Without loss of generality, if the boundary is at a constant ξ location in the isoparametric domain, f can be differentiated with respect to ξ (normal direction of the boundary) using derivatives of Lagrange polynomials at FP:

$$\frac{\partial f}{\partial \xi}(\xi) = \sum_{k=1}^{p+2} f_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi) \quad (5.1)$$

where f_k is the value of f at FP k in direction ξ . This situation is represented in Figure 5.1 for a boundary located at $\xi = 1$ when the polynomial degree is $p = 3$ in the element close to the boundary (five FP along ξ direction). In that case, imposing $(\partial f / \partial \xi)(1)$ to a given value implies that f at $\xi = 1$,

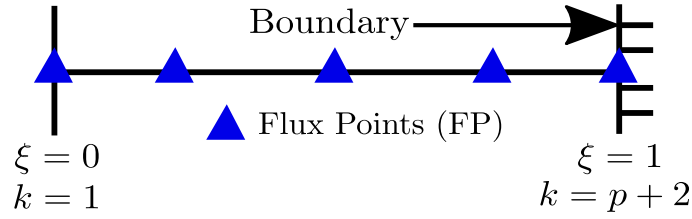


Figure 5.1. – Illustration of the computation of a function f at a boundary FP to impose $(\partial f / \partial \xi)$ at this boundary FP. The polynomial degree is set to $p = 3$ with five FP along direction ξ .

which is f_{p+2} , is computed by inverting Eq. (5.1) to get:

$$f_{p+2} = \frac{\frac{\partial f}{\partial \xi}(1) - \sum_{k=1}^{p+1} f_k \frac{\partial l_k^{FP}}{\partial \xi}(1)}{\frac{\partial l_{p+2}^{FP}}{\partial \xi}(1)} \quad (5.2)$$

The same principle can be used for a boundary located at $\xi = 0$, where imposing $(\partial f / \partial \xi)(0)$ leads to f at $\xi = 0$ given by Eq. (5.3):

$$f_1 = \frac{\frac{\partial f}{\partial \xi}(0) - \sum_{k=2}^{p+2} f_k \frac{\partial l_k^{FP}}{\partial \xi}(0)}{\frac{\partial l_1^{FP}}{\partial \xi}(0)} \quad (5.3)$$

More generally, this reasoning is exactly the same for boundaries along η and ζ directions in the isoparametric domain.

5.1.2. Generalized characteristic form of the mono-species and non-reacting Navier-Stokes equations

Considered the NSE of Eq. (4.50) with $\mathbf{S} = \mathbf{0}$. Using the flux splitting $\mathbf{E} = \mathbf{E}_c + \mathbf{E}_d$, $\mathbf{F} = \mathbf{F}_c + \mathbf{F}_d$ and $\mathbf{G} = \mathbf{G}_c + \mathbf{G}_d$, Eq. (4.50) becomes:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}_c}{\partial x} + \frac{\partial \mathbf{F}_c}{\partial y} + \frac{\partial \mathbf{G}_c}{\partial z} + \frac{\partial \mathbf{E}_d}{\partial x} + \frac{\partial \mathbf{F}_d}{\partial y} + \frac{\partial \mathbf{G}_d}{\partial z} = \mathbf{0} \quad (5.4)$$

Then, following the work of Kim and Lee [234, 235], the chain rule introduced in Eq. (4.64) is applied to Eq. (5.4) to get:

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} + & \left(\xi_x \frac{\partial \mathbf{E}_c}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}_c}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}_c}{\partial \xi} \right) + \left(\xi_x \frac{\partial \mathbf{E}_d}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}_d}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}_d}{\partial \xi} \right) \\ & + \left(\eta_x \frac{\partial \mathbf{E}_c}{\partial \eta} + \eta_y \frac{\partial \mathbf{F}_c}{\partial \eta} + \eta_z \frac{\partial \mathbf{G}_c}{\partial \eta} \right) + \left(\eta_x \frac{\partial \mathbf{E}_d}{\partial \eta} + \eta_y \frac{\partial \mathbf{F}_d}{\partial \eta} + \eta_z \frac{\partial \mathbf{G}_d}{\partial \eta} \right) \\ & + \left(\zeta_x \frac{\partial \mathbf{E}_c}{\partial \zeta} + \zeta_y \frac{\partial \mathbf{F}_c}{\partial \zeta} + \zeta_z \frac{\partial \mathbf{G}_c}{\partial \zeta} \right) + \left(\zeta_x \frac{\partial \mathbf{E}_d}{\partial \zeta} + \zeta_y \frac{\partial \mathbf{F}_d}{\partial \zeta} + \zeta_z \frac{\partial \mathbf{G}_d}{\partial \zeta} \right) = \mathbf{0} \end{aligned} \quad (5.5)$$

Remember that, because of Eq. (4.65), $|J| \times \text{Eq. (5.5)}$ is actually equal to:

$$|J| \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \hat{\mathbf{E}}_c}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}_c}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_c}{\partial \zeta} + \frac{\partial \hat{\mathbf{E}}_d}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}_d}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_d}{\partial \zeta} = \mathbf{0} \quad (5.6)$$

The idea of the NSCBC technique is to modify the flux derivative in the normal direction of the boundary to account for the characteristic waves that enter and leave the computational domain.

Let's consider, without loss of generality, a boundary located at a ξ -normal face (boundary at a constant ξ). Firstly, the characteristic decomposition of the ξ -convective flux is found using a

mathematical analysis of the 3D-Euler equations in isoparametric coordinates which is detailed in Appendix F for a multi-species thermally perfect gas:

$$A_\xi = \left(\xi_x \frac{\partial \mathbf{E}_c}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}_c}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}_c}{\partial \xi} \right) = P_U^{-1} \Lambda_{A_\xi} P_U \frac{\partial \mathbf{U}}{\partial \xi} \quad (5.7)$$

where $\Lambda_{A_\xi} = \text{diag}(u_n, u_n, u_n, u_n + c, u_n - c, u_n)$ is a diagonal matrix containing the eigenvalues of matrix A_ξ explicitly expressed in Eq. (F.19) and P_U is the transformation matrix from conservative to characteristic variables \mathbf{W} . This matrix can be expressed using P_Q and $\partial \mathbf{Q} / \partial \mathbf{U}$, whose expressions are derived in Appendix F in the multi-species case and are recalled here for clarity:

$$P_U = P_Q \cdot \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \quad (5.8)$$

$$P_Q = \begin{bmatrix} n_x^u & 0 & n_z^u \rho / c & -n_y^u \rho / c & -n_x^u / c^2 & 0 \\ n_y^u & -n_z^u \rho / c & 0 & n_x^u \rho / c & -n_y^u / c^2 & 0 \\ n_z^u & n_y^u \rho / c & -n_x^u \rho / c & 0 & -n_z^u / c^2 & 0 \\ 0 & n_x^u / \sqrt{2} & n_y^u / \sqrt{2} & n_z^u / \sqrt{2} & 1 / (\sqrt{2} \rho c) & 0 \\ 0 & -n_x^u / \sqrt{2} & -n_y^u / \sqrt{2} & -n_z^u / \sqrt{2} & 1 / (\sqrt{2} \rho c) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.9)$$

$$\frac{\partial \mathbf{Q}}{\partial \mathbf{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -u & \frac{1}{\rho} & 0 & 0 & 0 & 0 \\ \frac{\rho}{-v} & \rho & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{\rho}{-w} & 0 & \rho & 0 & 0 & 0 \\ \frac{\rho}{-Y_k} & 0 & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{(\gamma - 1) \|\mathbf{u}\|_2^2}{2} & (1 - \gamma) u & (1 - \gamma) v & (1 - \gamma) w & \gamma - 1 & (1 - \gamma) \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ \frac{-Y_k}{\rho} & 0 & 0 & 0 & 0 & \frac{1}{\rho} \end{bmatrix} \quad (5.10)$$

The wave strengths of the ξ -convective characteristics, introduced in the developments of Appendix F, can be also expressed with P_U and from Eq. (5.7):

$$\mathcal{L} = \Lambda_{A_\xi} P_U \frac{\partial \mathbf{U}}{\partial \xi} = P_U \left(\xi_x \frac{\partial \mathbf{E}_c}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}_c}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}_c}{\partial \xi} \right) \quad (5.11)$$

The matrix P_U is also used to define the following quantities:

$$\partial \mathbf{W} = P_U \partial \mathbf{U} \quad (5.12)$$

$$\mathcal{T}_c = P_U \left[\left(\eta_x \frac{\partial \mathbf{E}_c}{\partial \eta} + \eta_y \frac{\partial \mathbf{F}_c}{\partial \eta} + \eta_z \frac{\partial \mathbf{G}_c}{\partial \eta} \right) + \left(\zeta_x \frac{\partial \mathbf{E}_c}{\partial \zeta} + \zeta_y \frac{\partial \mathbf{F}_c}{\partial \zeta} + \zeta_z \frac{\partial \mathbf{G}_c}{\partial \zeta} \right) \right] \quad (5.13)$$

$$\mathcal{T}_d = P_U \left[\left(\eta_x \frac{\partial \mathbf{E}_d}{\partial \eta} + \eta_y \frac{\partial \mathbf{F}_d}{\partial \eta} + \eta_z \frac{\partial \mathbf{G}_d}{\partial \eta} \right) + \left(\zeta_x \frac{\partial \mathbf{E}_d}{\partial \zeta} + \zeta_y \frac{\partial \mathbf{F}_d}{\partial \zeta} + \zeta_z \frac{\partial \mathbf{G}_d}{\partial \zeta} \right) \right] \quad (5.14)$$

$$\mathcal{D} = P_U \left(\xi_x \frac{\partial \mathbf{E}_d}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}_d}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}_d}{\partial \xi} \right) \quad (5.15)$$

which allow to rewrite Eq. (5.5) into the characteristic space by multiplying it with matrix P_U to get:

$$\frac{\partial \mathbf{W}}{\partial t} + \mathcal{L} + \mathcal{T}_c + \mathcal{T}_d + \mathcal{D} = \mathbf{0} \quad (5.16)$$

Since $|J| \times \text{Eq. (5.5)}$ and Eq. (5.6) are equivalent, another expression for \mathcal{L} can be obtained by multiplying this time Eq. (5.6) with P_U :

$$\begin{aligned}\mathcal{L} &= \frac{P_U}{|J|} \left(\xi_x |J| \frac{\partial \mathbf{E}_c}{\partial \xi} + \xi_y |J| \frac{\partial \mathbf{F}_c}{\partial \xi} + \xi_z |J| \frac{\partial \mathbf{G}_c}{\partial \xi} \right) \\ \Rightarrow \mathcal{L} &= \frac{P_U}{|J|} \left(\frac{\partial \hat{\mathbf{E}}_c}{\partial \xi} - \left[\mathbf{E}_c \frac{\partial}{\partial \xi} (\xi_x |J|) + \mathbf{F}_c \frac{\partial}{\partial \xi} (\xi_y |J|) + \mathbf{G}_c \frac{\partial}{\partial \xi} (\xi_z |J|) \right] \right) \\ \Rightarrow \mathcal{L} &= \frac{P_U}{|J|} \left(\frac{\partial \hat{\mathbf{E}}_c}{\partial \xi} - \mathcal{A}_c(\xi) \right)\end{aligned}\quad (5.17)$$

Note that the term \mathcal{A}_c is zero when the mesh is orthogonal. Similarly, the expressions of \mathcal{T}_c , \mathcal{T}_d and \mathcal{D} can be recast [117]:

$$\mathcal{T}_c = \frac{P_U}{|J|} \left(\frac{\partial \hat{\mathbf{F}}_c}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_c}{\partial \zeta} - [\mathcal{A}_c(\eta) + \mathcal{A}_c(\zeta)] \right) \quad (5.18)$$

$$\mathcal{T}_d = \frac{P_U}{|J|} \left(\frac{\partial \hat{\mathbf{F}}_d}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_d}{\partial \zeta} - [\mathcal{A}_d(\eta) + \mathcal{A}_d(\zeta)] \right) \quad (5.19)$$

$$\mathcal{D} = \frac{P_U}{|J|} \left(\frac{\partial \hat{\mathbf{E}}_d}{\partial \xi} - \mathcal{A}_d(\xi) \right) \quad (5.20)$$

It should be noted that neglecting the \mathcal{A}_c and \mathcal{A}_d terms on unstructured meshes, leads to instabilities since they act like constant source terms at the boundary. As described in Fievet *et al.* [117], the equations presented in this section were used by Kim and Lee [234, 235] to derive characteristic boundary conditions in generalized coordinates. Their method follows Algorithm 4 shown in paragraph B.3.1.

5.1.3. Compact formulation of the characteristic boundary conditions for a mono-species non-reacting gas

In this paragraph, a very recent and different formulation is presented for the characteristic boundary conditions where the previous convective/diffusive characterization is replaced with a normal/tangent determination. A boundary located at a ξ -normal face is still considered without loss of generality.

5.1.3.1. General algorithm

Firstly, the convective and diffusive fluxes are summed up during their evaluation to get the compact form of Eq. (4.67) . Thanks to Eq. (5.17) and Eq. (5.20) , an expression of the total (convective and diffusive) flux derivative in the normal direction of the boundary, $\partial \hat{\mathbf{E}} / \partial \xi$, is found:

$$\begin{cases} \frac{\partial \hat{\mathbf{E}}_c}{\partial \xi} = |J| P_U^{-1} \cdot \mathcal{L} + \mathcal{A}_c(\xi) \\ \frac{\partial \hat{\mathbf{E}}_d}{\partial \xi} = |J| P_U^{-1} \cdot \mathcal{D} + \mathcal{A}_d(\xi) \end{cases} \Rightarrow \frac{\partial \hat{\mathbf{E}}}{\partial \xi} = |J| P_U^{-1} \cdot (\mathcal{L} + \mathcal{D}) + \mathcal{A}(\xi) \quad (5.21)$$

with $\mathcal{A}(\xi)$ the total correction term given by:

$$\mathcal{A}(\xi) = \mathcal{A}_c(\xi) + \mathcal{A}_d(\xi) = (\mathbf{E}_c + \mathbf{E}_d) \frac{\partial}{\partial \xi} (\xi_x |J|) + (\mathbf{F}_c + \mathbf{F}_d) \frac{\partial}{\partial \xi} (\xi_y |J|) + (\mathbf{G}_c + \mathbf{G}_d) \frac{\partial}{\partial \xi} (\xi_z |J|) \quad (5.22)$$

Consequently, Eq. (4.67) becomes:

$$|J| \frac{\partial \mathbf{U}}{\partial t} + |J| P_U^{-1} \cdot (\mathcal{L} + \mathcal{D}) = - \left(\frac{\partial \hat{\mathbf{F}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}}{\partial \zeta} + \mathcal{A}(\xi) \right) \quad (5.23)$$

Therefore, by multiplying Eq. (5.23) with P_U , a new equivalent form of Eq. (5.16) is obtained:

$$|J| \frac{\partial \mathbf{W}}{\partial t} + \mathcal{N} = -\mathcal{S} \quad (5.24)$$

where \mathcal{N} and \mathcal{S} are respectively the normal and tangential strengths of the characteristics waves which are given by:

$$\mathcal{N} \equiv \begin{pmatrix} \mathcal{N}_1 \\ \mathcal{N}_2 \\ \mathcal{N}_3 \\ \mathcal{N}_+ \\ \mathcal{N}_- \\ \mathcal{N}_{5+k} \end{pmatrix} = |J| (\mathcal{L} + \mathcal{D}) = P_U \left(\frac{\partial \hat{\mathbf{E}}}{\partial \xi} - \mathcal{A}(\xi) \right) \quad (5.25)$$

$$\mathcal{S} \equiv \begin{pmatrix} \mathcal{S}_1 \\ \mathcal{S}_2 \\ \mathcal{S}_3 \\ \mathcal{S}_+ \\ \mathcal{S}_- \\ \mathcal{S}_{5+k} \end{pmatrix} = P_U \left(\frac{\partial \hat{\mathbf{F}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}}{\partial \zeta} + \mathcal{A}(\xi) \right) = |J| (\mathcal{T}_c + \mathcal{T}_d) \quad (5.26)$$

To go back to the system of equations with primitive variables \mathbf{Q} , Eq. (5.24) is simply multiplied by P_Q^{-1} to get:

$$|J| \frac{\partial \mathbf{Q}}{\partial t} + P_Q^{-1} \mathcal{N} = -P_Q^{-1} \mathcal{S} \quad (5.27)$$

The boundary condition is then applied as follows. If the ξ -normal face is located at $\xi = 1$ as in Figure 5.1, the NSCBC treatment imposes a value of $\left(\partial \hat{\mathbf{E}} / \partial \xi \right)$ at the boundary FP in $\xi = 1$ according to the waves crossing this boundary. Firstly, Eq. (5.25) is employed to get first values of \mathcal{N} in $\xi = 1$. Then, depending on the type of NSCBC to be imposed, some values of \mathcal{N} are changed which gives a new normal strengths vector \mathcal{N}^* . Finally, a new flux derivative in the normal direction $\left(\partial \hat{\mathbf{E}} / \partial \xi \right)^*$ in $\xi = 1$ is obtained by inverting Eq. (5.25). However as shown in Eq. (4.84), in SD the flux derivatives are computed at SP from the flux polynomial built at the FP. Thus, the NSCBC approach actually provides the expected value of $\left(\partial \hat{\mathbf{E}} / \partial \xi \right)$ at the FP in $\xi = 1$ that will be used to deduce a flux value at this FP. Let's denote by $\hat{\mathbf{E}}^*$ the flux values obtained after the NSCBC treatment applied in $\xi = 1$ which is still unknown here. This flux at the boundary FP to be imposed is actually given by Eq. (5.2) with $f = \hat{\mathbf{E}}$:

$$\hat{\mathbf{E}}_{p+2}^* = \frac{\frac{\partial \hat{\mathbf{E}}^*}{\partial \xi}(1) - \sum_{k=1}^{p+1} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(1)}{\frac{\partial l_{p+2}^{FP}}{\partial \xi}(1)} \quad (5.28)$$

Applying Eq. (4.81) with this new flux at the boundary FP gives:

$$\begin{aligned} \frac{\partial \hat{\mathbf{E}}^*}{\partial \xi}(SP) &= \hat{\mathbf{E}}_{p+2}^* \frac{\partial l_{p+2}^{FP}}{\partial \xi}(SP) + \sum_{k=1}^{p+1} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(SP) \\ \Rightarrow \frac{\partial \hat{\mathbf{E}}^*}{\partial \xi}(SP) &= \left[\frac{\partial \hat{\mathbf{E}}^*}{\partial \xi}(1) - \sum_{k=1}^{p+1} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(1) \right] \frac{\frac{\partial l_{p+2}^{FP}}{\partial \xi}(SP)}{\frac{\partial l_{p+2}^{FP}}{\partial \xi}(1)} + \sum_{k=1}^{p+1} \hat{\mathbf{E}}_k \frac{\partial l_k^{FP}}{\partial \xi}(SP) \end{aligned} \quad (5.29)$$

Finally, by simultaneously adding and subtracting $\widehat{\mathbf{E}}_{p+2} \left(\partial l_{p+2}^{FP} / \partial \xi \right) (1)$ in the first sum, Eq. (5.29) becomes [117]:

$$\frac{\partial \widehat{\mathbf{E}}^*}{\partial \xi} (SP) = \left[\frac{\partial \widehat{\mathbf{E}}^*}{\partial \xi} (1) - \frac{\partial \widehat{\mathbf{E}}}{\partial \xi} (1) \right] \frac{\frac{\partial l_{p+2}^{FP}}{\partial \xi} (SP)}{\frac{\partial l_{p+2}^{FP}}{\partial \xi} (1)} + \frac{\partial \widehat{\mathbf{E}}}{\partial \xi} (SP) \quad (5.30)$$

Eq. (5.30) shows how the value of $\left(\partial \widehat{\mathbf{E}} / \partial \xi \right)$ at SP is corrected into $\left(\partial \widehat{\mathbf{E}} / \partial \xi \right)^* (SP)$ thanks to the NSCBC treatment. This corrected value will be the one used in Eq. (4.84) to march $\widehat{\mathbf{U}}$ in time. The main steps of this algorithm that shows characteristic boundary conditions with a normal/tangent determination of the waves is given in Algorithm 5 of paragraph B.3.2.

Compared to Kim and Lee's formulation presented in paragraph 5.1.2, this methodology has the advantage to not store both convective and diffusive fluxes separately which also avoids two extrapolations, instead of one here, for evaluating convective tangent fluxes and diffusive tangent fluxes. Additionally, correction terms \mathcal{A}_c and \mathcal{A}_d are needed along ξ direction only according to Eq. (5.25) and Eq. (5.26) whereas they have to be known along ξ , η and ζ directions under Kim and Lee formulation because of Eq. (5.11), Eq. (5.13), Eq. (5.14) and Eq. (5.15). This new formulation was successfully tested by Fievet *et al.* [117] on 1D and 2D Euler test cases such as one-dimensional acoustic wave or the convection of a 2D vortex. However, they did not test it with both a viscous mono-species or multi-species gases which is something performed in this work. It is presented in Chapter 6 for a viscous mono-species gas and in Chapters 7 and 8 for a viscous multi-species gas.

5.1.3.2. A comment on the computation of correction terms for the mesh's non-orthogonality

The computation of correction terms $\mathcal{A}(\xi)$ (see Eq. (5.22)), needed at NSCBC FP, can be done using two different approaches:

1. **Approach 1:** first store the $\partial \left(\xi_{x/y/z} |J| \right) / \partial \xi$ terms at SP in elements containing at least one NSCBC. Then, still at SP in these elements: compute $\mathbf{E}_{c/d}$, $\mathbf{F}_{c/d}$ and $\mathbf{G}_{c/d}$ and evaluate $\mathcal{A}(\xi)$. Finally, extrapolate $\mathcal{A}(\xi)$ from SP to the NSCBC FP. This method requires the knowledge of $\nabla \mathbf{U}$ at SP to compute diffusive fluxes. This is the one that was used during this work.
2. **Approach 2:** is an alternative to approach 1 to avoid storing $\nabla \mathbf{U}$ at SP in elements containing NSCBC faces. Thanks to the SD discretization, the fluxes in the isoparametric domain $\widehat{\mathbf{E}}$, $\widehat{\mathbf{F}}$ and $\widehat{\mathbf{G}}$ are known at FP respectively along ξ , η and ζ directions. Thus, from their flux polynomials built at FP, $\widehat{\mathbf{E}}$, $\widehat{\mathbf{F}}$ and $\widehat{\mathbf{G}}$ can be evaluated at SP and then Eq. (4.68) can be inverted to get \mathbf{E} , \mathbf{F} and \mathbf{G} at SP. Finally, correction terms are computed at SP and extrapolated at NSCBC FP as in approach 1. Consequently, this method requires one interpolation and one matrix inversion but there is no need to store $\nabla \mathbf{U}$ and evaluate fluxes at SP. Moreover, it is expected to have more accurate results than the first approach since it takes into account the flux values at interface FP obtained with the Riemann solver and the diffusion scheme.

For future work, it could be interesting to compare both approaches in terms of results and computational efficiency.

5.1.4. Types of NSCBC

The objective of this paragraph is to explain several types of NSCBC implemented in step 2 of Algorithm 5. A subsonic outflow and a subsonic inflow are considered along a ξ -normal boundary as shown in Figure 5.2 with their associated waves.

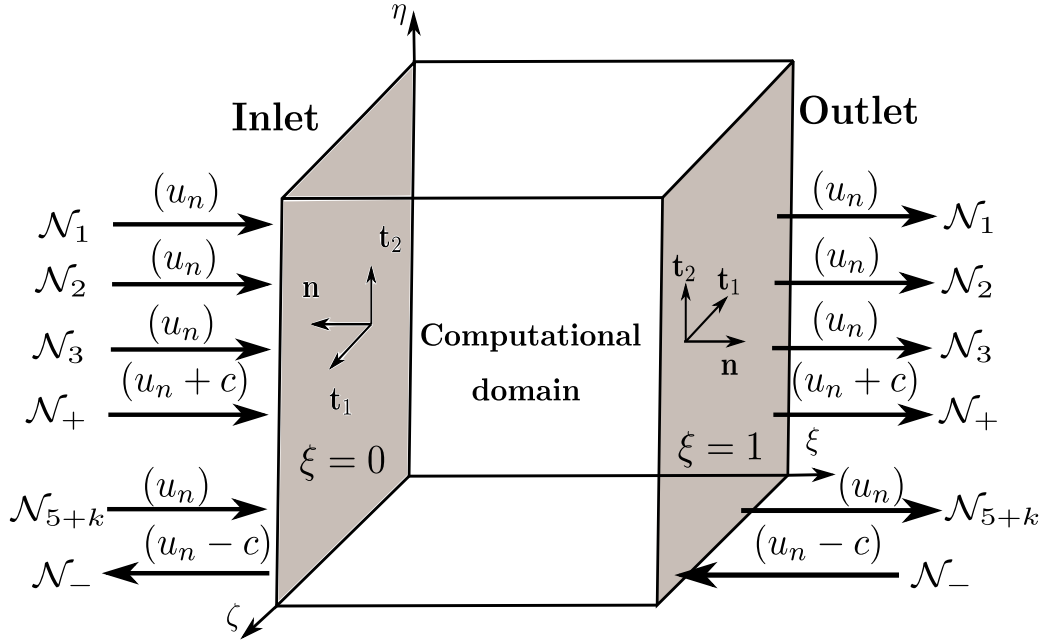


Figure 5.2. – Waves crossing NSCBC inlet and outlet.

5.1.4.1. Subsonic outflow imposing a constant pressure

The aim here is to impose a constant pressure at the outlet boundary based on the waves that cross it. According to Figure 5.2, for a subsonic outflow, $4 + N_s$ characteristic waves are leaving the domain (1 at speed $u_n + c$ and $3 + N_s$ at speed u_n) while 1 characteristic wave is entering the domain at speed $u_n - c$. Consequently, only one thermodynamic relation must be imposed to have a numerically well-behaved boundary condition [231]. Then, the entering wave is specified as [117]:

$$\mathcal{N}_-^* = |J|K_P (P - P^t) + \alpha \mathcal{S}_-^{\text{exact}} - (1 - \alpha) \mathcal{S}_- \quad (5.31)$$

where $K_P [s^{-1}]$ is the pressure relaxation coefficient, P^t and P are respectively the target and current pressure at the boundary and α is a relaxation parameter usually taken as the averaged bulk Mach number over the whole boundary [117, 236]. In Eq. (5.31), $\mathcal{S}_-^{\text{exact}}$ is the exact value of \mathcal{S}_- that can be sometimes obtained using Eq. (5.26) in some analytical test cases where mathematical expressions of $(\partial \hat{\mathbf{F}} / \partial \eta)$ and $(\partial \hat{\mathbf{G}} / \partial \zeta)$ are known [237, 238]. It is always set to zero in this work. Note that with this formulation, the pressure is weakly imposed to limit undesired noise. At this point, only convective fluxes have been treated. According to [231], for a subsonic outflow, diffusive fluxes have to satisfy (in the case of a ξ -normal boundary):

$$\frac{\partial [(\boldsymbol{\tau} \cdot \mathbf{n}^u) \cdot \mathbf{t}_1^u]}{\partial \xi} = 0, \quad \frac{\partial [(\boldsymbol{\tau} \cdot \mathbf{n}^u) \cdot \mathbf{t}_2^u]}{\partial \xi} = 0, \quad \frac{\partial \mathbf{q} \cdot \mathbf{n}^u}{\partial \xi} = 0, \quad \frac{\partial \mathbf{M}_k \cdot \mathbf{n}^u}{\partial \xi} = 0 \text{ for } k = 1, N_s \quad (5.32)$$

where \mathbf{t}_1^u and \mathbf{t}_2^u are two unit tangential vectors in the exit plane, computed from \mathbf{n}^u , whose expressions are given in Section A.3. Derivatives in Eq. (5.32) are imposed following the method described in paragraph 5.1.1. As they concern diffusive fluxes, \mathbf{E}_d , \mathbf{F}_d and \mathbf{G}_d are modified to take into account these conditions before the computation of $\hat{\mathbf{E}}_d$ which is needed to get \mathcal{N} through Eq. (5.25).

5.1.4.2. Subsonic inflow imposing velocities and temperature

The objective here is to impose either time-derivatives or time-constant velocities and temperature at the inlet boundary still based on the waves crossing this boundary. According to Figure 5.2, for

a subsonic inflow, $4 + N_s$ characteristic waves enter the domain while the remaining one exits it. Then, there are now $4 + N_s$ thermodynamic relations to impose for a numerically well-posed boundary condition [3]. Let's say that u , v , w , T and Y_k are prescribed at the inflow boundary. In that case, Eq. (5.24) is multiplied by $P_{\mathbf{Q}}^{-1}$ and after some rearrangements described in Section A.5, the system to solve for the unknown wave amplitudes \mathcal{N}_1^* , \mathcal{N}_2^* , \mathcal{N}_3^* , \mathcal{N}_+^* , \mathcal{N}_{5+k}^* is given by:

$$|J| \frac{\partial u}{\partial t} = \frac{n_z^u c}{\rho} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{n_y^u c}{\rho} (\mathcal{N}_3^* + \mathcal{S}_3) - \frac{n_x^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (5.33)$$

$$|J| \frac{\partial v}{\partial t} = -\frac{n_z^u c}{\rho} (\mathcal{N}_1^* + \mathcal{S}_1) + \frac{n_x^u c}{\rho} (\mathcal{N}_3^* + \mathcal{S}_3) - \frac{n_y^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (5.34)$$

$$|J| \frac{\partial w}{\partial t} = \frac{n_y^u c}{\rho} (\mathcal{N}_1^* + \mathcal{S}_1) - \frac{n_x^u c}{\rho} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{n_z^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (5.35)$$

$$|J| \frac{\partial T}{\partial t} = \frac{T}{\rho} \left[n_x^u (\mathcal{N}_1^* + \mathcal{S}_1) + n_y^u (\mathcal{N}_2^* + \mathcal{S}_2) + n_z^u (\mathcal{N}_3^* + \mathcal{S}_3) \right] - \frac{T(\gamma - 1)}{\sqrt{2}c} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \\ + TW \sum_{k=1}^{N_s} \frac{\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}}{W_k} \quad (5.36)$$

$$|J| \frac{\partial Y_k}{\partial t} = -(\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}) \quad \text{for } k = 1, N_s \quad (5.37)$$

For the velocity components, what is actually imposed are the time derivatives of normal and tangential velocities along the face (in the face referential) defined by:

$$\frac{\partial u_n}{\partial t} = n_x^u \frac{\partial u}{\partial t} + n_y^u \frac{\partial v}{\partial t} + n_z^u \frac{\partial w}{\partial t} \quad (5.38)$$

$$\frac{\partial u_{t1}}{\partial t} = t_{1x}^u \frac{\partial u}{\partial t} + t_{1y}^u \frac{\partial v}{\partial t} + t_{1z}^u \frac{\partial w}{\partial t} \quad (5.39)$$

$$\frac{\partial u_{t2}}{\partial t} = t_{2x}^u \frac{\partial u}{\partial t} + t_{2y}^u \frac{\partial v}{\partial t} + t_{2z}^u \frac{\partial w}{\partial t} \quad (5.40)$$

where $u_{t1/2} = \mathbf{u} \cdot \mathbf{t}_{1/2}^u$ with \mathbf{t}_1^u and \mathbf{t}_2^u the tangential vectors given in Appendix A.3 for the cases $|n_z^u| < 0.7$ and $|n_z^u| \geq 0.7$. Therefore, the derivatives of normal and tangential velocities are linked to the wave amplitudes by:

- Derivative normal to the face:

$$|J| \frac{\partial u_n}{\partial t} = \frac{\mathcal{N}_- + \mathcal{S}_- - \mathcal{N}_+^* - \mathcal{S}_+}{\sqrt{2}} \quad (5.41)$$

- Tangential derivatives if $|n_z^u| < 0.7$:

$$|J| \frac{\partial u_{t1}}{\partial t} = \frac{n_x^u n_z^u c}{\rho \sqrt{(n_x^u)^2 + (n_y^u)^2}} (\mathcal{N}_1^* + \mathcal{S}_1) + \frac{n_y^u n_z^u c}{\rho \sqrt{(n_x^u)^2 + (n_y^u)^2}} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{c \sqrt{(n_x^u)^2 + (n_y^u)^2}}{\rho} (\mathcal{N}_3^* + \mathcal{S}_3) \\ \iff |J| \frac{\partial u_{t1}}{\partial t} = -\frac{c}{\rho} \left[t_{2x}^u (\mathcal{N}_1^* + \mathcal{S}_1) + t_{2y}^u (\mathcal{N}_2^* + \mathcal{S}_2) + t_{2z}^u (\mathcal{N}_3^* + \mathcal{S}_3) \right] \quad (5.42)$$

$$|J| \frac{\partial u_{t2}}{\partial t} = \frac{c}{\rho} \left[\frac{n_y^u (\mathcal{N}_1^* + \mathcal{S}_1) - n_x^u (\mathcal{N}_2^* + \mathcal{S}_2)}{\sqrt{(n_x^u)^2 + (n_y^u)^2}} \right] \quad (5.43)$$

$$\iff |J| \frac{\partial u_{t2}}{\partial t} = \frac{c}{\rho} \left[t_{1x}^u (\mathcal{N}_1^* + \mathcal{S}_1) + t_{1y}^u (\mathcal{N}_2^* + \mathcal{S}_2) \right]$$

- Tangential derivatives if $|n_z^u| \geq 0.7$:

$$\begin{aligned}
 |J| \frac{\partial u_{t1}}{\partial t} &= \frac{c \sqrt{(n_y^u)^2 + (n_z^u)^2}}{\rho} (\mathcal{N}_1^* + \mathcal{S}_1) - \frac{n_x^u n_y^u c}{\rho \sqrt{(n_y^u)^2 + (n_z^u)^2}} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{n_x^u n_z^u c}{\rho \sqrt{(n_y^u)^2 + (n_z^u)^2}} (\mathcal{N}_3^* + \mathcal{S}_3) \\
 \Leftrightarrow |J| \frac{\partial u_{t1}}{\partial t} &= \frac{c}{\rho} \left[t_{2x}^u (\mathcal{N}_1^* + \mathcal{S}_1) + t_{2y}^u (\mathcal{N}_2^* + \mathcal{S}_2) + t_{2z}^u (\mathcal{N}_3^* + \mathcal{S}_3) \right]
 \end{aligned} \tag{5.44}$$

$$\begin{aligned}
 |J| \frac{\partial u_{t2}}{\partial t} &= \frac{c}{\rho} \left[\frac{n_z^u (\mathcal{N}_2^* + \mathcal{S}_2) - n_y^u (\mathcal{N}_3^* + \mathcal{S}_3)}{\sqrt{(n_y^u)^2 + (n_z^u)^2}} \right] \\
 \Leftrightarrow |J| \frac{\partial u_{t2}}{\partial t} &= -\frac{c}{\rho} \left[t_{1y}^u (\mathcal{N}_2^* + \mathcal{S}_2) + t_{1z}^u (\mathcal{N}_3^* + \mathcal{S}_3) \right]
 \end{aligned} \tag{5.45}$$

In the case of time-constant target values at the inlet boundary denoting by:

$$u = u^t, v = v^t, w = w^t, T = T^t \text{ and } Y_k = Y_k^t \text{ for } k = 1, N_s \tag{5.46}$$

a relaxation procedure is applied as in the subsonic outflow condition, so that time derivatives in Eq. (5.33)-Eq. (5.36) are replaced by:

$$\begin{aligned}
 \frac{\partial u_n}{\partial t} &= K_u [\mathbf{u}^t - \mathbf{u}] \cdot \mathbf{n}^u, \quad \frac{\partial u_{t1}}{\partial t} = K_u [\mathbf{u}^t - \mathbf{u}] \cdot \mathbf{t}_1^u, \quad \frac{\partial u_{t2}}{\partial t} = K_u [\mathbf{u}^t - \mathbf{u}] \cdot \mathbf{t}_2^u \\
 \frac{\partial T}{\partial t} &= K_T [T^t - T], \quad \frac{\partial Y_k}{\partial t} = K_{Y_k} [Y_k^t - Y_k] \text{ for } k = 1, N_s
 \end{aligned} \tag{5.47}$$

where $\mathbf{u}^t = (u^t, v^t, w^t)^T$ and K_u, K_T and K_{Y_k} are respectively the relaxation coefficients for velocity components, temperature and species mass fractions. Note that here no condition on diffusive fluxes is imposed as stated in [231, 233].

5.2. Symmetry and no-slip wall boundary conditions for a multi-species non-reacting gas

The objective of this section is to define what are the properties that symmetries and no-slip walls must satisfy and how to set them using SD formalism.

5.2.1. General case

In this paragraph, a symmetry or a wall of normal \mathbf{n} is considered.

5.2.1.1. Symmetry boundary condition for a multi-species non-reacting gas

A symmetry for a multi-species gas is a boundary condition where the following conditions must be fulfilled:

$$u_n = 0, \quad \frac{\partial u_{t1}}{\partial n} = 0, \quad \frac{\partial u_{t2}}{\partial n} = 0, \quad \frac{\partial P}{\partial n} = 0, \quad \frac{\partial T}{\partial n} = 0 \text{ and } \frac{\partial Y_k}{\partial n} = 0 \text{ for } k = 1, N_s \tag{5.48}$$

where $u_{t1/2}$ have been defined in paragraph 5.1.4.2 and $(\partial f / \partial n) \equiv \nabla f \cdot \mathbf{n} = n_x (\partial f / \partial x) + n_y (\partial f / \partial y) + n_z (\partial f / \partial z)$ is the notation for the gradient of any scalar function f (for instance u_{t1} in Eq. (5.48)) projected on \mathbf{n} .

5.2.1.2. No-slip wall boundary conditions for a multi-species non-reacting gas

A no-slip wall for a multi-species gas is a boundary condition where:

$$\mathbf{u} = \mathbf{0} \text{ and } \frac{\partial Y_k}{\partial n} = 0 \text{ for } k = 1, N_s \quad (5.49)$$

Then, two types of no-slip walls are commonly defined depending on the temperature condition:

$$\frac{\partial T}{\partial n} = 0 \text{ (Adiabatic no-slip wall)} \quad (5.50)$$

$$T = T_w \text{ (Isothermal no-slip wall)} \quad (5.51)$$

with T_w a prescribed temperature to be set at the wall. One condition remains to be found for pressure. Let's consider the three dimensional momentum equations taken from Eq. (2.89) projected along \mathbf{n} :

$$\begin{aligned} \frac{\partial(\rho u_n)}{\partial t} + \frac{\partial P}{\partial n} + \frac{\partial(\rho u u_n)}{\partial x} - \rho u^2 \frac{\partial n_x}{\partial x} - \rho u v \frac{\partial n_y}{\partial x} - \rho u w \frac{\partial n_z}{\partial x} \\ + \frac{\partial(\rho v u_n)}{\partial y} - \rho u v \frac{\partial n_x}{\partial y} - \rho v^2 \frac{\partial n_y}{\partial y} - \rho v w \frac{\partial n_z}{\partial y} \\ + \frac{\partial(\rho w u_n)}{\partial z} - \rho u w \frac{\partial n_x}{\partial z} - \rho v w \frac{\partial n_y}{\partial z} - \rho w^2 \frac{\partial n_z}{\partial z} = n_x g_x + n_y g_y + n_z g_z \end{aligned} \quad (5.52)$$

where g_x , g_y and g_z are respectively the viscous fluxes components of x , y and z -momentum. Applying Eq. (5.49) on Eq. (5.52) ends up with:

$$\frac{\partial P}{\partial n} = n_x g_x + n_y g_y + n_z g_z \quad (5.53)$$

Eq. (5.53) shows that for no-slip walls, the normal pressure gradient should balance the projection of viscous fluxes on the normal to the wall. Usually, these viscous fluxes are often neglected and the normal pressure gradient is set to zero at no-slip walls.

5.2.2. How to satisfy a normal gradient condition

It was shown in paragraph 5.2.1 that for symmetry and no-slip walls, normal gradients to the boundary of some quantities as pressure or tangential velocities have to be imposed. As in paragraph 5.2.1.1, let's denote by f any scalar function for which the normal gradient ($\partial f / \partial n$) has to be imposed at the boundary and is then assumed to be known here. The gradient along a normal \mathbf{n} (not necessarily an unit normal) in the physical space is:

$$\begin{aligned} \frac{\partial f}{\partial n} &\equiv n_x \frac{\partial f}{\partial x} + n_y \frac{\partial f}{\partial y} + n_z \frac{\partial f}{\partial z} \\ \frac{\partial f}{\partial n} &= n_x \left(\xi_x \frac{\partial f}{\partial \xi} + \eta_x \frac{\partial f}{\partial \eta} + \zeta_x \frac{\partial f}{\partial \zeta} \right) + n_y \left(\xi_y \frac{\partial f}{\partial \xi} + \eta_y \frac{\partial f}{\partial \eta} + \zeta_y \frac{\partial f}{\partial \zeta} \right) + n_z \left(\xi_z \frac{\partial f}{\partial \xi} + \eta_z \frac{\partial f}{\partial \eta} + \zeta_z \frac{\partial f}{\partial \zeta} \right) \\ \frac{\partial f}{\partial n} &= (n_x \xi_x + n_y \xi_y + n_z \xi_z) \frac{\partial f}{\partial \xi} + (n_x \eta_x + n_y \eta_y + n_z \eta_z) \frac{\partial f}{\partial \eta} + (n_x \zeta_x + n_y \zeta_y + n_z \zeta_z) \frac{\partial f}{\partial \zeta} \end{aligned} \quad (5.54)$$

Taking for example a ξ -normal boundary, \mathbf{n} will be given by Eq. (A.12) so that Eq. (5.54) is now:

$$\frac{\partial f}{\partial n} = A_{FP}^2 \frac{\partial f}{\partial \xi} + (n_x \eta_x + n_y \eta_y + n_z \eta_z) \frac{\partial f}{\partial \eta} + (n_x \zeta_x + n_y \zeta_y + n_z \zeta_z) \frac{\partial f}{\partial \zeta} \quad (5.55)$$

where $A_{FP} = \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}$. Finally, Eq. (5.55) can be expressed for $(\partial f / \partial \xi)$:

$$\frac{\partial f}{\partial \xi} = \frac{1}{A_{FP}^2} \left(\frac{\partial f}{\partial n} - (n_x \eta_x + n_y \eta_y + n_z \eta_z) \frac{\partial f}{\partial \eta} - (n_x \zeta_x + n_y \zeta_y + n_z \zeta_z) \frac{\partial f}{\partial \zeta} \right) \quad (5.56)$$

Since $(\partial f / \partial n)$, the metrics and the derivatives in tangential directions of the boundary are known, the RHS of Eq. (5.56) can be computed. Then, $(\partial f / \partial \xi)$ is known and the methodology explained in paragraph 5.1.1 is used to compute the value of f at the boundary FP.

5.2.3. Practical implementation

In a SD context, a boundary condition is a particular kind of interface, noted I , where only one state, the one from the computational domain, is known. The remaining state is outside the computational domain and has to be determined to satisfy the type of boundary condition. As it will be shown in Section 7.1 for the multi-species case at FP, primitive variables, denoted by $\mathbf{Q} = (T, \mathbf{u}, P, Y_k)^T$ here, are employed for stability reasons. Then during the following reasoning, each boundary condition is prescribed by setting up these primitive variables according to the considered boundary condition needed. There are two different methods to specify a boundary condition due to the fact that only one state is available at boundary interface I . Assuming that the boundary left state \mathbf{Q}_{FP}^L is the known state from the interior domain, the convective fluxes on I can be set with:

- **Method 1 (Weak-Riemann for convective fluxes):** a ghost state defined for the right state at I noted \mathbf{Q}_{FP}^R . Then, the Riemann solver is applied between \mathbf{Q}_{FP}^L and \mathbf{Q}_{FP}^R to get convective fluxes on I . This method was named the Weak-Riemann approach by Mengaldo *et al.* [239].
- **Method 2 (Weak-Prescribed for convective fluxes):** a prescribed value set for both states \mathbf{Q}_{FP}^{BC} such that: $\mathbf{Q}_{FP}^L = \mathbf{Q}_{FP}^R = \mathbf{Q}_{FP}^{BC}$. Consequently, the convective fluxes on I are evaluated using \mathbf{Q}_{FP}^{BC} as the solution vector. This method was named the Weak-Prescribed approach by Mengaldo *et al.* [239].

For diffusive fluxes on I , the same kind of reasoning can be done except that they also need the knowledge of left and right gradients $(\nabla \mathbf{Q})^L$ and $(\nabla \mathbf{Q})^R$. In the same way as convective fluxes, diffusive fluxes on I can be set with:

- **Method 1 (Weak-Riemann for diffusive fluxes):** from the known boundary left state and gradient \mathbf{Q}_{FP}^L and $(\nabla \mathbf{Q})^L$, define a ghost state as for convective fluxes \mathbf{Q}_{FP}^R and an appropriate value of the gradient in this ghost state $(\nabla \mathbf{Q})^R$. Then, compute the interface diffusive flux based on $(\mathbf{Q}_{FP}^L, \mathbf{Q}_{FP}^R, (\nabla \mathbf{Q})^L, (\nabla \mathbf{Q})^R)$ as for interior faces using the strategies explained in Section 4.4.
- **Method 2 (Weak-Prescribed for diffusive fluxes):** a prescribed value for the interface state \mathbf{Q}_{FP}^{BC} used to compute a gradient in the boundary element $(\nabla \mathbf{Q})^{BC}$. Then, compute the interface diffusive flux based on $(\mathbf{Q}_{FP}^{BC}, (\nabla \mathbf{Q})^{BC})$.

In this work, only the Weak-Riemann approach for convective fluxes and the Weak-Prescribed approach for diffusive fluxes was considered. This choice was made after multiple tests using the different methods presented above and appeared to be more robust in most of the simulated cases in this work and especially for the multi-species and reacting cases. It is in accordance with the work of Mengaldo *et al.* [239] for the convective fluxes whereas for the diffusive fluxes these authors also recommend to use a Weak-Riemann approach. However, their paper explained the implementation of boundary conditions in a DG and FR context for the compressible NSE and a mono-species gas. One major difference with the SD method is that DG and FR use a single set of points whereas SD use SP for the interior domain discretization and FP for the boundaries. Taking the example of imposing null velocities at a wall: as SP are strictly inside the computational domain, their velocity is non-zero and the extrapolated velocity from SP to a boundary FP (the left velocity at the boundary) will never be

zero. On the contrary, for DG or FR, this left velocity is directly imposed to zero and used in the numerical scheme. Let's now consider a boundary located at $\xi = 1$ as in Figure 5.1 with its associated convective and diffusive fluxes respectively $\hat{\mathbf{E}}_c^I$ and $\hat{\mathbf{E}}_d^I$.

5.2.3.1. Symmetry boundary condition

From Eq. (5.48), the right state is set as:

$$\mathbf{Q}_{FP}^R = \begin{pmatrix} T_{FP}^R \\ u_{n,FP}^R \\ u_{t1,FP}^R \\ u_{t2,FP}^R \\ P_{FP}^R \\ Y_{k,FP}^R \end{pmatrix} = \begin{pmatrix} T^* \\ 0 \\ u_{t1}^* \\ u_{t2}^* \\ P^* \\ Y_k^* \end{pmatrix} \quad (5.57)$$

where values with superscript $*$ are either:

- computed from \mathbf{Q}_{FP}^L to satisfy Eq. (5.48). For instance, $P^* = P^L$ can be set to respect $\partial P / \partial n = 0$ at the interface FP on the symmetry.
- or computed using Eq. (5.2) where each value of $(\partial f / \partial \xi)$ was found using Eq. (5.56) for each value of $(\partial f / \partial n)$ corresponding to satisfy Eq. (5.48). For instance, the pressure is computed to respect $\partial P / \partial n = 0$ at the interface FP on the symmetry. This was the methodology used throughout this work.

In Eq. (5.57), velocity components are expressed in the $(\mathbf{n}^u, \mathbf{t}_1^u, \mathbf{t}_2^u)$ basis. Thus, to retrieve the cartesian velocity at the boundary FP, the transformation shown in Eq. (5.58) is applied:

$$\begin{aligned} u_{FP}^R &= u_{t1,FP}^R t_{1x}^u + u_{t2,FP}^R t_{2x}^u \\ v_{FP}^R &= u_{t1,FP}^R t_{1y}^u + u_{t2,FP}^R t_{2y}^u \\ w_{FP}^R &= u_{t1,FP}^R t_{1z}^u + u_{t2,FP}^R t_{2z}^u \end{aligned} \quad (5.58)$$

Once left and right states are known, the convective and diffusive fluxes at interface I are computed as follows:

$$\hat{\mathbf{E}}_c^I = \hat{\mathbf{E}}_c^I(\mathbf{Q}_{FP}^L, \mathbf{Q}_{FP}^R) \text{ and } \hat{\mathbf{E}}_d^I = \hat{\mathbf{E}}_d^I(\mathbf{Q}_{FP}^R, (\nabla \mathbf{Q})_{FP}^R) \quad (5.59)$$

with $(\nabla \mathbf{Q})_{FP}^R$ the gradient of each variable in \mathbf{Q}_{FP} computed using the values of \mathbf{Q}_{FP}^R on the symmetry. Therefore, for the diffusive fluxes, the prescribed state at the boundary FP is implicitly the right state introduced in Eq. (5.57): $\mathbf{Q}_{FP}^{BC} = \mathbf{Q}_{FP}^R$.

5.2.3.2. Adiabatic no-slip wall

From Eq. (5.49) and Eq. (5.53) completed by Eq. (5.50), the right state is set as:

$$\mathbf{Q}_{FP}^R = \begin{pmatrix} T_{FP}^R \\ u_{FP}^R \\ v_{FP}^R \\ w_{FP}^R \\ P_{FP}^R \\ Y_{k,FP}^R \end{pmatrix} = \begin{pmatrix} T^* \\ u^* \\ v^* \\ w^* \\ P^* \\ Y_k^* \end{pmatrix} \quad (5.60)$$

where again values with superscript $*$ are either:

- set from \mathbf{Q}_{FP}^L such that $T^* = T_{FP}^L$, $P^* = P_{FP}^L$, $Y_k^* = Y_{k,FP}^L$, $u^* = -u_{FP}^L$, $v^* = -v_{FP}^L$ and $w^* = -w_{FP}^L$. In that case, u^* , v^* and w^* are then set to zero after applying the Riemann solver before computing velocity gradients and using the diffusion scheme to compute $\hat{\mathbf{E}}_d^I$.
- or set from Eq. (5.2) and Eq. (5.56) for T^* , P^* and Y_k^* . For velocities, $u^* = v^* = w^* = 0$ is directly imposed.

$\hat{\mathbf{E}}_c^I$ and $\hat{\mathbf{E}}_d^I$ are also computed using Eq. (5.59). The first choice was suggested in [239] as the *Weak-Riemann-A1* approach in order to have an intermediate Riemann state where the normal component of velocity is zero which is not the case if the second approach is used in Eq. (5.59) for $\hat{\mathbf{E}}_c^I$. Both approaches gave the same results for the cases considered in this work but the second one is preferred here since the pressure at the wall really follows Eq. (5.53).

5.2.3.3. Isothermal no-slip wall

This boundary condition is identical to the adiabatic no-slip wall except that T_{FP}^R is set to the prescribed wall temperature T_w :

$$\mathbf{Q}_{FP}^R = \begin{pmatrix} T_{FP}^R \\ u_{FP}^R \\ v_{FP}^R \\ w_{FP}^R \\ P_{FP}^R \\ Y_{k,FP}^R \end{pmatrix} = \begin{pmatrix} T_w \\ u^* \\ v^* \\ w^* \\ P^* \\ Y_k^* \end{pmatrix} \quad (5.61)$$

with again $\hat{\mathbf{E}}_c^I$ and $\hat{\mathbf{E}}_d^I$ obtained using Eq. (5.59). The two different methodologies, described for the adiabatic wall to set the star values, can be considered.

5.3. Summary of this chapter

In this chapter, an algorithm for the use of NSCBC within the SD method was presented. It is based on a compact formulation of the NSCBC approach written in generalized coordinates, originally derived by Kim and Lee [234, 235], and applied in the SD context by Fievet *et al.* [117]. Their compact formulation is completely equivalent to Kim and Lee algorithm while being less expensive in terms of memory and computational cost. Equations for prescribing subsonic inflow and outflow boundary conditions using the NSCBC technique have been written. Symmetry and no-slip wall boundary conditions were detailed for both convective and diffusive fluxes either under the Weak-Riemann or the Weak-Prescribed formulation. It appeared that for this work, the best results were obtained by using the Weak-Riemann approach for convective fluxes and the Weak-Prescribed approach for diffusive fluxes at FP on a symmetry or on a no-slip wall boundary condition. A last boundary condition needed to simulate turbulent combustion is to have an inlet that correctly reproduces a turbulent flow behavior. This is the topic of the next chapter.

Generation of turbulent inlet conditions in the context of the Spectral Difference method

Overview

| | |
|---|------------|
| 6.1. Turbulent flows in channels and pipes | 103 |
| 6.1.1. Turbulent channel flows | 103 |
| 6.1.2. Turbulent pipe flows | 107 |
| 6.1.3. How to create a turbulent flow in channels and pipes in CFD | 109 |
| 6.2. Time-average velocity profiles imposed at turbulent inlets | 110 |
| 6.2.1. Turbulent channel flows | 110 |
| 6.2.2. Turbulent pipe flows | 111 |
| 6.3. Spectral description of turbulence | 111 |
| 6.3.1. Homogeneous and isotropic turbulence | 112 |
| 6.3.2. Two-point correlation and velocity-spectrum tensors | 112 |
| 6.3.3. The energy spectrum function | 113 |
| 6.3.4. Passot-Pouquet energy spectrum function | 113 |
| 6.3.5. von Karman-Pao energy spectrum function | 114 |
| 6.4. Injection of turbulence using a synthetic random Fourier method | 115 |
| 6.4.1. Different methods to inject turbulence in a computational domain | 115 |
| 6.4.2. The synthetic random Fourier method | 116 |
| 6.4.3. Imposing a turbulent velocity field at a NSCBC inlet | 118 |
| 6.5. Features to help reproduce near-wall turbulence | 118 |
| 6.5.1. Injection of non-homogeneous and anisotropic turbulence | 119 |
| 6.5.2. Artificial tripping | 120 |
| 6.6. Injection of turbulence through a periodic channel | 121 |
| 6.6.1. Presentation of the case and numerical setup | 121 |
| 6.6.2. Validation case | 122 |
| 6.6.3. Influence of the energy spectrum | 123 |
| 6.6.4. Influence of the number of modes and of their distribution | 125 |
| 6.7. Turbulent wall-bounded channel flow | 129 |
| 6.7.1. Numerical setup | 129 |
| 6.7.2. Injection of HIT only at the inlet | 130 |
| 6.7.3. Injection of HIT at the inlet coupled with artificial tripping | 131 |
| 6.7.4. Injection of non-homogeneous and anisotropic turbulence at the inlet | 133 |
| 6.7.5. Axial pressure gradient and acoustics in the channel | 134 |
| 6.8. Turbulent wall-bounded pipe flow | 137 |
| 6.8.1. Numerical setup | 137 |
| 6.8.2. Injection of HIT only at the inlet | 138 |
| 6.8.3. Injection of HIT at the inlet coupled with artificial tripping | 139 |

| | |
|---|------------|
| 6.8.4. Injection of non-homogeneous and anisotropic turbulence at the inlet | 140 |
| 6.8.5. Axial pressure gradient and acoustics in the pipe | 142 |
| 6.9. Summary of this chapter | 144 |

This chapter is dedicated to the simulation of turbulent channels and pipes using the SD method. First, a brief introduction is made to turbulent channel and pipe flows, followed by a spectral characterization of turbulence. Next, the synthetic random Fourier method, whose principles are derived from turbulent wavenumber spectral, is detailed. Two additional treatments helping turbulence injection to reproduce near-wall turbulence are detailed. Finally, three validation test cases are considered: a periodic channel, a turbulent wall-bounded channel and a turbulent wall-bounded pipe.

6.1. Turbulent flows in channels and pipes

The objective of this section is to introduce basic notations and physical phenomena occurring in turbulent channel and pipe flows.

6.1.1. Turbulent channel flows

6.1.1.1. Presentation of the case and governing equations

Turbulent planar channel flows have been widely studied in the CFD community [240–244]. The ideal situation of such flows is presented here to introduce notations and provide some context to the studies presented in this chapter. This situation is typically represented as in Figure 6.1 where a fluid evolves in a long channel ($L/h \gg 1$) with a large aspect ratio ($b/h \gg 1$).

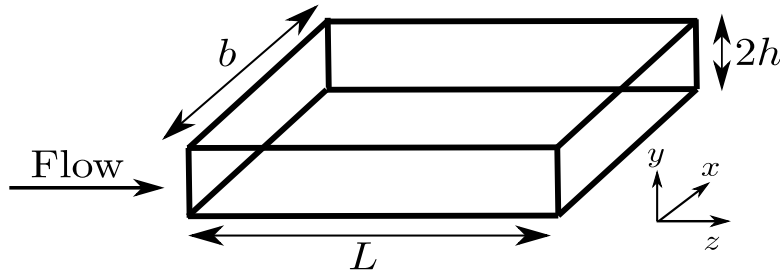


Figure 6.1. – Geometry of a channel flow.

The mean flow is mostly in the axial direction z with the mean velocity varying mainly in the transverse (also called cross-stream) direction y , where bottom and top walls are respectively located at $y = -h$ and $y = +h$. Since the spanwise direction x is very large compared to h , the flow is supposed statistically independent of x [150]. The channel is also assumed to be fully developed so that it is statistically stationary and statistically one-dimensional with velocity statistics only depending on y . The flow is characterized by a bulk Reynolds number based on the bulk velocity U_b , the channel height $2h$ and the kinematic viscosity of the fluid ν :

$$\text{Re}_b = \frac{U_b 2h}{\nu} \quad (6.1)$$

where

$$U_b = \frac{1}{2h} \int_{-h}^h \bar{w}(y) dy \quad (6.2)$$

with \bar{w} the time average of the component of velocity along the z -axis (channel axis). At $\text{Re}_b < 1350$ the flow is laminar, whereas it is fully turbulent at $\text{Re}_b > 1800$ [150]. Mean axial and cross-stream

momentum equations for the turbulent case are:

$$\frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{w}}{\partial y} - \overline{v'w'} \right) - \frac{1}{\rho} \frac{\partial \bar{P}}{\partial z} = 0 \quad (6.3)$$

$$\frac{\partial}{\partial y} \left(\overline{v'^2} \right) + \frac{1}{\rho} \frac{\partial \bar{P}}{\partial y} = 0 \quad (6.4)$$

6.1.1.2. Total shear stress

Integration of Eq. (6.4) between $-h$ and any $y \in]-h, h[$ gives the evolution of the mean pressure:

$$\bar{P}(y, z) = P_w(z) - \overline{\rho v'^2}(y) \quad (6.5)$$

with $P_w(z) \equiv \bar{P}(y = -h, z)$ the mean pressure on the bottom wall. From Eq. (6.5) two remarks can be made:

1. As opposed to the laminar case, the mean pressure is not constant along a plane perpendicular to the wall because of the velocity fluctuations.
2. Taking the partial derivative of this equation with respect to z and using Eq. (6.3) for $(\partial \bar{P} / \partial z)$ gives:

$$\frac{dP_w}{dz}(z) = \frac{\partial \bar{P}}{\partial z} = \rho \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{w}}{\partial y} - \overline{v'w'} \right) \quad (6.6)$$

According to Eq. (6.6), the term $\rho \partial \left(\nu \frac{\partial \bar{w}}{\partial y} - \overline{v'w'} \right) / \partial y$ is only a function of z . However, \bar{w} and $\overline{v'w'}$ are only functions of y (following the hypothesis made in paragraph 6.1.1.1) and therefore $\rho \partial \left(\nu \frac{\partial \bar{w}}{\partial y} - \overline{v'w'} \right) / \partial y$ is also only function of y . Consequently, this term is necessary equal to a constant Π such that:

$$\frac{dP_w}{dz} = \rho \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{w}}{\partial y} - \overline{v'w'} \right) = \Pi \quad (6.7)$$

Eq. (6.7) can be integrated between $-h$ and any $y \in]-h, h[$ to obtain:

$$\rho \left(\nu \left[\frac{\partial \bar{w}}{\partial y}(y) - \left(\frac{\partial \bar{w}}{\partial y} \right)_{y=-h} \right] - \left[\overline{v'w'}(y) - 0 \right] \right) = \Pi(y + h) \quad (6.8)$$

Introducing the total shear stress $\tau(y)$ and the wall shear stress τ_w defined as

$$\tau(y) = \rho \nu \frac{\partial \bar{w}}{\partial y}(y) - \rho \overline{v'w'}(y) \quad (6.9)$$

$$\tau_w = \rho \nu \left(\frac{\partial \bar{w}}{\partial y} \right)_{y=-h} = \tau(-h) \quad (6.10)$$

Eq. (6.8) becomes:

$$\tau(y) = \Pi(y + h) + \tau_w \quad (6.11)$$

In Eq. (6.9), the first and second terms in the right-hand side are respectively called the viscous stress and the Reynolds stress [150]. Finally, because $\tau(y)$ is antisymmetric [150], total shear is zero at channel center ($\tau(y = 0) = 0$) and then from Eq. (6.11) the axial constant pressure gradient is:

$$\Pi \equiv \frac{dP_w}{dz} = \frac{-\tau_w}{h} \quad (6.12)$$

Therefore, the total shear stress expression is:

$$\tau(y) = \frac{-\tau_w y}{h} \quad (6.13)$$

Eq. (6.13) shows that theoretically, the total shear stress is zero along the centerline $y = 0$ of fully turbulent channel flows. As a summary, the pressure drop between the inlet and outlet of the channel drives the flow and in regions where the flow is fully developed, the constant mean pressure gradient Π and the shear-stress gradient $d\tau/dy = -\tau_w/h$ balance each other. An interesting result to notice is that for a given pressure gradient Π and channel half-height h , the total shear profile $\tau(y)$ is obtained through Eq. (6.12) (to compute τ_w) and Eq. (6.13) which is then not dependent on either the fluid properties and whether the flow is laminar or turbulent.

6.1.1.3. Definitions of quantities to characterize friction

A normalized wall shear stress can be introduced, called skin-friction coefficient and noted C_f , from the values of τ_w and U_b :

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho U_b^2} \quad (6.14)$$

Close to the wall, viscous effects dominate so that two quantities can be defined to characterize the flow in this region which are the friction velocity u_τ and the viscous length scale δ_ν :

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \text{ and } \delta_\nu = \frac{\nu}{u_\tau} \quad (6.15)$$

It can be seen that the Reynolds number based on u_τ , δ_ν and ν , noted Re_ν , is actually equal to unity:

$$\text{Re}_\nu = \frac{u_\tau \delta_\nu}{\nu} = \frac{u_\tau}{\nu} \times \frac{\nu}{u_\tau} = 1 \quad (6.16)$$

Additionally, a friction Reynolds number Re_τ based on u_τ , h and ν can also be considered (note that it is based on h and not on $2h$ as for Re_b) [150]:

$$\text{Re}_\tau = \frac{u_\tau h}{\nu} = \frac{h}{\delta_\nu} \quad (6.17)$$

This friction Reynolds number is linked to the bulk Reynolds number using a correlation built from experimental data [150]:

$$\text{Re}_\tau \approx 0.09 \text{Re}_b^{0.88} \quad (6.18)$$

Finally, a last very important variable has to be introduced which is the normal distance from the wall measured in viscous lengths denoted by y^+ :

$$y^+ = \frac{y}{\delta_\nu} = \frac{u_\tau y}{\nu} \quad (6.19)$$

The quantity y^+ is widely employed to describe a mesh close to a wall using the y value of the closest point to this wall: a smaller value of y^+ typically indicates that the mesh is more refined. It actually describes a mesh with regard to the boundary layer and to have a full resolution of it, a mesh must verify $y^+ \leq 1$. It is also used to define different regions of the flows within the channel which are summed up in Table 6.1.

| Name of the region | Location | Description |
|---------------------|----------------------------|---|
| Viscous sublayer | $y^+ < 5$ | Reynolds stress negligible compared to viscous stress |
| Log-law region | $y^+ > 30$ and $y/h < 0.3$ | Region where log-law is valid |
| Buffer layer | $5 < y^+ < 30$ | Region between viscous sublayer and log-law region |
| Viscous wall region | $y^+ < 50$ | Viscous stress has significant impact on total shear |
| Outer layer | $y^+ > 50$ | Viscosity effects negligible on mean flow |

Table 6.1. – Regions of the flow and their characteristics in a turbulent channel based on y^+ values.
Adapted from Table 7.1 in Pope [150].

6.1.1.4. The law of the wall

The law of the wall is the designation of the boundary layer model expressing the axial mean velocity \bar{w} normalized by u_τ , noted u^+ , as a function of y^+ :

$$u^+ = \frac{\bar{w}}{u_\tau} = f_w(y^+) \quad (6.20)$$

Experimental data indicates that for Reynolds values far from the transition limit, the function f_w is universal for channel, pipe and boundary layer flows [150]. For small and large values of y^+ , expressions of f_w can be estimated. For instance, it can be shown that for the viscous sublayer and the log-law region, the law of the wall are respectively [150]:

$$\text{Viscous sublayer } (y^+ < 5) : u^+ = y^+ \quad (6.21)$$

$$\text{Log-law region } (y^+ > 30) : u^+ = \frac{1}{\kappa} \ln(y^+) + B \text{ where } \kappa \approx 0.41 \text{ and } B \approx 5.2. \quad (6.22)$$

6.1.1.5. The velocity-defect law and the overlap region

In the outer layer where $y^+ > 50$, it has been proposed that the walls tend to reduce the centerline velocity $U_{cl} = \bar{w}(0)$ by a process not dependent on viscosity [245]. The velocity defect is then the difference between U_{cl} and the mean velocity \bar{w} . Thus, the velocity-defect law assumes that this velocity defect normalized by u_τ is only a function of y/h [150]:

$$U_{cl}^+ - u^+ = f_D\left(\frac{y}{h}\right) \quad (6.23)$$

where $U_{cl}^+ = U_{cl}/u_\tau$ and subscript D stands for *Defect*. As opposed to the law of the wall, there is no statement saying that f_D is the same for all turbulent flows. However, for small y/h value (typically $y/h < 0.3$), a log-law can be used for f_D [150]:

$$U_{cl}^+ - u^+ = \frac{-1}{\kappa} \ln\left(\frac{y}{h}\right) + B_1 \quad (6.24)$$

with B_1 a flow-dependent constant. Consequently, at large Reynolds number, the log-law defined in Eq. (6.22) and the other log-law of Eq. (6.24) are valid providing that $y^+ > 50$ and $y/h < 0.3$. This was called the overlap region by Millikan [245] and shows that fully turbulent flows cannot be easily characterized by a single velocity profile formula mostly because of this region. This will be discussed further in Section 6.2.

6.1.1.6. Computation of y^+

Usually in CFD to design a mesh it is important to compute its associated y^+ value for given fluid properties ν and given flow conditions characterized by either a bulk Reynolds number Re_b (or a bulk

velocity U_b) or a friction Reynolds number Re_τ . In combustion applications, the value of U_b is an important input parameter since it is linked to the injected fuel mass flow rate. Consequently, to obtain the y^+ of a given mesh for known ν and U_b , the methodology is the following:

1. Use Eq. (6.1) to compute Re_b .
2. Use Eq. (6.18) to compute Re_τ .
3. Use Eq. (6.17) to compute u_τ .
4. Use Eq. (6.15) to compute δ_ν .
5. Use Eq. (6.19) to compute y^+ .

For step 5, a value of y corresponding to the wall normal coordinate of the closest grid point to the wall is needed in Eq. (6.19). This value is denoted y_w here. In SD, y_w is the normal distance of the closest SP to the wall. However, in practice the exact location of SP is not known before starting the first computation. Indeed, SP are placed after applying the isoparametric transformation on a given mesh and using Eq. (4.14) on each element. Thus, to estimate y_w before launching any computation, the element size in the normal direction to the wall Δ_w is divided by the number of SP in the normal direction to the wall estimated as $N_{SP}^{1d} = p + 1$ to get:

$$y_w \approx \frac{\Delta_w}{p + 1} \quad (6.25)$$

Eq. (6.25) is an approximation because it assumes that the $p + 1$ SP are equally distributed along the wall-normal direction which is not true as it was shown in Figures 4.3 and 4.4. For a given p value, the closest SP will be in reality closer than the one obtained for a uniform repartition of SP within an element. Therefore, Eq. (6.25) will always give a value of y_w higher than the exact one and so will be the corresponding estimate of y^+ . This methodology is employed since the value of Δ_w is easily known when building a mesh. A last comment should be made on the axial and spanwise distance from the wall in viscous length units respectively z^+ and x^+ in the situation shown in Figure 6.1. Their values are also needed to characterize a mesh and they are computed by first using Eq. (6.25) with Δ_w being the element size either in the axial (for z^+) or the spanwise (for x^+) direction to the wall and then Eq. (6.19) is employed to get z^+ and x^+ .

6.1.2. Turbulent pipe flows

6.1.2.1. Presentation of the case

Turbulent pipe flows have been also widely studied in CFD [246–248]. The geometry is usually represented as in Figure 6.2 for a pipe of radius R and length $L \gg R$ along z direction.

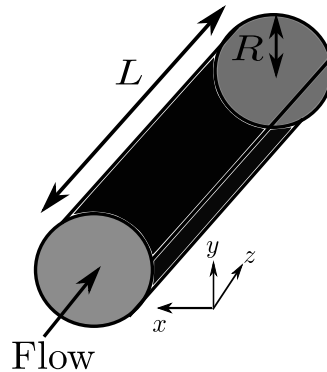


Figure 6.2. – Geometry of a pipe flow.

For a fully developed turbulent pipe, velocity statistics depend only on the radial coordinate r [150]. As for channels, the flow is characterized by a bulk Reynolds number based on bulk velocity U_b , pipe diameter $D = 2R$ and kinematic viscosity of the fluid ν :

$$\text{Re}_b = \frac{U_b D}{\nu} \quad (6.26)$$

where this time

$$U_b = \frac{1}{\pi R^2} \int_0^R \bar{w}(r) 2\pi r dr \quad (6.27)$$

6.1.2.2. Characterization of friction

Friction is traditionally written using a friction factor noted λ :

$$\lambda = \frac{D \Delta P}{\frac{1}{2} \rho U_b^2 L} \quad (6.28)$$

where ΔP is the pressure drop between the inlet and the outlet of the pipe. It can be shown from the momentum equations written in cylindrical coordinates that:

$$\lambda = 4C_f \quad (6.29)$$

$$u_\tau = U_b \sqrt{\frac{\lambda}{8}} \quad (6.30)$$

where C_f and u_τ have been defined respectively in Eq. (6.14) and Eq. (6.15). As mentioned in paragraph 6.1.1.4, from experimental data, the law of the wall f_w is the same for both fully turbulent channels and pipes. Consequently, taking this time $y = R - r$ the normal distance to the walls in a pipe, expressions of f_w in viscous sublayer and log-law regions introduced respectively in Eq. (6.21) and Eq. (6.22) can be employed for pipe flows. It should be noted that according to Pope [150], the values of κ and B , used in the log-law, that best fits the experimental data for pipe flows are $\kappa = 0.436$ and $B = 6.13$.

Finally, for smooth pipes, there is a friction law obtained from experimentation that is valid for all turbulent Reynolds number which implicitly determines λ for a given Re_b value:

$$\frac{1}{\sqrt{\lambda}} = 2 \log_{10} \left(\sqrt{\lambda} \text{Re}_b \right) - 0.8 \quad (6.31)$$

Eq. (6.31) is called the Prandtl's friction law for smooth pipes. Notice that this friction law is very different from the Poiseuille's friction law obtained for fully developed laminar flow where: $\lambda = 64/\text{Re}_b$.

6.1.2.3. Algorithm to compute y^+ on a given configuration

As for channel flows, for given mesh, ν and U_b , the different steps to compute the y^+ are the following:

1. Use Eq. (6.26) to compute Re_b .
2. Solve Eq. (6.31) for λ .
3. Use Eq. (6.30) to compute u_τ .
4. Use Eq. (6.15) to compute δ_ν .
5. Use Eq. (6.19) to compute y^+ (remember that $y = R - r$ in pipe flows).

The wall normal coordinate of the closest SP to the wall is also estimated using Eq. (6.25). In practice when simulating pipe flows, y^+ is named r^+ since, rigorously speaking, the wall normal coordinate for a pipe is along the radial direction. Following the same idea, the spanwise distance from the wall is commonly expressed with a variable $R\theta^+$ based on the length of pipe perimeter:

$$R\theta^+ = \frac{2\pi R / [\Delta_w / (p + 1)]}{\delta_\nu} \quad (6.32)$$

where this time Δ_w is the element size in the azimuthal direction to the wall. Finally, there is no new consideration for the axial distance from the wall compared to a channel flow case because the axial coordinate remains unchanged.

6.1.2.4. How to compute velocities in cylindrical coordinates

For pipe flows, the velocities are very often illustrated using radial, azimuthal and axial velocities respectively denoted by u_r , u_θ and u_z . However, CFD codes generally output velocities in the Cartesian basis: u , v and w . Thus, the link between these two systems of coordinates needs to be written and is given by:

$$\begin{pmatrix} \bar{u}_r \\ \bar{u}_\theta \\ \bar{u}_z \end{pmatrix} = \mathcal{Q} \cdot \begin{pmatrix} \bar{u} \\ \bar{v} \\ \bar{w} \end{pmatrix} \quad (6.33)$$

where \mathcal{Q} is the passage matrix for the rotation of an angle θ in the xy -plane:

$$\mathcal{Q} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (6.34)$$

showing obviously that: $u_z = w$. The velocity correlation tensor R in Cartesian coordinates, noted R^{cart} here for clarity purposes (its expressions are given in Eq. (6.44) and Eq. (6.93)), contains the Reynolds stresses in the Cartesian basis. Thus, to obtain the Reynolds stresses in the cylindrical coordinates system in a new tensor R^{cyl} , the following transformation is applied:

$$R^{cyl} \equiv \begin{pmatrix} \overline{u_r'^2} & \overline{u_r' u_\theta'} & \overline{u_r' u_z'} \\ \overline{u_r' u_\theta'} & \overline{u_\theta'^2} & \overline{u_\theta' u_z'} \\ \overline{u_r' u_z'} & \overline{u_\theta' u_z'} & \overline{u_z'^2} \end{pmatrix} = \mathcal{Q}^T \cdot R^{cart} \cdot \mathcal{Q} \quad (6.35)$$

6.1.3. How to create a turbulent flow in channels and pipes in CFD

Turbulent channels and pipes have been introduced in paragraphs 6.1.1 and 6.1.2 along with their flow characteristics. In a CFD context, it is very important to be able to reproduce such flows but the transition to turbulence is not easy to reach since L must be very large compare to h for channels or R for pipes. In other words, at first sight the computational domain appears to be necessary quite big resulting in a very costly simulation. One way to overcome this issue is by replacing inlet and outlet boundary conditions by periodic boundary conditions coupled with an additional pressure gradient source term to make the flow move from inlet to outlet. This methodology actually reproduces an infinite channel or pipe flow configuration and has been widely employed for DNS of channel [249] and pipe [247, 248] flows. However, it is not very useful when the channel or the pipe is the inlet of a more complex system such as a combustion chamber. In that case, periodic boundary conditions cannot be considered since the outlet is most of the time an outlet pressure boundary condition at the combustion pressure chamber. That is why the turbulence is usually injected at the inlet of the

channel or the pipe with the objective to reproduce a turbulent flow within a reasonable axial length $L \approx 7 - 8 \times 2h$ for channels or $L \approx 7 - 8 \times D$ for pipes.

Actually, the injection of a turbulent velocity field at an inlet, noted \mathbf{u}_{in}^{turb} , is mostly done by injecting a time-average velocity profile $\bar{\mathbf{u}}_{in}$ superimposed by a turbulent fluctuation velocity \mathbf{u}'_{in} coming from the Reynolds decomposition:

$$\mathbf{u}_{in}^{turb} = \bar{\mathbf{u}}_{in} + \mathbf{u}'_{in} \quad (6.36)$$

The formulas used to impose $\bar{\mathbf{u}}_{in}$ for turbulent channel and pipe flows are explained in Section 6.2 whereas the different methods to generate vector \mathbf{u}'_{in} are introduced in Section 6.4.

6.2. Time-average velocity profiles imposed at turbulent inlets

The objective of this section is to explain the formulas used for the time-average velocity profile injected at a turbulent inlet $\bar{\mathbf{u}}_{in}$, also called mean inlet velocity, which was introduced in Eq. (6.36). The situations that will be considered are the turbulent channel and pipe flows presented respectively in paragraphs 6.1.1 and 6.1.2.

6.2.1. Turbulent channel flows

The channel flow situation presented in Figure 6.1 is again considered here. The aim is to find expressions for the components of $\bar{\mathbf{u}}_{in} = (\bar{u}_{in}, \bar{v}_{in}, \bar{w}_{in})^T$ to inject at the inlet of the channel. Firstly, if a fully turbulent flow is injected at the inlet, then the average transverse velocities must be zero (the average flow is purely axial) therefore:

$$\bar{u}_{in} = \bar{v}_{in} = 0 \quad (6.37)$$

It remains now to find an expression for \bar{w}_{in} . In Section 6.1 it was shown that there is no explicit mathematical formula for the axial velocity profile in a fully turbulent channel or pipe. It is mainly due to the lack of modeling of the overlap region. However, at an inlet face, a general formula for the mean velocity profile is still needed to set the mean inlet velocity from its near-wall region to its face center far from walls. Experimental works suggest to consider a power-type law for the axial mean velocity of the form [250–254]:

$$\bar{w}_{in}(y) = U_{cl} \left(1 - \frac{|y|}{h}\right)^{1/n} \quad (6.38)$$

where the exponent $1/n$ depends on Re_b and centerline velocity. In practice it is imposed to maintain the expected bulk velocity at the inlet using:

$$\begin{aligned} U_b &= \frac{\int_{-h}^h U_{cl} \left(1 - \frac{|y|}{h}\right)^{1/n} dy}{2h} = \frac{U_{cl}}{2h} \left(\left[\frac{nh}{n+1} \left(1 + \frac{y}{h}\right)^{1+1/n} \right]_{-h}^0 + \left[\frac{-nh}{n+1} \left(1 - \frac{y}{h}\right)^{1+1/n} \right]_0^h \right) = \frac{nU_{cl}}{n+1} \\ &\iff U_{cl} = \frac{n+1}{n} U_b \end{aligned} \quad (6.39)$$

In Eq. (6.38), for a large range of values of Re_b , the exponent $1/n$ is close to 0.143 or equivalently $n \approx 7$ for most turbulent flows. This exponent $1/n$ actually decreases with increasing Re_b making the profile defined in Eq. (6.38) stiffer close to the wall as it can be seen in Figure 6.3. For the interested reader, a summary of many correlations available for n as a function of Re_b was presented in Afzal *et al.* [254].

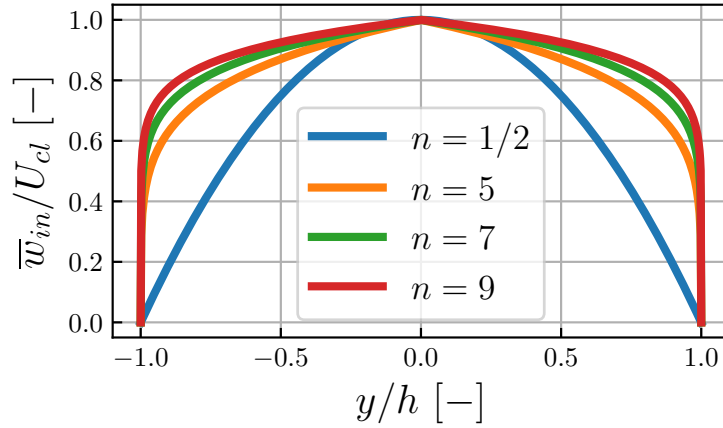


Figure 6.3. – Non-dimensionalized velocity profiles \bar{w}_{in}/U_{cl} for different values of n . The case $n = 1/2$ corresponds to the laminar Poiseuille's flow shown here for comparison.

6.2.2. Turbulent pipe flows

In this paragraph, the pipe flow situation presented in Figure 6.2 is considered and components of mean inlet velocity $\bar{\mathbf{u}}_{in} = (\bar{u}_{in}, \bar{v}_{in}, \bar{w}_{in})^T$ have to be determined. As for the fully turbulent channel flow, the fully turbulent pipe flow has a purely axial flow on average so that Eq. (6.37) has to be verified. A power-type law is also employed:

$$\bar{w}_{in}(r) = U_{cl} \left(1 - \frac{r}{R}\right)^{1/n} \quad (6.40)$$

with still $1/n$ depending on Re_b and U_{cl} set according to the expected U_b following Eq. (6.41):

$$\begin{aligned} U_b &= \frac{\int_0^R U_{cl} \left(1 - \frac{r}{R}\right)^{1/n} 2\pi r dr}{\pi R^2} = \frac{2U_{cl} \left(0 - 0 + \frac{nR}{n+1} \int_0^R \left(1 - \frac{r}{R}\right)^{1+1/n} dr\right)}{R^2} = \frac{2U_{cl}}{R^2} \frac{n^2 R^2}{(n+1)(2n+1)} \\ &\iff U_{cl} = \frac{(n+1)(2n+1)}{2n^2} U_b \end{aligned} \quad (6.41)$$

The estimation of n and U_{cl} can be a little bit different for pipe flows compared to channel flows if White's correlation [255, 256] is used:

$$\frac{U_b}{U_{cl}} = 0.811 + 0.038 [\log_{10}(Re_b) - 4] \quad (6.42)$$

Eq. (6.42) gives a value for the ratio U_b/U_{cl} as a function of Re_b which can be applied in Eq. (6.41) to find the corresponding n value through the solution of the following second-order equation:

$$\frac{U_b}{U_{cl}} = \frac{2n^2}{(n+1)(2n+1)} \iff n^2 \left(\frac{U_b}{U_{cl}} - 1\right) + n3\frac{U_b}{U_{cl}} + \frac{U_b}{U_{cl}} = 0 \iff n = \frac{3\frac{U_b}{U_{cl}} + \sqrt{\frac{U_b}{U_{cl}} \left(\frac{U_b}{U_{cl}} + 8\right)}}{4 \left(1 - \frac{U_b}{U_{cl}}\right)} \quad (6.43)$$

6.3. Spectral description of turbulence

In Section 6.2, expressions for the time-average velocity profiles $\bar{\mathbf{u}}_{in}$ to be imposed at the inlet of turbulent channels and pipes have been described. Thus, it remains to introduce the different

methodologies available to create the vector \mathbf{u}'_{in} in Eq. (6.36). To explain the generation of turbulence fluctuations it is common sense to work with variables defined in the spectral space or Fourier space. The objective of this section is to briefly introduce how turbulent processes can be described in the spectral space and set the hypotheses on which the turbulent generation in Section 6.4 will be done.

6.3.1. Homogeneous and isotropic turbulence

In a fully developed turbulent flow far enough from the boundaries, there exists a region of turbulence where the dynamic quantities can be defined as homogeneous and isotropic. The concept of homogeneous and isotropic turbulence (HIT) is an ideal assumption widely used in the turbulence theory. The homogenous property is defined when all the statistics are independent of the position of the flow so that it is statistically invariant by translations [150]. The isotropic property corresponds to statistics independent of the orientation of the reference domain meaning that they are invariant by rotations and translations of the reference domain [150]. Since the objective is to inject velocity fluctuations with the characteristics of a fully turbulent flow, the HIT assumption will be considered throughout this Section and also in Section 6.4.

6.3.2. Two-point correlation and velocity-spectrum tensors

The two-point correlation tensor is a matrix used to evaluate the influence of the flow at a position \mathbf{x} on the flow at position $\mathbf{x} + \mathbf{r}$ and at time t , defined as:

$$R_{ij}(\mathbf{r}, \mathbf{x}, t) = \overline{u'_i(\mathbf{x}, t) u'_j(\mathbf{x} + \mathbf{r}, t)} \quad (6.44)$$

where u'_i is the fluctuating part of the i -th component of velocity given by:

$$u'_i = u_i - \overline{u_i} \quad (6.45)$$

In the case of homogeneous turbulence, R_{ij} is independent of \mathbf{x} and depends only on the distance between the two points on the flow and still on time:

$$R_{ij}(\mathbf{r}, \mathbf{x}, t) = R_{ij}(\mathbf{r}, t) \text{ for } \mathbf{homogeneous} \text{ turbulence} \quad (6.46)$$

When $\mathbf{r} = \mathbf{0}$, the two-point correlation tensor becomes the autocorrelation tensor $R_{ij}(\mathbf{0}, t)$ from which the turbulent kinetic energy (TKE) at a given point ($\mathbf{r} = \mathbf{0}$) can be extracted by considering one half of its trace:

$$\text{TKE} = \frac{1}{2} R_{ii}(\mathbf{0}, t) = \frac{\overline{u'_1 u'_1} + \overline{u'_2 u'_2} + \overline{u'_3 u'_3}}{2} \quad (6.47)$$

This value of TKE is the same at all points for homogeneous turbulence. In addition, if the turbulence is also isotropic, turbulent velocity fluctuations do not have any directional preference then:

$$\overline{u'_i u'_j} = 0 \text{ for } i \neq j \text{ and } \overline{u'_1 u'_1} = \overline{u'_2 u'_2} = \overline{u'_3 u'_3} = u_{rms}^2 \text{ for } \mathbf{isotropic} \text{ turbulence} \quad (6.48)$$

with u_{rms} (subscript *rms* stands for *root mean square*) the intensity of turbulence in any direction of the flow. Consequently, the two-point correlation tensor under HIT assumptions is the same at all points in all directions of the flow which is translated mathematically as:

$$R_{ij} = u_{rms}^2 \delta_{ij} \text{ for } \mathbf{homogeneous} \text{ AND } \mathbf{isotropic} \text{ turbulence} \quad (6.49)$$

Combining Eq. (6.47) and Eq. (6.48) gives the link between TKE and u_{rms} for a HIT field:

$$\text{TKE} = \frac{3}{2} u_{rms}^2 \iff u_{rms} = \sqrt{\frac{2\text{TKE}}{3}} \quad (6.50)$$

The velocity-spectrum tensor is defined for homogeneous turbulence as the Fourier transform of the two-point correlation tensor [150]:

$$\Phi_{ij}(\boldsymbol{\kappa}, t) = \frac{1}{(2\pi)^3} \int \int_{-\infty}^{\infty} e^{-i\boldsymbol{\kappa} \cdot \mathbf{r}} R_{ij}(\mathbf{r}, t) d\mathbf{r} \quad (6.51)$$

with $\boldsymbol{\kappa} = (\kappa_x, \kappa_y, \kappa_z)^T$ is the wavenumber vector. It is a very important quantity since it represents the Reynolds-stress density in spectral space.

6.3.3. The energy spectrum function

Although Φ_{ij} contains a lot of information because it is a matrix of a vector, it is also hard to manipulate especially for qualitative discussions on turbulence. That is why, a much simpler variable called the energy spectrum function, noted E , is often considered. This function can be viewed as a mass kinetic energy density per wavenumber since its units are $[\text{m}^3 \cdot \text{s}^{-2}] = [\text{m}^2 \cdot \text{s}^{-2} / \text{m}^{-1}]$. It is obtained from Φ_{ij} after getting rid of all the directional information by taking half the trace of Φ_{ij} and integrating over the surface of a sphere \mathcal{S} of radius $\kappa = \|\boldsymbol{\kappa}\|_2$ [150]:

$$E(\kappa) = \oint \frac{1}{2} \Phi_{ii}(\boldsymbol{\kappa}) d\mathcal{S}(\kappa) \quad (6.52)$$

where $\Phi_{ii} = \Phi_{11} + \Phi_{22} + \Phi_{33}$ is the trace of Φ_{ij} . As shown in Eq. (6.52), the energy spectrum function is simpler than Φ_{ij} since it is a scalar function of a scalar. One of the biggest interest of using $E(\kappa)$ is that for a HIT field, all the information characterizing the turbulent structures is contained in $E(\kappa)$. In other words, the knowledge of the energy spectrum function determines the kind of HIT field generated. Indeed, it is possible to show under HIT assumptions that:

$$\text{TKE} = \int_0^\infty E(\kappa) d\kappa \quad (6.53)$$

$$\epsilon = 2\nu \int_0^\infty \kappa^2 E(\kappa) d\kappa \quad (6.54)$$

$$L_f = \frac{\pi}{2u_{rms}^2} \int_0^\infty \frac{E(\kappa)}{\kappa} d\kappa \quad (6.55)$$

where ϵ and L_f are respectively the dissipation rate of turbulence and the longitudinal integral length scale of the HIT field. Finally, given the quantities introduced in Eq. (6.53), Eq. (6.54) and Eq. (6.55), two other important quantities can be deduced:

$$\tau_\epsilon = \frac{L_f}{u_{rms}} \quad (6.56)$$

$$Re_t = \frac{u_{rms} L_f}{\nu} \quad (6.57)$$

with τ_ϵ and Re_t respectively the large eddies turnover time and the turbulent Reynolds number. The next two paragraphs are dedicated to two examples of energy spectrum functions: the Passot-Pouquet spectrum [257] and the von Karman-Pao spectrum [258].

6.3.4. Passot-Pouquet energy spectrum function

Given a turbulent intensity u_{rms} and a characteristic length noted L_e , the Passot-Pouquet (PP) energy spectrum is defined by [257]:

$$E(\kappa) = A \left(\frac{\kappa}{\kappa_{e,PP}} \right)^4 \exp \left(-2 \left(\frac{\kappa}{\kappa_{e,PP}} \right)^2 \right) \quad (6.58)$$

with $\kappa_{e,PP} = 2\pi/L_e$ and the constant A is obtained by writing that $E(\kappa)$ must satisfy:

$$\int_0^\infty E(\kappa) d\kappa = \frac{3}{2} u_{rms}^2 \iff A = \frac{16 u_{rms}^2}{\kappa_{e,PP}} \sqrt{\frac{2}{\pi}} \quad (6.59)$$

According to Eq. (6.58), $\kappa_{e,PP}$ is the wavenumber at which $E(\kappa)$ is maximum which means that L_e corresponds to the size of the most energetic structures generated with the PP spectrum. Using Eq. (6.54) and Eq. (6.55) with $E(\kappa)$ given by Eq. (6.58), the dissipation rate and the integral length scale for the PP spectrum are:

$$\epsilon_{PP} = \frac{15}{4} \nu u_{rms}^2 \kappa_{e,PP}^2 \quad (6.60)$$

$$L_{f,PP} = \frac{\sqrt{2\pi}}{\kappa_{e,PP}} \quad (6.61)$$

This spectrum correctly describes the large eddies (small κ values) of turbulence but the small eddies (large κ values) are not represented. Consequently, it should be used when the Reynolds number is small (not small enough to be laminar, but small compared to a highly turbulent flow).

6.3.5. von Karman-Pao energy spectrum function

As for the PP spectrum, with a turbulent intensity u_{rms} and a characteristic length noted L_e , the von Karman-Pao (VKP) energy spectrum is defined by:

$$E(\kappa) = \alpha \frac{u_{rms}^2}{\kappa_{e,VKP}} \frac{(\kappa/\kappa_{e,VKP})^4}{(1 + (\kappa/\kappa_{e,VKP})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] \quad (6.62)$$

where α , $\kappa_{e,VKP}$ and κ_{Kol} are three constants determined hereafter. The constant α is obtained by writing that:

$$\int_0^\infty E(\kappa) d\kappa = \frac{3}{2} u_{rms}^2 \iff \alpha = \frac{55\Gamma(5/6)}{9\sqrt{\pi}\Gamma(1/3)} \approx 1.4527621122109746 \quad (6.63)$$

$\left(\frac{\kappa_{Kol}}{\kappa_{e,VKP}} \right)^2 \rightarrow \infty$

where Γ stands for the gamma function [259]. As highlighted by Eq. (6.63), this value of α such that the integral of the spectrum equals the TKE, is only valid if the ratio $(\kappa_{Kol}/\kappa_{e,VKP})^2$ is big. For $\kappa_{e,VKP}$, it has an expression that satisfies:

$$L_e = \frac{\pi}{2u_{rms}^2} \int_0^\infty \frac{E(\kappa)}{\kappa} d\kappa \iff \kappa_{e,VKP} = \frac{9\pi\alpha}{55L_e} \quad (6.64)$$

$\left(\frac{\kappa_{Kol}}{\kappa_{e,VKP}} \right)^2 \rightarrow \infty$

Thus, for the VKP spectrum, the input characteristic length L_e is actually the longitudinal integral length scale L_f . Finally, κ_{Kol} is obtained by solving the following system for κ_{Kol} and ϵ :

$$\kappa_{Kol} = \left(\frac{\epsilon}{\nu^3} \right)^{1/4} \quad (6.65)$$

$$\epsilon = 2\nu \int_0^\infty \kappa^2 E(\kappa) d\kappa \quad (6.66)$$

Combining Eq. (6.65) and Eq. (6.66) in the limit $(\kappa_{Kol}/\kappa_{e,VKP})^2 \rightarrow \infty$ gives an expression for ϵ_{VKP} :

$$\epsilon_{VKP} = [\alpha\Gamma(2/3)]^{3/2} \frac{\kappa_{e,VKP}}{2} u_{rms}^3 \quad (6.67)$$

and κ_{Kol} is deduced from Eq. (6.65). It should be mentioned that very often, ϵ_{VKP} is simply computed by:

$$\epsilon_{VKP} = \frac{u_{rms}^3}{L_e} \quad (6.68)$$

It should also be mentioned that the maximum energy occurs at the wave number $\sqrt{12/5}\kappa_{e,VKP}$.

The VKP spectrum can represent both the large and small turbulent structures. From a numerical point of view it is a more realistic spectrum than the one from Passot and Pouquet but it needs a better mesh resolution to capture the small scales. Table 6.2 sums up the main turbulence variables expressions for these two turbulence spectrums. Moreover, the demonstrations of Eq. (6.63), Eq. (6.64) and Eq. (6.67) are given in Appendix G.

| Spectrum | κ_e [m^{-1}] | ϵ [$\text{m}^2.\text{s}^{-3}$] | L_f [m] |
|----------------|--------------------------------|---|--------------------------------|
| Passot-Pouquet | $\frac{2\pi}{L_e}$ | $\frac{15}{4}\nu u_{rms}^2 k_e^2$ | $\frac{\sqrt{2\pi}}{\kappa_e}$ |
| von Karman-Pao | $\frac{9\pi\alpha}{55L_e}$ | $(\alpha\Gamma(2/3))^{3/2} \frac{\kappa_e}{2} u_{rms}^3$ or $\frac{u_{rms}^3}{L_e}$ | L_e |

Table 6.2. – Summary of the values taken by κ_e , ϵ and L_f for both Passot-Pouquet and von Karman-Pao spectrums.

6.4. Injection of turbulence using a synthetic random Fourier method

The objective of this section is to describe the approach used in this work to generate turbulent velocity fluctuations \mathbf{u}'_{in} at the inlet of a computational domain.

6.4.1. Different methods to inject turbulence in a computational domain

The injection of turbulence in a computational domain is a vast topic of research and plenty of approaches exist in the literature. The classification of these different methods depend on the author and some nice reviews are the ones of Tabor *et al.* [260], Wu [261] and Dhamankar *et al.* [262]. Following the categorization of Wu [261], two main families of methods can be considered:

- Recycling methods containing:
 - ★ The strong recycling method where a turbulent flow originating from a former simulation is injected at the domain inlet.
 - ★ The weak recycling method where the outgoing flow is reintroduced at the inflow boundary [261].
- Synthetic methods containing:
 - ★ The synthetic random Fourier method where turbulent fluctuations are injected by assimilating them to a sum of random Fourier modes [263, 264].
 - ★ The digital filtering method where a random white-noise signal is first generated and then filtered in a particular way to create spatial and temporal correlation on the original signal [265].
 - ★ The synthetic eddy method where a turbulent flow is represented through a distribution of convected randomly turbulent spots whose size and turbulent intensity vary in space and time [266].

- ★ The volume forcing method where several planes located downstream of the inlet will add a body force term in the NSE to accelerate the transition to turbulence [267, 268].

This work is focused on combustion applications where channels or pipes are very often the injector part of a combustion system. Thus, recycling methods were not adapted in that case. Indeed, the strong recycling would have been too costly and for the weak recycling method the flow at the outlet of the injector is not supposed to be re-injected at the inlet. Concerning synthetic methods, the digital and the synthetic eddy methods, although they are widely used, are more complicated to implement in a CFD code compared to the synthetic random Fourier method and still employed in combustion simulations. Thus, a synthetic random Fourier method was implemented during this work in order to be used on turbulent combustion applications after being validated on non-reactive turbulent cases which is the purpose of Sections 6.6, 6.7 and 6.8. Finally, the volume forcing method was not employed in this work, but a methodology to artificially reproduce roughness in a channel or pipe flow was tried to also accelerate the development of turbulence. It is detailed in paragraph 6.5.2.

6.4.2. The synthetic random Fourier method

Originally introduced by Kraichnan [263] for the diffusion of particles in random velocity fields, a divergence-free synthetic HIT field can be generated based on the summation of N Fourier modes, each characterized by a wavenumber vector $\boldsymbol{\kappa}_n$ with $n \in \llbracket 1, N \rrbracket$. In Kraichnan's method, the components of $\boldsymbol{\kappa}_n$ are chosen stochastically and indirectly to satisfy a given energy spectrum. For instance, in order to have a PP spectrum as in Eq. (6.58), each component of $\boldsymbol{\kappa}_n$ has to be picked from a Gaussian distribution of standard deviation $\kappa_e/2$. However, this methodology is not really practical since if another energy spectrum is desired by the user, the corresponding stochastic distribution must be found to compute vectors $\boldsymbol{\kappa}_n$. Consequently, it is more useful to have a way to obtain $\boldsymbol{\kappa}_n$ components deterministically, that is with an explicit mathematical formula, from a given energy spectrum.

Thus, following the works of Karweit *et al.* [269] and Bechara *et al.* [264], a turbulent velocity vector on a given point \mathbf{x} can be computed using Eq. (6.69):

$$\mathbf{u}'_{in}(\mathbf{x}) = 2 \sum_{n=1}^N u_{tn} \cos(\boldsymbol{\kappa}_n \cdot \mathbf{x} + \psi_n) \boldsymbol{\sigma}_n \quad (6.69)$$

where u_{tn} , ψ_n and $\boldsymbol{\sigma}_n$ are respectively the amplitude, the phase and the direction (it is a unit vector) of the n -th Fourier mode associated with the wavenumber vector $\boldsymbol{\kappa}_n$.

6.4.2.1. Determination of u_{tn}

The amplitude u_{tn} is determined from the turbulent kinetic energy spectrum E , chosen by the user, using Eq. (6.70):

$$u_{tn} = \sqrt{E(\kappa_n) \Delta\kappa_n} \text{ for } n \in \llbracket 1, N \rrbracket \quad (6.70)$$

with $\kappa_n = \|\boldsymbol{\kappa}_n\|_2$ and $\Delta\kappa_n$ being the difference between two consecutive modes which could be linear (Eq. (6.71)) or logarithmic (Eq. (6.72)) so that $\forall n \in \llbracket 1, N \rrbracket$:

$$\Delta\kappa_n = \frac{\kappa_N - \kappa_1}{N - 1} \equiv \Delta\kappa_{lin} \text{ with } \kappa_n = \kappa_1 + (n - 1) \Delta\kappa_{lin} \quad (6.71)$$

$$\Delta\kappa_n = \kappa_{n+1} - \kappa_n \text{ with } \kappa_n = \exp(\ln(\kappa_1) + (n - 1) \Delta\kappa_{log}) \text{ and } \Delta\kappa_{log} = \frac{\ln(\kappa_N) - \ln(\kappa_1)}{N - 1} \quad (6.72)$$

The logarithmic distribution will give a better discretization of the spectrum in the lower wavenumber range associated with the larger energy containing eddies. A nice property satisfied by the u_{tn} terms, to check if the generation process gave the expected HIT field, is [264]:

$$\sum_{n=1}^N u_{tn}^2 = \text{TKE} = \frac{3}{2} u_{rms}^2 \quad (6.73)$$

6.4.2.2. Determination of κ_n components, ψ_n and σ_n

The turbulent velocity field should satisfy the incompressibility condition for each mode:

$$\nabla \cdot \mathbf{u}'_{in} = 0 \iff \kappa_n \cdot \sigma_n = 0 \text{ for } n \in \llbracket 1, N \rrbracket \quad (6.74)$$

meaning that κ_n and σ_n are always perpendicular. The wavenumber vector κ_n is described using its spherical coordinates $(\kappa_n, \phi_n, \theta_n)$ and the unit vector σ_n is determined by its polar angle α_n lying in the (κ_1, κ_2) plane as shown in Figure 6.4. The HIT field is obtained by appropriately choosing

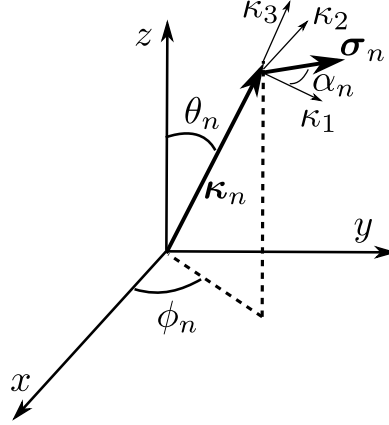


Figure 6.4. – Geometry of the n -th Fourier mode.

probability density functions for the four random angles ϕ_n , θ_n , ψ_n and α_n . It can be shown that ϕ_n , α_n and ψ_n should be uniform densities such that [261, 269]:

$$\mathcal{P}(\phi_n) = \mathcal{P}(\alpha_n) = \mathcal{P}(\psi_n) = \frac{1}{2\pi} \quad (6.75)$$

and a sine function should be used for θ_n :

$$\mathcal{P}(\theta_n) = \frac{\sin(\theta_n)}{2} \quad (6.76)$$

Choosing a random number following a uniform probability density can be done using free software such as the one proposed by Takuji Nishimura and Makoto Matsumoto [270] employed in this work. This software will give a random number in $[0, 1]$ following a uniform probability density and then this number is simply multiplied by 2π to get the appropriated ϕ_n , α_n and ψ_n values. However, to obtain a sine probability density for θ_n , an additional treatment is required. Using the so-called inverse method [271], θ_n can be generated with Eq. (6.77) [272, 273]:

$$\theta_n = \cos^{-1}(2a - 1) \quad (6.77)$$

where a follows a uniform density on $[0, 1]$ also obtained with the generator available in [270]. Once ϕ_n and θ_n have been obtained, κ_n components in Cartesian coordinates are given by:

$$\kappa_n = \kappa_n \begin{pmatrix} \cos(\phi_n) \sin(\theta_n) \\ \sin(\phi_n) \sin(\theta_n) \\ \cos(\theta_n) \end{pmatrix} \quad (6.78)$$

with κ_n already computed either through Eq. (6.71) or Eq. (6.72). For σ_n , in practice α_n is not computed and an intermediate unit vector ζ_n , whose Cartesian components are randomly generated from a uniform density on $[-1, 1]$, is employed instead such that:

$$\sigma_n = \frac{\zeta_n \times \kappa_n}{\|\zeta_n \times \kappa_n\|_2} \quad (6.79)$$

It is worth mentioning that in Saad *et al.* [272], ζ_n is generated using its spherical coordinates selected randomly with the same methodology as for κ_n components. These two methods must be equivalent since ζ_n is only an intermediate vector picked randomly on a sphere of radius unity and used to ensure that σ_n satisfies Eq. (6.74) which is ensured by Eq. (6.79). Vector σ_n therefore has a random polar angle α_n lying in $[0, 2\pi]$ and not an explicit random value determined from Eq. (6.75). The different steps to generate the turbulent velocity field \mathbf{u}'_{in} on a given point \mathbf{x} using this methodology are summed up in Algorithm 6 of Section B.4.

6.4.2.3. Time-dependent synthetic HIT field

For turbulent injection through a boundary, a time dependence has to be introduced in Eq. (6.69). It was done by Bailly and Juvé [274] who used a constant (in time and space) velocity vector \mathbf{u}_c at which the turbulent field is convected from an imaginary box outside of the computational domain so that Eq. (6.69) becomes:

$$\mathbf{u}'_{in}(\mathbf{x}, t) = 2 \sum_{n=1}^N u_{tn} \cos(\kappa_n \cdot (\mathbf{x} - t\mathbf{u}_c) + \psi_n) \sigma_n \quad (6.80)$$

where t is the time. The formulation given by Eq. (6.80) corresponds to Eq. (7) in Bailly and Juvé [274] with ω_n set to zero and is the formula employed in this work. In practice, $\|\mathbf{u}_c\|_2 = U_b$ and is in the direction of the channel or pipe axis. Since ERK time integration methods are employed, Eq. (6.80) is evaluated at each RK stage. Thus, the time t at each RK stage q is needed noted t^q . It is computed from the values of $\mathbf{c} = [c_q]_{1 \leq q \leq Q}^T$ in the Butcher tableau of a RK method:

$$t^q = t^n + c_q (\Delta t)^n \quad (6.81)$$

where t^n and $(\Delta t)^n$ are respectively the time and timestep at instant n .

6.4.3. Imposing a turbulent velocity field at a NSCBC inlet

In paragraph 5.1.4.2, the procedure to impose a subsonic inflow with time-constant velocities was described. However, the objective is now to impose at this NSCBC inlet, the turbulent velocity defined in Eq. (6.36) which is necessarily time-dependent because of $\mathbf{u}'_{in}(\mathbf{x}, t)$. The time derivatives in Eq. (5.33)-Eq. (5.35) can be computed since the time derivative of Eq. (6.36) is given by:

$$\frac{d\mathbf{u}'_{in}}{dt}(\mathbf{x}, t) = \mathbf{0} + \frac{d\mathbf{u}'_{in}}{dt}(\mathbf{x}, t) = 2 \sum_{n=1}^N u_{tn} (\kappa_n \cdot \mathbf{u}_c) \sin(\kappa_n \cdot (\mathbf{x} - t\mathbf{u}_c) + \psi_n) \sigma_n \quad (6.82)$$

and technically the system of equations Eq. (5.33)-Eq. (5.36) can be solved for \mathcal{N}_1^* , \mathcal{N}_2^* , \mathcal{N}_3^* and \mathcal{N}_+^* . However, this turbulent signal must also impose a given mean value, defined by $\bar{\mathbf{u}}_{in}$, at the boundary. In these situations, specifying only the time derivatives will not be sufficient for the CFD code to keep the expected mean value. Thus, a relaxation procedure has to be added to the time derivatives as in [275, 276] so that:

$$\frac{d\mathbf{u}'_{in}}{dt} \text{ is replaced by } \frac{d\mathbf{u}'_{in}}{dt} + K_u [(\bar{\mathbf{u}}_{in} + \mathbf{u}'_{in}) - \mathbf{u}] \quad (6.83)$$

6.5. Features to help reproduce near-wall turbulence

The turbulent velocity introduced in paragraph 6.4.2 corresponds to a divergence-free HIT field. However, for fully turbulent flows the homogeneous and isotropic properties of turbulence are only verified far from the boundaries of the domain and especially far from the walls. Indeed, in near-wall

regions, the cross-correlations $\overline{u'_i u'_j}$ (with $i \neq j$) are not zero. Thus, if a turbulent field generated using Eq. (6.80) is imposed at the inlet of a channel or a pipe, it will technically not be sufficient to reproduce near-wall turbulence in this channel or pipe. The objective of this section is to show some available features that can be added to the generation of the HIT field described in paragraph 6.4.2 in order to better reproduce near-wall turbulence.

6.5.1. Injection of non-homogeneous and anisotropic turbulence

It is actually possible to transform a HIT field, generated by any method described in paragraph 6.4.1, into a non-homogeneous and anisotropic turbulent field provided that the velocity correlation tensor R (defined in Eq. (6.44)) of the turbulent flow field is known. Let's denote by $\mathbf{u}'_{in}{}^{\text{HIT}}$, a turbulent HIT field with a null mean velocity and a unit RMS velocity: $u_{rms} = 1 \text{ m.s}^{-1}$. A non-homogeneous and anisotropic turbulent field can be generated from $\mathbf{u}'_{in}{}^{\text{HIT}}$ by considering the scaling:

$$\mathbf{u}'_{in}{}^* = A \cdot \mathbf{u}'_{in}{}^{\text{HIT}} \quad (6.84)$$

where A is the Cholesky decomposition of R given by [277]:

$$A = \begin{pmatrix} (R_{11})^{1/2} & 0 & 0 \\ \frac{R_{21}}{(R_{11})^{1/2}} & (R_{22} - a_{21}^2)^{1/2} & 0 \\ \frac{R_{31}}{(R_{11})^{1/2}} & \frac{(R_{32} - a_{21}a_{31})}{a_{22}} & (R_{33} - a_{31}^2 - a_{32}^2)^{1/2} \end{pmatrix} \quad (6.85)$$

This methodology was successfully applied by Smirnov and Celik [278] on a HIT field generated using the original Kraichnan's method. Later on, Billson *et al.* [279, 280] and Shur *et al.* [281] used it for a HIT field obtained with Karweit and Bechara methodology explained in paragraph 6.4.2. In this work, the computation of $\mathbf{u}'_{in}{}^{\text{HIT}}$ from \mathbf{u}'_{in} introduced in Eq. (6.69) is done following the idea of Shur *et al.* [281]:

$$\mathbf{u}'_{in}{}^{\text{HIT}}(\mathbf{x}) \equiv 2 \sum_{n=1}^N u_{tn}^{\text{HIT}} \cos(\boldsymbol{\kappa}_n \cdot \mathbf{x} + \psi_n) \boldsymbol{\sigma}_n = \frac{\mathbf{u}'_{in}(\mathbf{x})}{\sqrt{\frac{2}{3} \sum_{m=1}^N u_{tm}^2}} = 2 \sum_{n=1}^N \frac{u_{tn}}{\sqrt{\frac{2}{3} \sum_{m=1}^N u_{tm}^2}} \cos(\boldsymbol{\kappa}_n \cdot \mathbf{x} + \psi_n) \boldsymbol{\sigma}_n \quad (6.86)$$

so that each mode amplitude u_{tn} , originally generated for any value of u_{rms} , is divided by the effective value of u_{rms} noted u_{rms}^{eff} and equal to $\sqrt{(2/3) \sum_{m=1}^N u_{tm}^2}$. It can be checked that by construction

$$\sum_{n=1}^N \left(u_{tn} / u_{rms}^{eff} \right)^2 = 3/2 \text{ (from Eq. (6.73) since } \mathbf{u}'_{in}{}^{\text{HIT}} \text{ must have } u_{rms} = 1 \text{):}$$

$$\sum_{n=1}^N \left(\frac{u_{tn}}{u_{rms}^{eff}} \right)^2 = \frac{3}{2} \frac{\sum_{n=1}^N u_{tn}^2}{\sum_{m=1}^N u_{tm}^2} = \frac{3}{2} \quad (6.87)$$

In the case of turbulent injection where a time-dependent synthetic HIT field is considered (see paragraph 6.4.2.3), $\mathbf{u}'_{in}{}^{\text{HIT}}$ is computed at each time instant based on Eq. (6.80) with $u_{tn} = u_{tn}^{\text{HIT}}$ and Eq. (6.84) is used to rescale the turbulent field at the given time instant. Hopefully, the Cholesky matrix A is not time-dependent so that it can be computed and stored once in pre-processing at each inlet FP.

This scaling approach is very helpful to inject realistic turbulence at inlet flows reducing the length of a channel or a pipe after which the flow is fully turbulent. However, it requires the knowledge of R usually obtained from experimental or DNS data made at the same Reynolds number as the one of the considered inlet.

6.5.2. Artificial tripping

Another way to help reproduce near-wall turbulence, as in paragraph 6.5.1, is by using a roughness element (also called step) at the wall of channels or pipes to trip the transition to turbulence. This so-called tripping method can be either:

1. directly set in the channel or pipe geometry and treated as a wall boundary condition. However, this methodology is intrusive and it was decided to not use it during this work.
2. reproduced artificially using a source term added to the momentum equations. It is named artificial tripping and is the methodology that was considered in this work.

The artificially tripping approach was proposed by Boudet *et al.* [282] for a Cartesian geometry like flat plate or channel flows as shown in Figure 6.5a for a step of lengths L_x , L_y and L_z respectively in x , y and z directions. The source term is added in the axial momentum equation (z -axis in that case) on the region defined by the step with the following formula:

$$\mathbf{S}_{trip} = \frac{-1}{2L_z} \rho C_D w |w| \mathbf{e}_z \quad (6.88)$$

where C_D is a user-specified drag coefficient and w is the local velocity along z -axis. It will induce a drag force in the axial direction of the form:

$$\mathbf{F}_{trip} = \frac{1}{2} \rho L_x L_y C_D w |w| \mathbf{e}_z \quad (6.89)$$

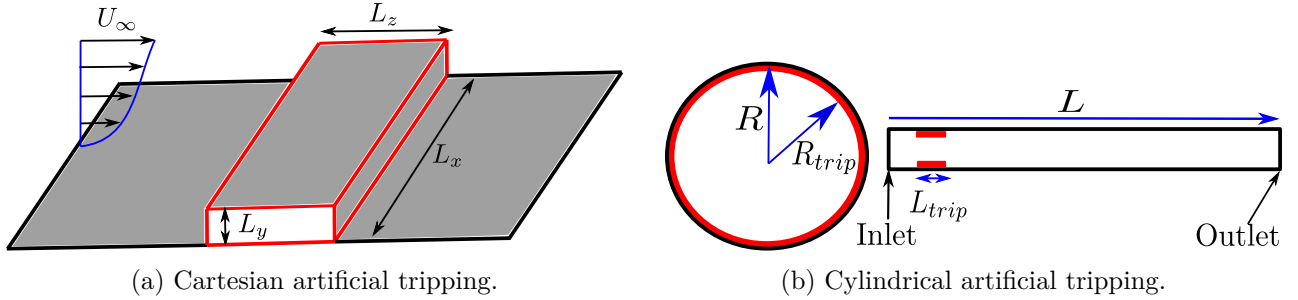


Figure 6.5. – Artificial tripping using a source term added to momentum equations.

During this work, an artificial tripping adapted to cylindrical geometries was developed. It is shown in Figure 6.5b for an artificial annular step of length L_{trip} and radius R_{trip} . In that case, the source term and the induced drag, added to the axial momentum equation on the region defined by the annular step, are given by:

$$\mathbf{S}_{trip} = \frac{-1}{2L_{trip}} \rho C_D w |w| \mathbf{e}_z \text{ and } \mathbf{F}_{trip} = \frac{1}{2} \rho \pi (R^2 - R_{trip}^2) C_D w |w| \mathbf{e}_z \quad (6.90)$$

This artificial tripping method is very appealing for three main reasons:

1. It is not intrusive so that the geometry does not need to be modified.
2. No boundary conditions have to be specified on the step.
3. It is neither computationally expensive nor difficult to implement since the source term expression is very simple.

Nevertheless, the location, the size and the value of C_D need some preliminary tests which are configuration-dependent. Some guidelines and good practices to set these parameters will be detailed in paragraph 6.7.3 for the Cartesian artificial tripping and in paragraph 6.8.3 for the Cylindrical artificial tripping.

6.6. Injection of turbulence through a periodic channel

The objective of this section is to validate the injection of turbulence explained in Section 6.4 and to see the influence of the different turbulent parameters on the results. In particular, the influence of the energy spectrum and the number of turbulent modes along with their distribution (linear or logarithmic) will be investigated in terms of accuracy and computational cost.

6.6.1. Presentation of the case and numerical setup

The computational domain is rectangular with turbulent injection using NSCBC principle at $z = 0$, subsonic outflow at constant pressure $P_{out} = 10^5$ Pa at $z = 4$ mm and periodic boundary conditions elsewhere as shown in Figure 6.6. The configuration shown in Figure 6.6 is an ideal case to study

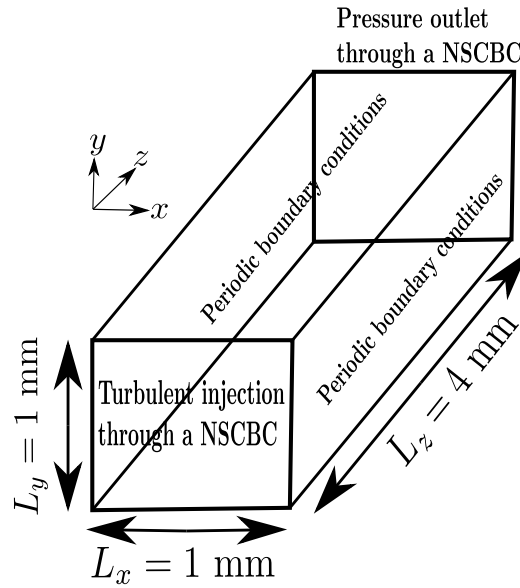


Figure 6.6. – Computational domain and boundary conditions for the injection of turbulence inside a 3D periodic channel.

the evolution of HIT since no walls are considered: dissipation of turbulence can only occur through the energy cascade principle introduced in paragraph 3.1.1. The objective is to reproduce a turbulent flow with a mean velocity $U_b = 100 \text{ m.s}^{-1}$ and velocity fluctuations in the three directions around $u_{rms} = 10 \text{ m.s}^{-1}$. As there are no walls in the domain, the injected mean flow is simply set using a constant axial velocity profile:

$$\bar{\mathbf{u}}_{in} = (0, 0, 100)^T \quad (6.91)$$

The fluid is air ($R_{gas} = 287.15 \text{ J.kg}^{-1}.\text{K}^{-1}$) at $P = 10^5$ Pa and $T = 300$ K so that its kinematic viscosity is $\nu \approx 1.477 \times 10^{-5} \text{ m}^2.\text{s}^{-1}$. All the three axes are discretized uniformly respectively with 80 segments for the z -axis and 20 segments for both x and y axes. This discretization ends up with 32 000 hexahedral elements where it was chosen to use a polynomial of degree $p = 4$ (order 5) inside all

of them to approximate the solution. It gives a total of 4×10^6 DOF using Eq. (4.74) with the same characteristic mesh sizes between two DOF in each direction of:

$$\Delta_x^{\text{DOF}} = \Delta_y^{\text{DOF}} = \Delta_z^{\text{DOF}} = 10^{-5} \text{ m} \equiv \Delta^{\text{DOF}} \quad (6.92)$$

All the simulations were first run for 25 convective times, based on L_z and U_b which corresponds to 1 ms of physical time, to evacuate the transient due to initialization. Simulations are then relaunched for 50 convective times from 1 ms to 3 ms to collect time-average variables. The post-treatment of these statistics is done as follows:

1. The code provides \bar{u} , \bar{v} , \bar{w} , $\overline{u^2}$, $\overline{v^2}$, $\overline{w^2}$, \overline{uv} , \overline{vw} and \overline{uw} at SP.
2. Then, the velocity correlation tensor in Cartesian coordinates is computed at each SP following:

$$R^{\text{cart}} \equiv \begin{pmatrix} \overline{u'^2} & \overline{u'v'} & \overline{u'w'} \\ \overline{u'v'} & \overline{v'^2} & \overline{v'w'} \\ \overline{u'w'} & \overline{v'w'} & \overline{w'^2} \end{pmatrix} = \begin{pmatrix} \overline{u^2} - \bar{u}^2 & \overline{uv} - \bar{u} \times \bar{v} & \overline{uw} - \bar{u} \times \bar{w} \\ \overline{uv} - \bar{u} \times \bar{v} & \overline{v^2} - \bar{v}^2 & \overline{vw} - \bar{v} \times \bar{w} \\ \overline{uw} - \bar{u} \times \bar{w} & \overline{vw} - \bar{v} \times \bar{w} & \overline{w^2} - \bar{w}^2 \end{pmatrix} \quad (6.93)$$

3. \bar{u} , \bar{v} , \bar{w} and R^{cart} at each SP are interpolated on a uniform Cartesian grid with n_x , n_y and n_z points respectively along x , y and z directions.
4. Since the domain is periodic in both x and y directions, \bar{u} , \bar{v} , \bar{w} and R^{cart} on the uniform grid are spatially averaged in these directions at several axial positions (defined by the user) between $z = 0$ and $z = L_z$.
5. At the end of this averaging process, \bar{u} , \bar{v} , \bar{w} and R^{cart} can be plotted on a 1D line between $z = 0$ and $z = L_z$.

It is worth noting that step 2 and step 3 cannot be done in reverse order because the interpolation process introduces numerical errors which will accumulate when using Eq. (6.93) on the interpolated grid. Therefore, it is essential to always first compute output data at SP and then do the interpolation. The remaining numerical parameters used for all the simulations are summed up in Table 6.3. Finally,

| Riemann solver | Diffusion scheme | Time-integration scheme |
|----------------|------------------|-------------------------|
| HLLC | Average approach | SSP-ERK(3,3) |

Table 6.3. – Riemann solver, diffusion scheme and time-integration scheme used for all simulations of injection of turbulence through a periodic channel.

in paragraph 6.6.2, some results are compared with the ones obtained using the AVBP solver [283, 284] which employs the original Kraichnan's method [263] to inject a PP spectrum. AVBP simulations have been done using the Two-step Taylor Galerkin version C (TTGC) FE scheme [179] (third-order in space and time) for convective fluxes, and a FE scheme of order 2 [179] for diffusion fluxes.

6.6.2. Validation case

The turbulent parameters used to generate \mathbf{u}'_{in} for the validation case are summed up in Table 6.4. The value of N was chosen large enough to ensure a good distribution of modes in the three directions

| Energy spectrum | N [-] | Mode distribution | L_e [m] | u_{rms} [m.s ⁻¹] |
|-----------------|---------|-------------------|--------------------|--------------------------------|
| Passot-Pouquet | 1000 | Logarithmic | 2×10^{-4} | 10 |

Table 6.4. – Turbulent parameters used for the validation of the injection of HIT in a periodic channel.

x , y and z needed for isotropic turbulence. The PP spectrum is generated for turbulent structures with sizes ranging from $L_{ref} = 1 \text{ mm}$ ($\kappa_{min} = 6283 \text{ m}^{-1}$) and $L_{cut} = 6\Delta^{\text{DOF}}$ ($\kappa_{max} = 1.047 \times 10^5 \text{ m}^{-1}$)

to have at least 6 SP in the smallest structures. It is represented in Figure 6.7. This truncated PP spectrum at κ_{max} does not take into account the Kolmogorov eddy since for these values of u_{rms} and L_e , the PP spectrum gives: $\kappa_{Kol} \approx 2.030 \times 10^5 \text{ m}^{-1} > \kappa_{max}$. However, as already stated in paragraph 6.3.4, the energies of smallest eddies for the PP spectrum are very small. Thus, for $\kappa \in [\kappa_{max}, \kappa_{Kol}]$, $E(\kappa)$ is negligible and Eq. (6.53) holds for the truncated spectrum in Figure 6.7.

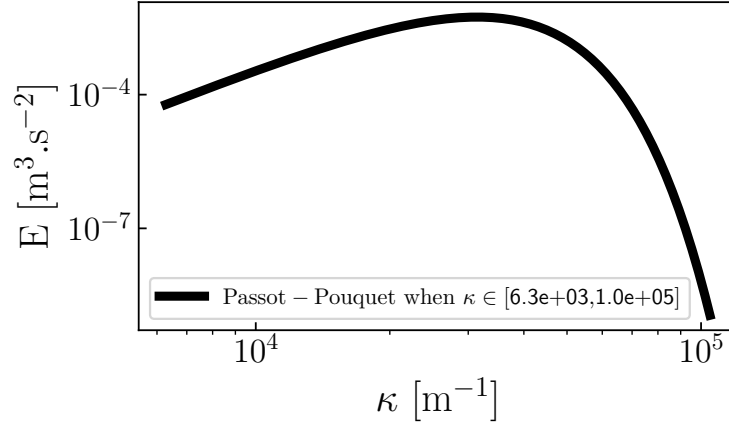


Figure 6.7. – Passot-Pouquet turbulent spectrum injected at $z = 0$.

Figure 6.8 shows the injected turbulent structures near the inlet plane which are convected into the channel. The TKE and RMS velocities decays along the channel axis are represented respectively in Figures 6.9a and 6.9b. The TKE decay obtained with AVBP solver, already validated on the same test case [276, 285], is also plotted on Figure 6.9a showing excellent agreement with the JAGUAR solution. Both codes give almost the same level of TKE at the inlet: $\text{TKE}(0) = 149.8 \text{ m}^2.\text{s}^{-2}$ for JAGUAR and $\text{TKE}(0) = 142.2 \text{ m}^2.\text{s}^{-2}$ for AVBP. These values are very close to the expected value of $150 \text{ m}^2.\text{s}^{-2}$ given by Eq. (6.50). The small difference between JAGUAR and AVBP is due to the presence of a mask put on inlet edges for AVBP where velocity is set to zero at the nodes to ensure the periodicity at these inlet nodes which are also shared with the periodic boundary condition. This is not needed in the JAGUAR case since there are no FP on the periodic boundary conditions that are shared with the NSCBC inlet. Looking now at Figure 6.9b, the three RMS velocities start very close to u_{rms} at the inlet and decay in a similar manner showing isotropy of the HIT field in the channel. Therefore, these results validate the implementation of turbulence injection in JAGUAR detailed in Section 6.4

6.6.3. Influence of the energy spectrum

The test case of paragraph 6.6.2 can be computed using the VKP energy spectrum with the same values of L_e and u_{rms} summed up in Table 6.5. The form of the injected VKP spectrum, still between

| Energy spectrum | N [-] | Mode distribution | L_e [m] | u_{rms} [m.s ⁻¹] |
|-----------------|---------|-------------------|--------------------|--------------------------------|
| von Karman-Pao | 1000 | Logarithmic | 2×10^{-4} | 10 |

Table 6.5. – Turbulent parameters used for the injection of HIT in a periodic channel with the VKP spectrum.

$\kappa_{min} = 6283 \text{ m}^{-1}$ and $\kappa_{max} = 1.047 \times 10^5 \text{ m}^{-1}$, is represented in Figure 6.10a. It is compared on Figure 6.10b with the PP spectrum of Figure 6.7 showing that the VKP spectrum has much more structures with high energy than the PP one. Consequently, the integral of the VKP spectrum in Figure 6.10a gives a TKE of $97 \text{ m}^2.\text{s}^{-2}$ instead of $150 \text{ m}^2.\text{s}^{-2}$. It is due to two factors:

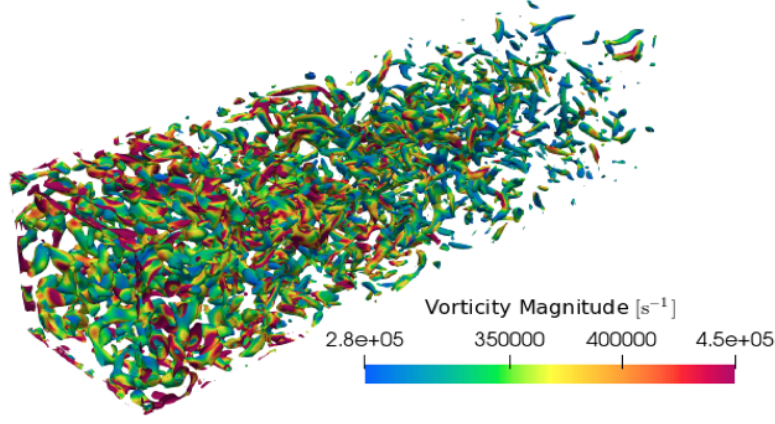
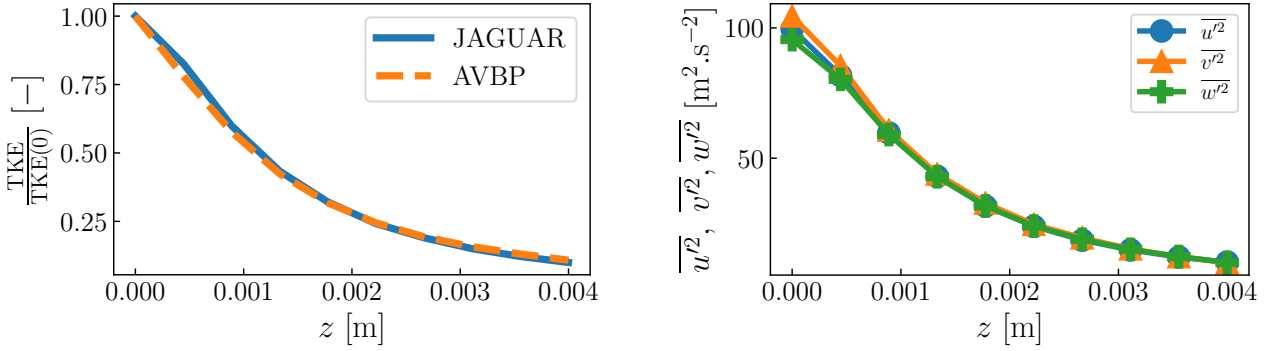


Figure 6.8. – Contours of Q-criterion colored by vorticity magnitude. The flow is going from the left to the right.



(a) TKE obtained with JAGUAR and AVBP.

(b) RMS velocities square obtained with JAGUAR.

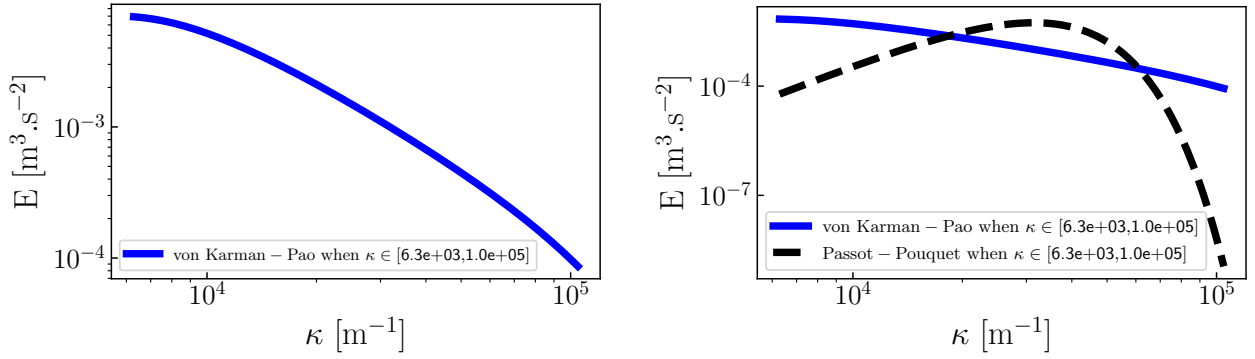
Figure 6.9. – Evolution of the TKE and of the RMS velocities along channel axis.

1. The whole VKP spectrum has a very large range of wavenumbers where $E > 10^{-9} \text{ m}^3 \cdot \text{s}^{-2}$ as shown in Figure 6.11a where it is plotted for κ such that $E(\kappa) > 10^{-12} \text{ m}^3 \cdot \text{s}^{-2}$ (blue curve), for $\kappa \in [\kappa_{min}, \kappa_{Kol}]$ (green curve) and for $\kappa \in [\kappa_{min}, \kappa_{max}]$ (orange curve). The same plot is represented for the PP spectrum used in paragraph 6.6.2 on Figure 6.11b. Because the PP spectrum has almost all its energy concentrated on a small range of wavenumbers, the integrals of the three curves are equal to the TKE. However, as already mentioned, for the VKP spectrum the integral of the discretized energy spectrum (orange curve) gives $97 \text{ m}^2 \cdot \text{s}^{-2} \neq 150 \text{ m}^2 \cdot \text{s}^{-2}$. The integral of the green curve, which takes into account Kolmogorov eddy, gives $100 \text{ m}^2 \cdot \text{s}^{-2}$ and the one of the blue curve is $124 \text{ m}^2 \cdot \text{s}^{-2}$. Thus, where are the $26 \text{ m}^2 \cdot \text{s}^{-2}$ left? This brings the reasoning to the second factor.
2. In the calculations of the VKP constants developed in paragraph 6.3.5 and in appendix G, the integrals are evaluated using the hypothesis that:

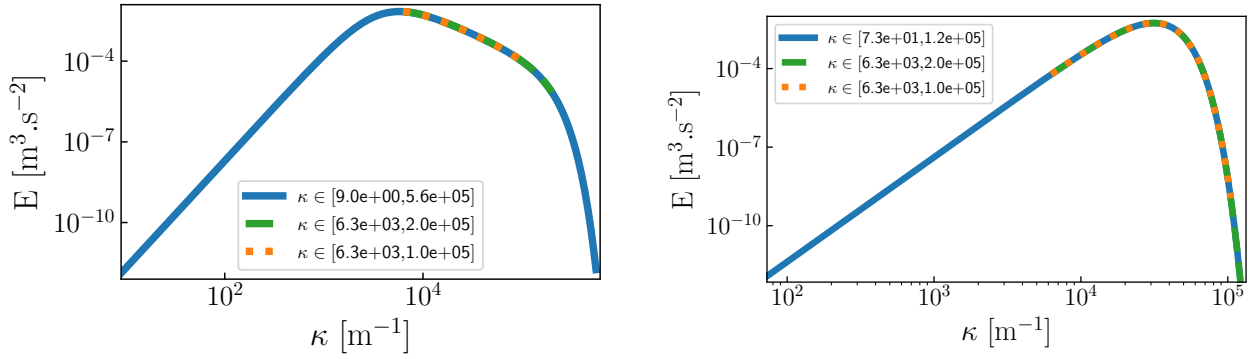
$$\left(\frac{\kappa_{Kol}}{\kappa_e} \right)^2 \rightarrow \infty \quad (6.94)$$

Nevertheless, in the case studied here: $\kappa_{Kol} \approx 2 \times 10^5 \text{ m}^{-1}$ and $\kappa_e = 3734 \text{ m}^{-1}$ which gives $(\kappa_{Kol}/\kappa_e)^2 \approx 2800$ not really "big".

Consequently, when using the VKP spectrum, the discrete values of E injected in the domain are usually rescaled so that the integral of the injected VKP spectrum is equal to TKE in order to have


 (a) Injected VKP spectrum at $z = 0$.

 (b) Injected PP and VKP spectrums at $z = 0$.

 Figure 6.10. – Comparison of injected PP and VKP spectrums at $z = 0$.

 (a) VKP spectrum for different κ values.

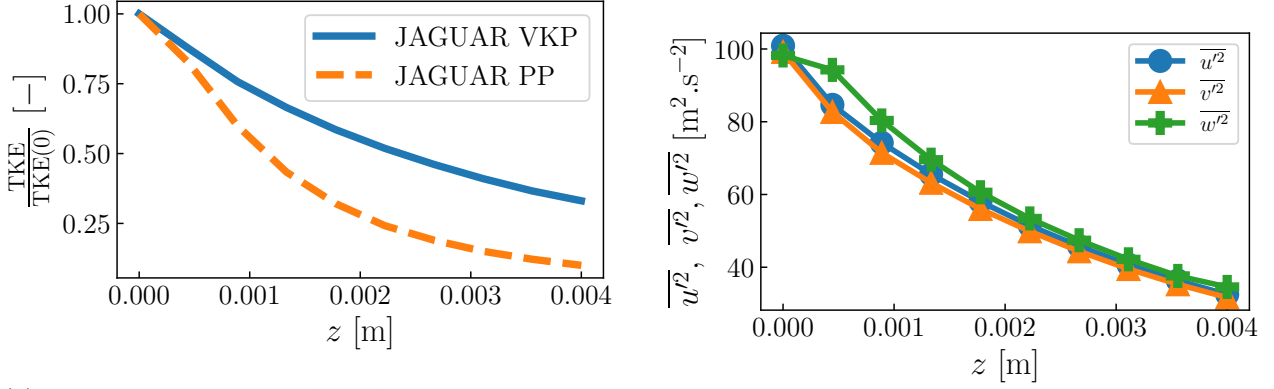
 (b) PP spectrum for different κ values.

 Figure 6.11. – VKP and PP spectrums represented for different ranges of κ . Blue curves, green curves and orange curves correspond respectively to κ such that $E(\kappa) > 10^{-12} \text{ m}^3 \cdot \text{s}^{-2}$, $\kappa \in [\kappa_{min}, \kappa_{Kol}]$ and $\kappa \in [\kappa_{min}, \kappa_{max}]$.

the expected level of energy at the inlet. That is why, the VKP structures are even more energetic than the PP ones as it is shown in Figure 6.12a where the TKE dissipates much less than for the PP spectrum using the same u_{rms} and L_e . The isotropy of the HIT field is also verified with the VKP spectrum since in Figure 6.12b all the RMS velocities decay in the same manner along channel axis.

6.6.4. Influence of the number of modes and of their distribution

In paragraphs 6.6.2 and 6.6.3, the value of N was set to 1000 in order to have enough modes in the three directions x , y and z . Otherwise, with small values for N , isotropy can be lost since a direction can have many more modes than the two remaining ones. However, using $N = 1000$ modes at the inlet is costly when considering bigger applications than the channel flow showed here. Load-balancing is possible by flagging, before the computation, the elements at inlet in order to put more processors on them. Nevertheless, although it is possible to gain 30% in computational time, many tests have to be made to find an optimal ratio of processors set to deal with element at inlet and the remaining ones dealing with the rest of the domain. That is why, a good compromise value for N is interesting to search for: high enough to keep sufficient isotropy but small enough to not be too expensive. Simulations of the periodic channel flow validated in paragraph 6.6.2 have been carried out with $N = 100$, 300 and 600 modes using a linear or a logarithmic distribution of the modes and for both



(a) TKE obtained with JAGUAR for PP and VKP spectrums.

(b) RMS velocities square for the VKP spectrum.

Figure 6.12. – Comparison of TKE decays between PP and VKP obtained with JAGUAR and RMS velocities decay in the channel for the VKP spectrum.

PP and VKP spectrums. Results in terms of decay of RMS velocities within the channel are presented in Figures 6.13 and 6.14 respectively for the PP and VKP spectrums. When $N = 100$, no matter the mode distribution or the type of energy spectrum: the velocity field generated at the inlet is not isotropic, as there is always at least one RMS velocity which is far from the expected value of 10 m.s^{-1} . When $N = 300$, isotropy is kept by the VKP spectrum, independently of mode distribution, whereas the PP spectrum does not show good isotropy with a logarithmic distribution of modes although it is almost the case for the linear distribution. Looking at Figure 6.10, it can come from the fact that values of E for the PP spectrum are in a larger range than the VKP one. Thus, if for instance there are more modes which have x -axis as the principal direction (highest components of σ_n along x -axis) then the differences in terms of E values are more accentuated. This is the case for the PP spectrum when the logarithmic distribution is employed because there are much more modes with high E values that are in the lower range of κ and only a small amount of modes which have low E values in the larger range of κ . The VKP spectrum does not have such issues since its values of E shown in Figure 6.10a vary in a smaller range. Finally, when $N = 600$, isotropy is retrieved in all cases especially for the PP spectrum with a logarithmic distribution of modes. In terms of computational cost, Table 6.6 sums up the iteration cost as a function of N showing that from $N = 600$ the turbulence injection is becoming very costly. Based on the RMS decays with N study, it seems that a good compromise in terms of accuracy and computational cost is $N = 300$ especially if the generated spectrum does not have a large ratio $\max_{\kappa}(E) / \min_{\kappa}(E)$. Otherwise, increasing N up to 600 maximum seems sufficient to have a correct HIT field but at a higher cost. It would be interesting in the future to see the influence of changing p , at a constant DOF value, on the results and also on the iteration cost with respect to N .

| N | 100 | 300 | 600 | 1000 |
|---|-----|------|------|------|
| Iteration cost [$\mu s.it^{-1}.DOF^{-1}$] | 6.4 | 7.4 | 9.5 | 11.0 |
| % of increase from $N = 100$ | - | 15 % | 48 % | 72 % |

Table 6.6. – Iteration cost as a function of N in the case of the periodic channel. All computations were done on 600 processors provided by the same machine.

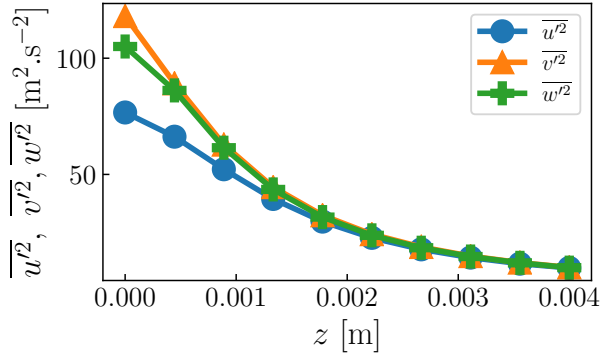
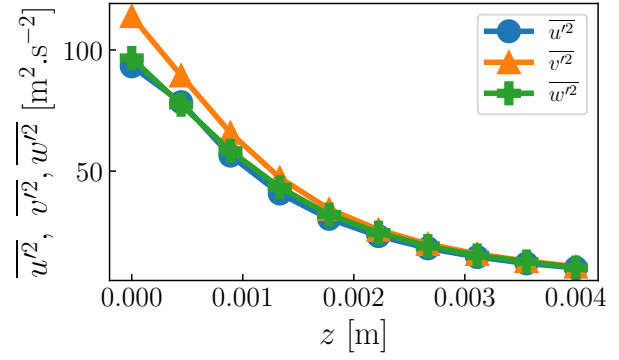
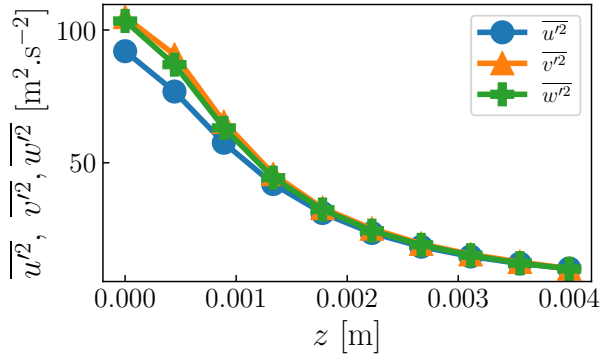
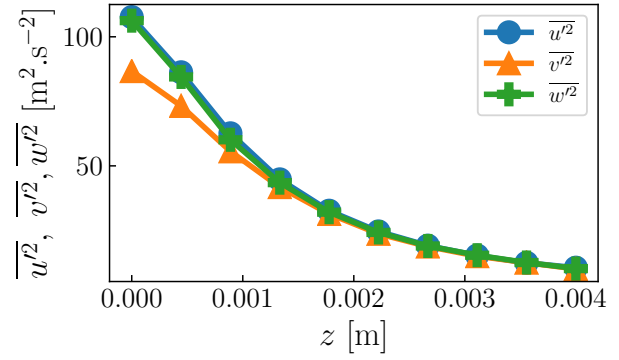
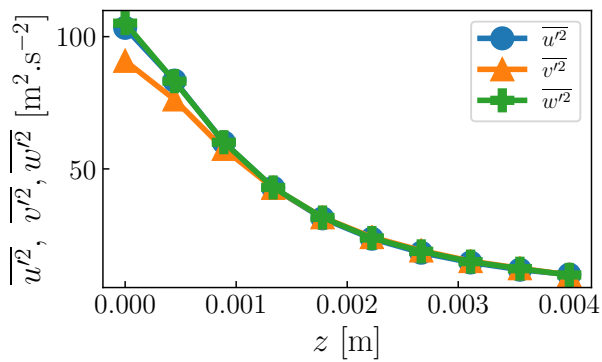
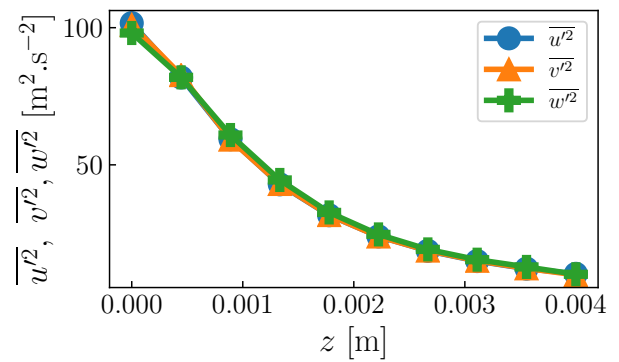
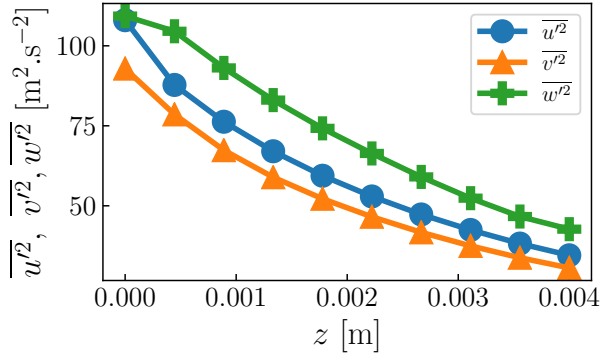
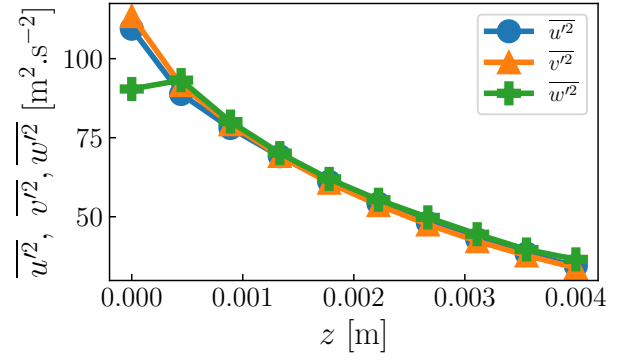

 (a) $N = 100$ and linear distribution

 (b) $N = 100$ and logarithmic distribution

 (c) $N = 300$ and linear distribution

 (d) $N = 300$ and logarithmic distribution

 (e) $N = 600$ and linear distribution

 (f) $N = 600$ and logarithmic distribution

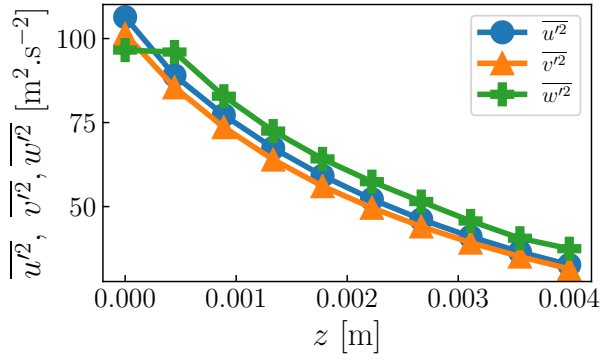
Figure 6.13. – Evolution of statistics using Passot-Pouquet energy spectrum for turbulence injection.



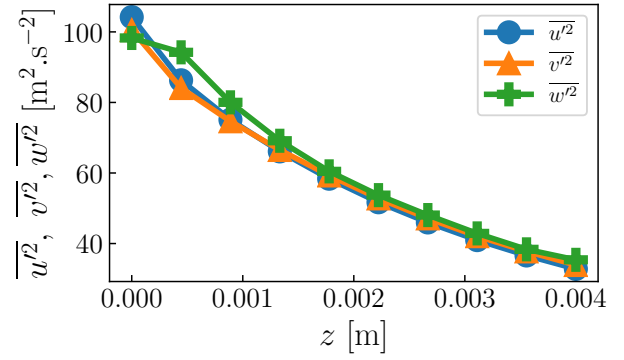
(a) $N = 100$ and linear distribution



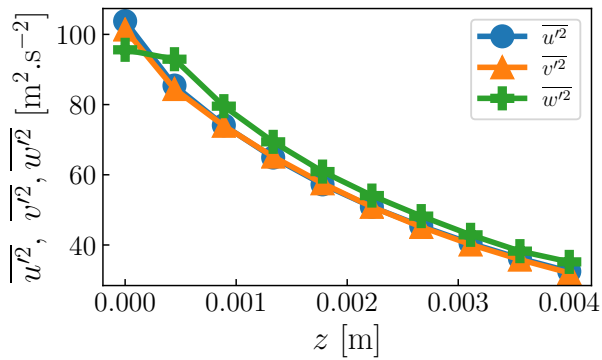
(b) $N = 100$ and logarithmic distribution



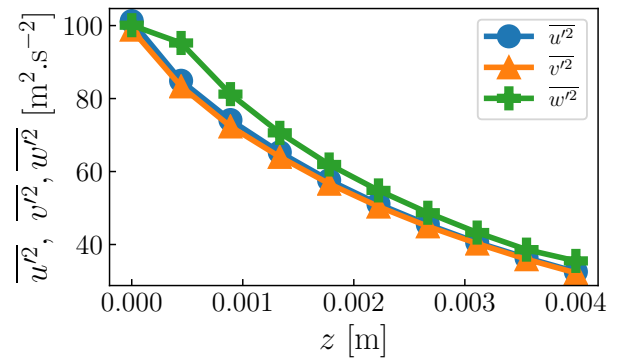
(c) $N = 300$ and linear distribution



(d) $N = 300$ and logarithmic distribution



(e) $N = 600$ and linear distribution



(f) $N = 600$ and logarithmic distribution

Figure 6.14. – Evolution of statistics using von Karman-Pao energy spectrum for turbulence injection.

6.7. Turbulent wall-bounded channel flow

In Section 6.6, the injection of HIT, generated by a synthetic random Fourier method, was validated on a channel flow where transverse and spanwise boundary conditions were periodic in order to maintain the hypotheses behind HIT flows. However, in real configurations, HIT flows are not common because walls are usually employed creating boundary layers due to viscous effects. It entails the apparition of mean axial pressure gradients and mean velocity gradients in the wall normal directions. Thus, according to Eq. (6.7), cross-correlations $\overline{u'_i u'_j}$ are non-zero for $i \neq j$ making the turbulence anisotropic and non-homogeneous in the presence of walls. The objective of this section is to reproduce a fully turbulent flow at $Re_\tau = 550$ within a wall-bounded channel on a LES mesh and to compare with the DNS data available from Del Alamo *et al.* [241]. Three methods are compared to create the fully turbulent flow:

- **Method 1:** using only HIT injection validated in Section 6.6.
- **Method 2:** using HIT injection validated in Section 6.6 along with artificial tripping introduced in paragraph 6.5.2.
- **Method 3:** using the injection of non-homogeneous and anisotropic turbulence described in paragraph 6.5.1.

The channel length required for a fully turbulent flow will be investigated.

6.7.1. Numerical setup

The computational domain is the wall-bounded channel flow represented on Figure 6.15. Boundary

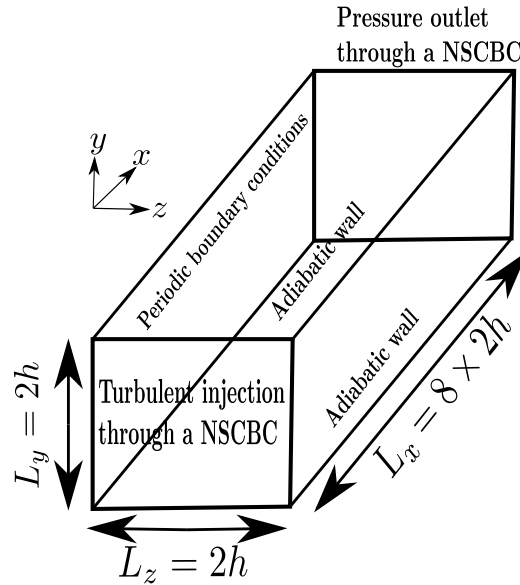


Figure 6.15. – Computational domain and boundary conditions for the turbulent wall-bounded channel flow.

conditions are set as follows:

- Turbulence injection through a subsonic NSCBC inlet is used at $x = 0$.
- Constant pressure at $P_{out} = 101325$ Pa through a subsonic NSCBC outlet is used at $x = 16h$.
- Adiabatic no-slip walls are considered at $y = -h$ and $y = h$.
- Periodic boundary conditions are employed between $z = -h$ and $z = h$.

where $h = 1.5$ mm is the channel half-height. Since $Re_\tau = 550$, the bulk Reynolds number based on $2h$ is $Re_b \approx 2 \times 10^4$ from Eq. (6.18). The fluid considered is air ($R_{gas} = 287.15$ J.kg⁻¹.K⁻¹) at $P = 101325$ Pa and $T = 300$ K with $\mu = 1.85 \times 10^{-5}$ kg.m⁻¹.s⁻¹ so that $\nu \approx 1.573 \times 10^{-5}$ m².s⁻¹. Consequently, the bulk velocity to be imposed at the inlet plane must be around:

$$U_b = \frac{Re_b \nu}{2h} \approx 105 \text{ m.s}^{-1} \quad (6.95)$$

which is in accordance with the integral of the axial mean velocity profile showed in Del Alamo *et al.* [241]. The mesh is designed to have:

$$y^+ \approx 1 \text{ and } x^+ = z^+ \approx 22 \quad (6.96)$$

Since $\delta_\nu = 2.72 \times 10^{-6}$ m from Eq. (6.17), the closest SP must be around $y = y_w \approx \delta_\nu$ and if the polynomial degree is set to $p = 4$ (order 5), the corresponding element size in the y -direction has to be $\Delta_w \approx 1.36 \times 10^{-5}$ m using Eq. (6.25). Consequently, to capture the viscous boundary layer, the y -direction is discretized using a geometric progression for the element size in this direction. Starting from Δ_w and with a common ratio of 0.07, it gives an element which is around $12\Delta_w$ of size, along the y -direction, at the center of the channel. Using Eq. (6.19) with x^+ and z^+ instead of y^+ and in order to have uniform discretizations in both axial and spanwise directions, the x and z directions are respectively discretized with 80 and 10 segments. The final mesh is then composed of 28800 hexahedral elements which gives a total number of DOF of 3.6×10^6 when $p = 4$. All the simulations were run first for 3 convective times, based on L_x and U_b of this case, which corresponds to 0.69 ms of physical time. From the restart solution at 0.69 ms, time-average variables are collected during 8 convective times corresponding to 1.83 ms. This value of 8 convective times was shown to be sufficient for the statistics to be converged. The post-treatment of these statistics is done as follows:

1. The code provides \bar{u} , \bar{v} , \bar{w} , $\overline{u^2}$, $\overline{v^2}$, $\overline{w^2}$, \overline{uv} , \overline{vw} and \overline{uw} at SP.
2. R^{cart} is computed at all SP using Eq. (6.93).
3. \bar{u} , \bar{v} , \bar{w} and R^{cart} at each SP are interpolated on an uniform Cartesian grid with n_x , n_y and n_z points respectively along x , y and z directions.
4. Since the domain is periodic in z direction, \bar{u} , \bar{v} , \bar{w} and R^{cart} are spatially averaged in this direction to get a 2D plane of the channel with $(x, y) \in [0, L_x] \times [-h, h]$.
5. At the end of this averaging process, 1D profiles along y direction at different axial positions can then be plotted from the 2D average plane.

The remaining numerical parameters for Riemann solver, diffusion scheme and time-integration procedure used for all the simulations are the same as in Table 6.3.

6.7.2. Injection of HIT only at the inlet

In this paragraph, HIT is injected at the inlet of the channel and is superimposed to an injected axial mean flow given by Eq. (6.38) (for \bar{u}_{in} instead of \bar{w}_{in}), with $n = 7$ and $U_{cl} = 121.1$ m.s⁻¹ obtained from Eq. (6.39) to have the expected bulk velocity. The objective is to see if only HIT injection in the presence of walls can be able to reproduce, after a certain distance from the inlet in the channel, a turbulent channel flow similar to the one simulated by Del Alamo *et al.* [241] with DNS and considered as the reference here. The choice of the turbulent parameters for injected HIT at the inlet is crucial and needs some preliminary tests. Table 6.7 sums up the final values retained in this paragraph for these parameters. Since they were only a few million of SP only, N was taken to 1000. Concerning the value of u_{rms} , based on the DNS data profiles, it seemed that RMS velocities at channel centers were around 4 m.s⁻¹ for $\sqrt{\overline{v'^2}}$ and $\sqrt{\overline{w'^2}}$ and around 5 m.s⁻¹ for $\sqrt{\overline{u'^2}}$. Thus, after some tests, a value a bit higher than 5 m.s⁻¹ was found appropriate to have correct RMS velocities at channel center. Characteristic sizes were chosen as follows:

| Energy spectrum | N [-] | Mode distribution | L_e [mm] | L_{ref} [mm] | L_c [mm] | u_{rms} [m.s ⁻¹] |
|-----------------|---------|-------------------|------------|----------------|------------|--------------------------------|
| Passot-Pouquet | 1000 | Logarithmic | 0.85 | 1.5 | 0.12 | 5.33 |

Table 6.7. – Turbulent parameters used for the HIT injection in a turbulent wall-bounded channel flow.

- based on previous studies using PP spectrum with walls [286, 287], L_e was set between $2h/4$ and $2h/3$.
- L_{ref} was set equal to h to have structures with a size of at most the channel half-width.
- L_{cut} was set equal to $2\Delta_x^{\text{DOF}} = 2\Delta_z^{\text{DOF}}$ since x and z have the coarser meshing and to have at least 2 SP in the smallest structures.

Transverse profiles along y direction at different axial positions x are represented in Figures 6.16 and 6.17 respectively for the mean axial flow and the RMS velocities. They are plotted against DNS data using friction variables: JAGUAR velocities are divided by $u_\tau \approx 5.80 \text{ m.s}^{-1}$ obtained in the simulation and close to the DNS value of 5.768 m.s^{-1} obtained from Eq. (6.17). The results show that a turbulent channel flow is almost reproduced from $x = 6 \times 2h$ where velocity profiles have converged to an established flow close to DNS results especially at channel center. The results at $x = 8 \times 2h$, corresponding to the outlet position, were not shown because they are on a boundary condition which usually does not exist in real configurations where the channel is only the injector part of the whole system. Therefore, it seems that a channel length between $6 \times 2h$ and $7 \times 2h$ is sufficient to have a turbulent channel flow with JAGUAR using the turbulence injection methodology described in Section 6.4. However even at $x = 7 \times 2h$, far from the channel center, the near-wall turbulence is under-predicted because only HIT is employed to trigger the turbulence that is why additional treatments such as artificial tripping or the injection of anisotropic turbulence can be needed and are the topics of the next two paragraphs.

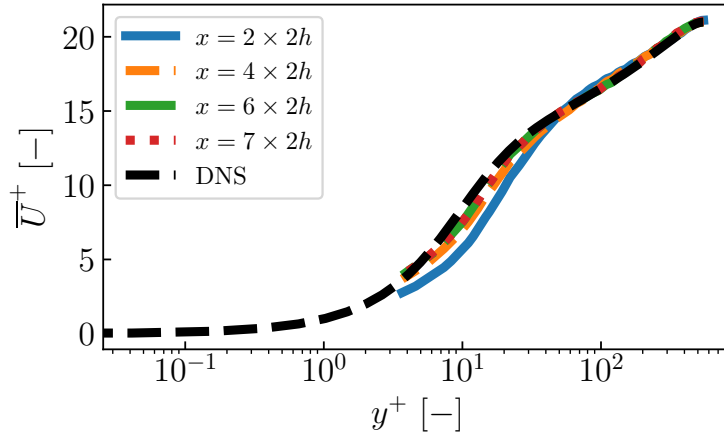


Figure 6.16. – Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at different axial positions in the channel when only HIT is injected at channel inlet. DNS results of Del Alamo *et al.* [241] are also represented for comparison.

6.7.3. Injection of HIT at the inlet coupled with artificial tripping

The same situation as the one shown in paragraph 6.7.2 is considered with same injected mean flow and same turbulent injection parameters. In order to better reproduce near-wall turbulence, two Cartesian artificial steps (one on each wall), like the one shown in Figure 6.5a, are added to

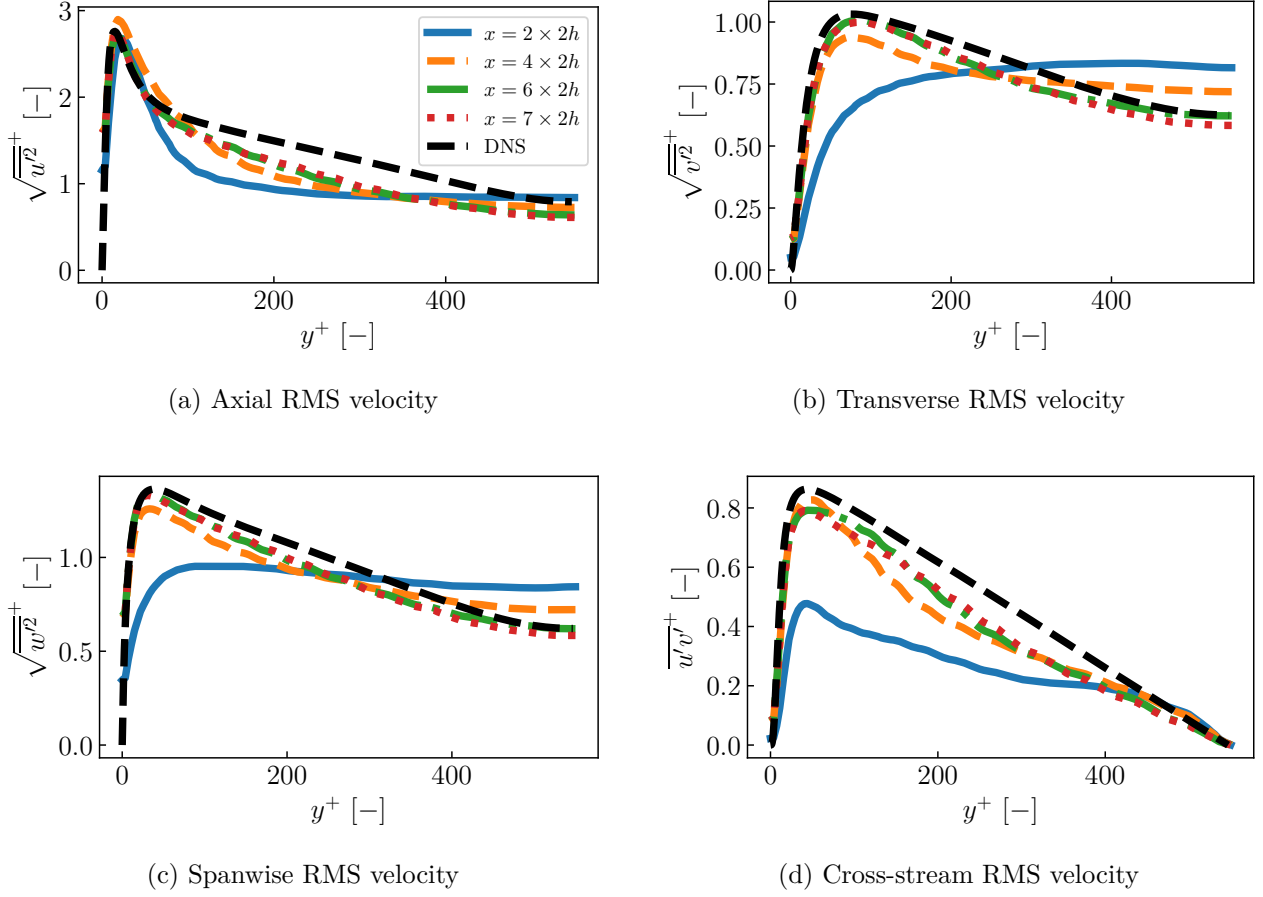


Figure 6.17. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at different axial positions in the channel. DNS results of Del Alamo *et al.* [241] are also represented for comparison.

the simulation. Firstly, it is important to locate them at appropriate positions based on experience and best practices with the code. Following what was done in Boudet *et al.* [282], the two artificial roughness elements start at $x = h = 1.5$ mm and end at $x = 1.8$ mm in order to have 5 SP inside steps length since $\Delta_x^{\text{DOF}} = 6 \times 10^{-5}$ m. The height of these steps is set to be 8% of h so that the artificial step at the top wall is applied for $y \in [0.92h, h]$ and at the bottom wall for $y \in [-h, -0.92h]$. Finally, both artificial steps are activated all along spanwise direction so for $z \in [-h, h]$.

Now that the two artificial roughness elements have been located, the only parameter remaining is the value of C_D introduced in Eq. (6.88) which needs to be calibrated. Thus, simulations have been performed using three different values of C_D : 1, 1.5 and 2. Results at $x = 7 \times 2h$ are compared in Figures 6.18 and 6.19 with the ones obtained in paragraph 6.7.2 with only HIT injection. The mean axial flow is not impacted by the artificial tripping treatment according to Figure 6.18. However, RMS velocities results are improved when using this treatment especially the axial (Figure 6.19a) and cross-stream (Figure 6.19d) components. All the three values of C_D give good results and $C_D = 1.5$ can be retained as the optimal choice between them since sometimes $C_D = 2$ over-predicts a bit DNS results whereas $C_D = 1$ under-predicts them sometimes. Of course, a better optimal value for C_D can be investigated (it is completely case-dependent) but the purpose of this reasoning was to show that artificial tripping can help HIT injection in reproducing near-wall turbulence. It should be mentioned that using only the artificial tripping, without HIT injection, takes way too much time

and a much longer channel length to reach an established turbulent flow. Therefore, the use of HIT injection is really the key element that accelerates the transition to turbulence and artificial tripping is an additional tool that can be added to have a better description of near-wall turbulence.

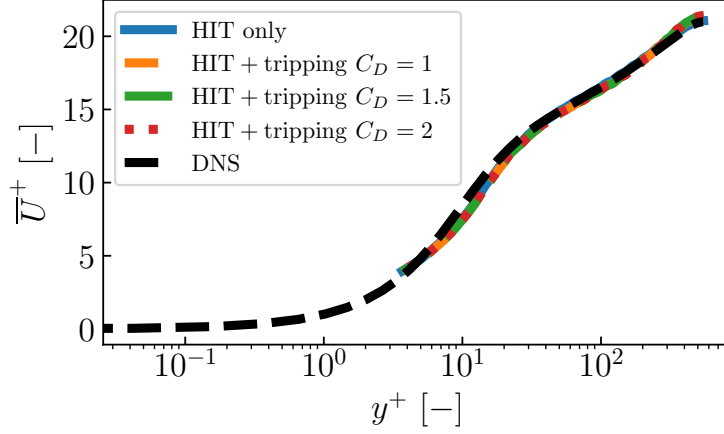


Figure 6.18. – Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at $x = 7 \times 2h$ for three values of C_D . DNS results of Del Alamo *et al.* [241] and results with only HIT injected at the inlet are also represented for comparison.

6.7.4. Injection of non-homogeneous and anisotropic turbulence at the inlet

In this paragraph, the injection of non-homogeneous and anisotropic turbulence, described in paragraph 6.5.1 and based on the DNS profiles of Del Alamo *et al.* [241], is employed to see if this technique is also able to reproduce a turbulent channel flow around $x = 6 - 7 \times 2h$. As explained in paragraph 6.5.1, a turbulent HIT field with $u_{rms} = 1 \text{ m.s}^{-1}$ has to be generated first before applying Eq. (6.84). Then, using Eq. (6.86) any given value of u_{rms} will end up with a turbulent HIT field satisfying $u_{rms} = 1 \text{ m.s}^{-1}$. Consequently, the same turbulent parameters written in Table 6.7 are employed except that at the end of the HIT generation process Eq. (6.86) is used to rescale the N turbulent modes ending up with \mathbf{u}_{in}^{HIT} . Then, Eq. (6.84) can be employed where the Cholesky matrix is computed as follows from the DNS data:

1. The DNS of Del Alamo *et al.* provides y^+ , $\bar{U}^+(y^+)$ and $R^{cart,+}(y^+)$ where $R^{cart,+} = R^{cart}/u_\tau^2$.
2. A python script with input data h, μ, R_{gas}, P, T and Re_τ converts y^+ , $\bar{U}^+(y^+)$ and $R^{cart,+}(y^+)$ into $y, \bar{u}(y)$ and $R^{cart}(y)$.
3. The Cholesky matrix $A(y)$ is computed from $R^{cart}(y)$ using Eq. (6.85).
4. The last step, due to the SD framework, is a 1D interpolation of $A(y)$ at inlet FP since the previous y coordinates are not the ones of inlet FP but those of the DNS data.

While u_{rms} does not play a role anymore, the value of L_e still has an impact on the injected turbulence: it gives the size of the most energetic structures of the non-homogeneous and anisotropic generated turbulent field. Thus, three different values of L_e are tested in this case to see their impact on the results: 0.75 mm, 0.85 mm (the value in Table 6.7) and 1 mm. A last difference with the two previous cases should be also mentioned: the injected mean flow at the inlet is the one coming from the DNS of Del Alamo *et al.* [241] since the objective was to inject available DNS data into the channel and see what happens. Results at $x = 7 \times 2h$ are summed up in Figures 6.20 and 6.21. As for the injected mean flow with artificial tripping, the mean flow is almost identical even with non-homogeneous and anisotropic injected turbulence and an injected mean flow corresponding to the DNS case. Concerning

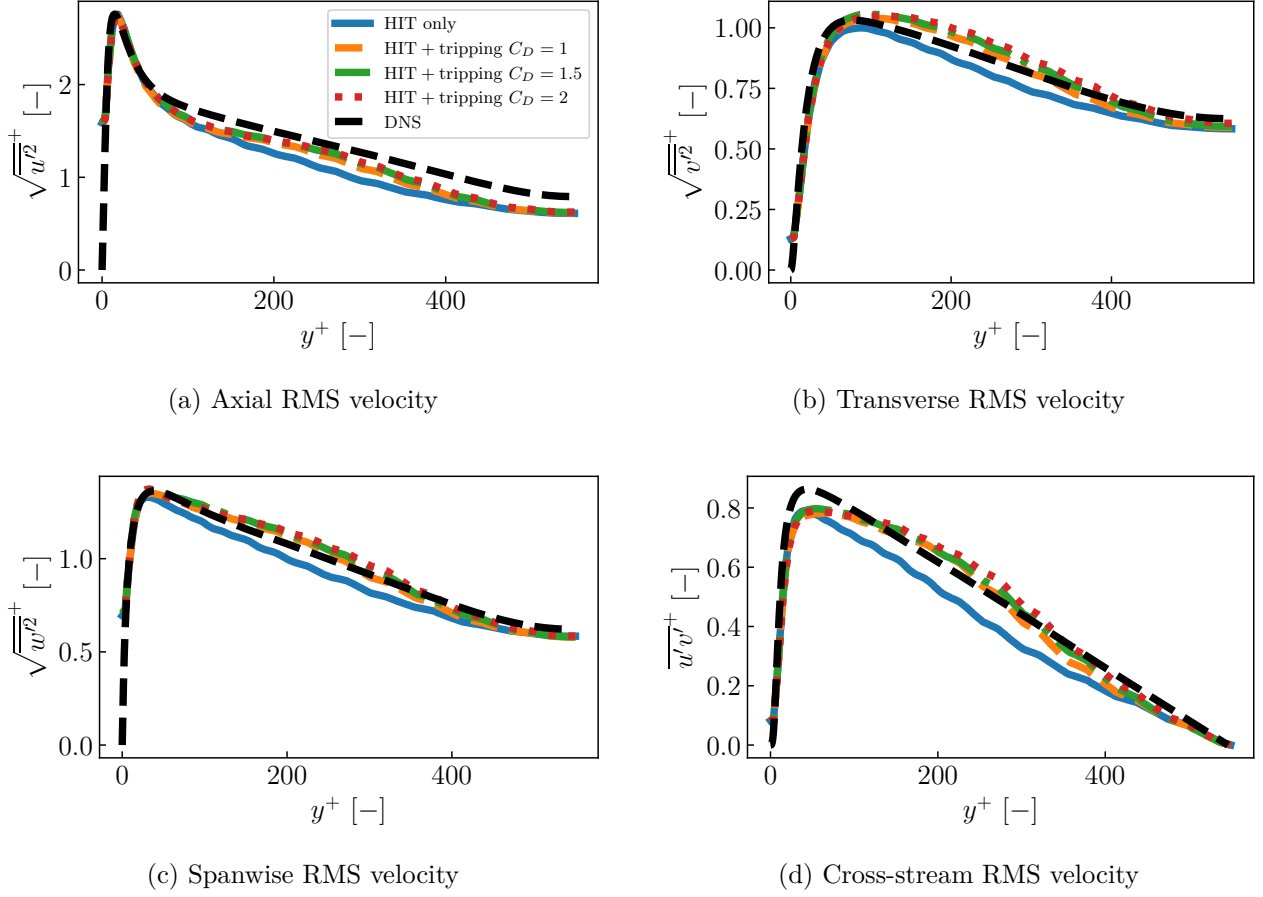


Figure 6.19. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at $x = 7 \times 2h$ for three values of C_D . DNS results of Del Alamo *et al.* [241] and results with only HIT injected at the inlet are also represented for comparison.

the RMS velocities illustrated in Figure 6.21, the near-wall turbulence is well captured for all values of L_e . However, it seems that using too high values of L_e can lead to an overestimation of the turbulence at channel center such as for $L_e = 0.85$ mm and $L_e = 1$ mm on Figure 6.21b. On the other hand, looking at cross-stream RMS on Figure 6.21d, using $L_e = 1$ mm gives the best results. Therefore, it seems that there is some trade-off to do for the value of L_e : not too big for turbulence at channel center but not too small either to not under-predict turbulence in the boundary layer. It is probable that this conclusion will be different if a VKP spectrum is used instead which has a more uniform energy distribution. Except these differences due to L_e , this methodology to inject non-homogeneous and anisotropic turbulence also shows that it is able to reproduce a turbulent channel flow after a distance around $7 \times 2h$.

6.7.5. Axial pressure gradient and acoustics in the channel

A last element that can be checked in a turbulent wall-bounded channel flow, is that the pressure gradient due to friction at walls is correctly estimated. Combining Eq. (6.12) and Eq. (6.15), the mean axial pressure gradient at channel center follows:

$$\frac{\partial \bar{P}}{\partial x} = \frac{-\rho u_\tau^2}{h} \iff \bar{P}(x, 0) = P_{out} + (L_x - x) \frac{\rho u_\tau^2}{h} \quad (6.97)$$

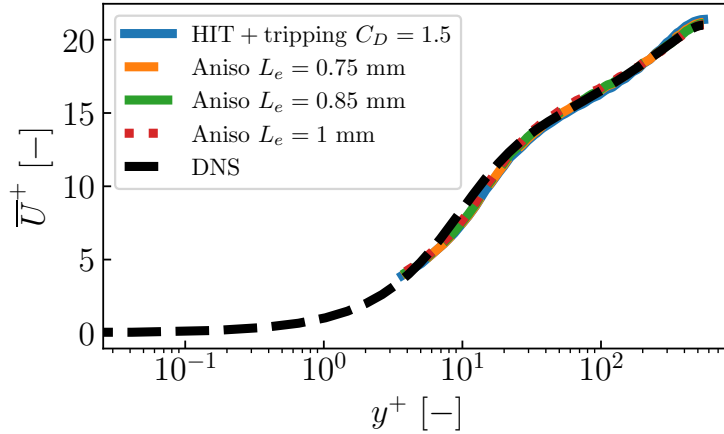


Figure 6.20. – Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at $x = 7 \times 2h$ for three values of L_e when injecting non-homogenous and anisotropic turbulence. DNS results of Del Alamo *et al.* [241] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison.

where $P_{out} = 101325$ Pa here. Figure 6.22a shows the evolution of $\bar{P}(x, 0)$ for the three previously studied injection cases: HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous anisotropic turbulence with $L_e = 0.85$ mm. The theoretical curve corresponds to Eq. (6.97) computed with $u_\tau = 5.768 \text{ m.s}^{-1}$ which is the expected value for $\text{Re}_\tau = 550$. In all cases, the expected evolution of $\bar{P}(x, 0)$ is retrieved in the second half of the channel. It is in accordance with the results on velocities showing that a turbulent channel flow was obtained from $x \approx 12h$. The artificial tripping induces a strong deviation of the mean pressure around its location in $x = h$ but the flow rapidly tends back to the expected pressure decay. Figure 6.22b presents the non-dimensionalized RMS of pressure along $y = 0$ within the channel giving an estimation of the noise generated by the injection of turbulence in the three cases: HIT only, HIT coupled with artificial tripping and non-homogeneous anisotropic turbulence. All the three methodologies generate noise mostly at the inlet but is rapidly dissipated due to friction along the channel. Although it is dissipated, this noise is still higher than what DNS results have predicted which should be around 5 for $\sqrt{\bar{P}^2}(x, 0)/\rho u_\tau^2$ at this friction Reynolds number [243, 288] instead of approximately 17 along the channel according to Figure 6.22b. One thing also not expected is that the HIT only case produces more noise at the inlet than the non-homogeneous and anisotropic case where Eq. (6.74) is not satisfied. More investigations are required to identify the source of this noise in both HIT and non-homogeneous and anisotropic turbulent cases. It can come from the NSCBC treatment itself at the inlet where some additional treatments for acoustics have been recently developed [276]. There is also the incompressibility condition of Eq. (6.74) for the HIT field that is satisfied in the continuous case without discretization of the space but there is no guarantee that the discrete divergence of velocity is zero and some additional corrections can be employed as in Saad *et al.* [272]. The way to treat properly these noise issues is beyond the scope of this manuscript and is left for future work.

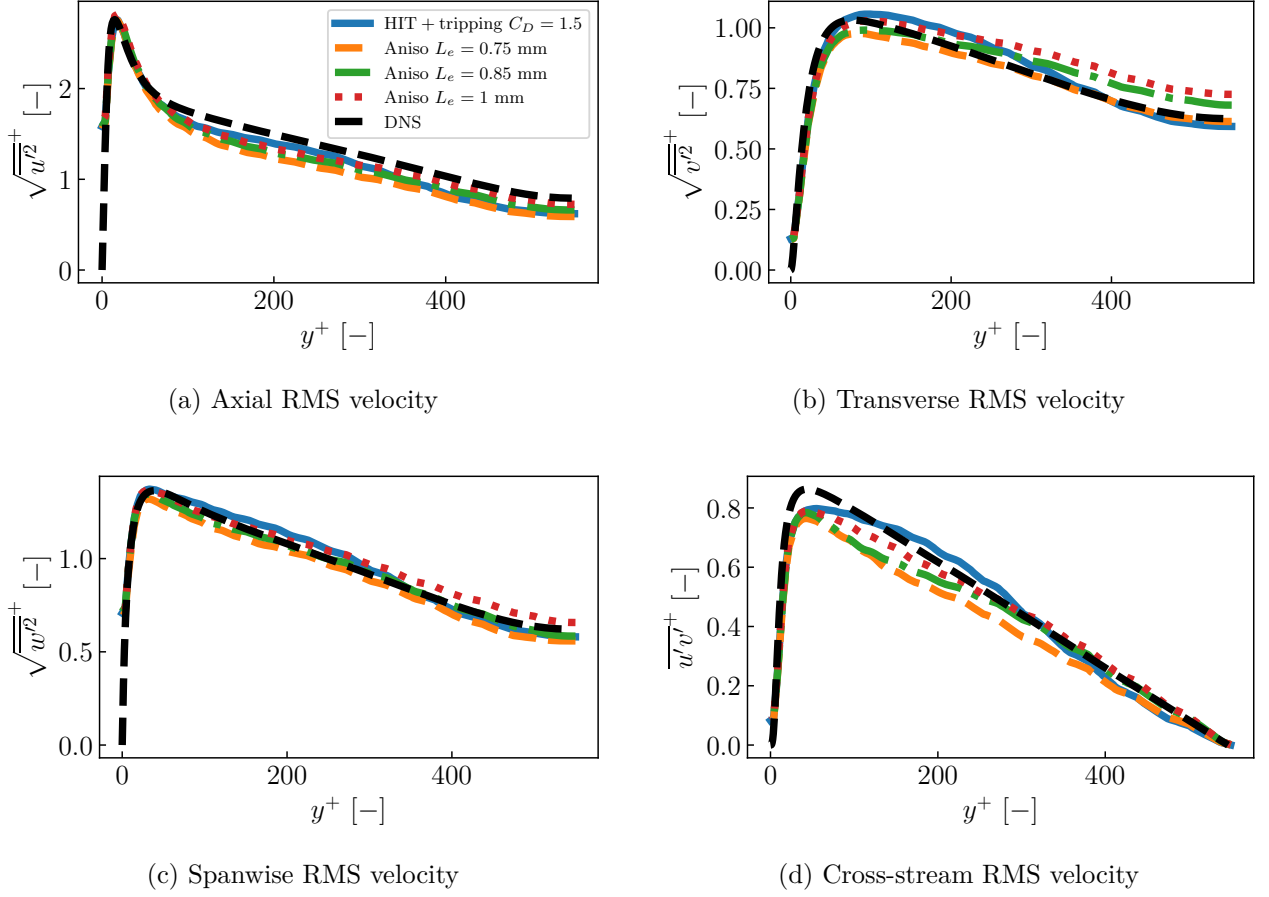


Figure 6.21. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at $x = 7 \times 2h$ for three values of L_e when injecting non-homogenous and anisotropic turbulence. DNS results of Del Alamo *et al.* [241] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison.

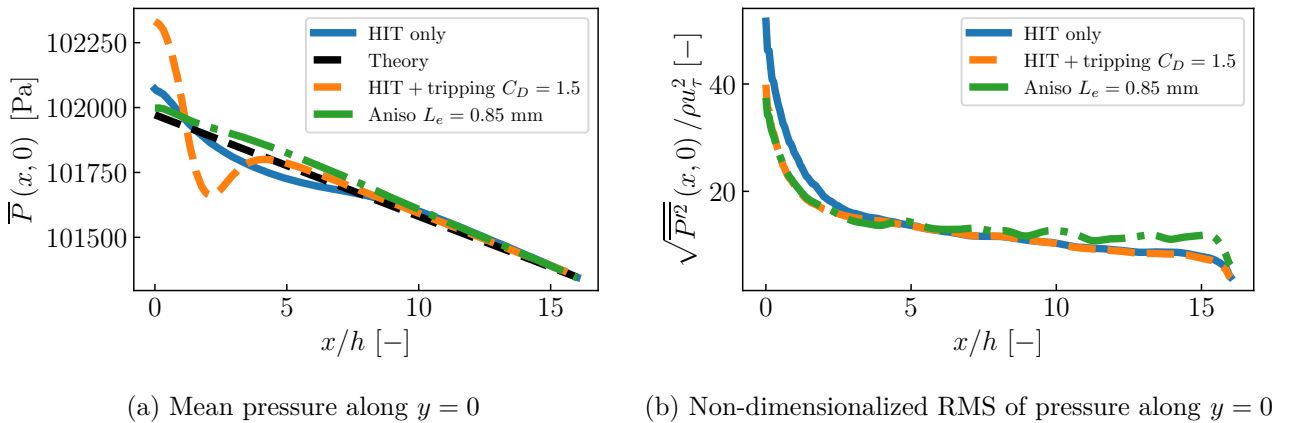


Figure 6.22. – Evolution of mean pressure and RMS of pressure at channel center ($y = 0$) along channel axis for HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous anisotropic turbulence with $L_e = 0.85$ mm cases.

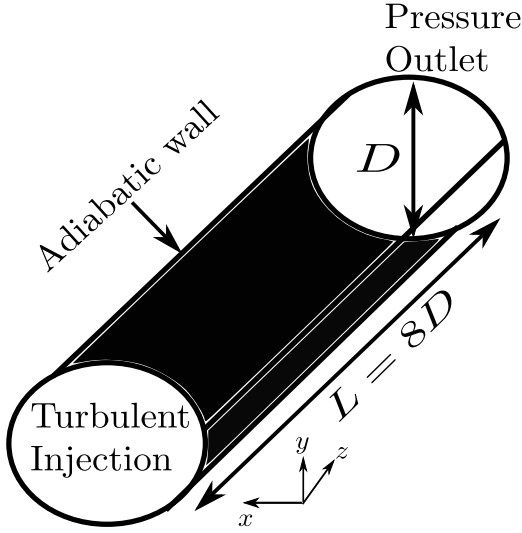
6.8. Turbulent wall-bounded pipe flow

Similar to what was done in Section 6.7, this section is dedicated to the simulation of a turbulent pipe flow using three approaches: HIT injection only, HIT injection coupled with artificial tripping and injection of non-homogeneous anisotropic turbulence. The turbulent pipe flow that is considered is the one simulated in DNS by El Khoury *et al.* [247] at $Re_b = 11700$ taken as reference for the results. Like for the turbulent channel flow case, the pipe length from which the flow is fully turbulent will be investigated.

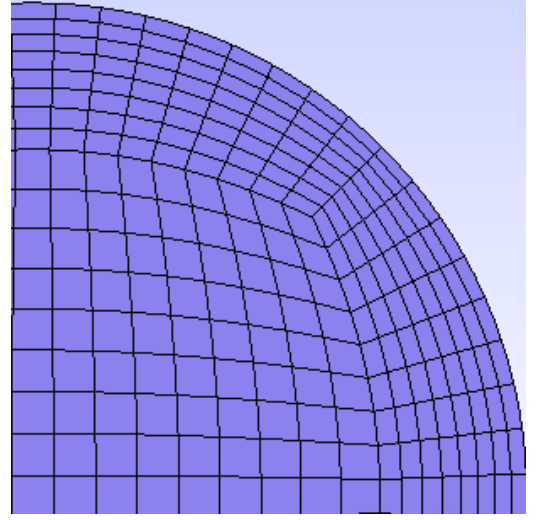
6.8.1. Numerical setup

The computational domain is a pipe of diameter $D = 2R$, with $R = 6.35$ mm, with a length of $L = 8D$. It is represented in Figure 6.23a with its associated boundary conditions set as follows:

- Turbulence injection through a subsonic NSCBC inlet is used at $z = 0$.
- Constant pressure at $P_{out} = 10^5$ Pa through a subsonic NSCBC outlet is used at $z = 8D$.
- Adiabatic no-slip wall is considered all along the side wall of the pipe where $r = R$.



(a) Computational domain and boundary conditions.



(b) View of a quarter-section of the mesh elements.

Figure 6.23. – Computational domain, boundary conditions and cross-sectional view of a quarter-section of the mesh elements used for the turbulent wall-bounded pipe flow.

The fluid is air ($R_{gas} = 287.15 \text{ J.kg}^{-1}.\text{K}^{-1}$) at $P = 10^5$ Pa and $T = 300$ K with $\mu = 1.716 \times 10^{-5} \text{ kg.m}^{-1}.\text{s}^{-1}$ so that $\nu \approx 1.477 \times 10^{-5} \text{ m}^2.\text{s}^{-1}$. Thus, the bulk velocity to be imposed at the inlet plane is deduced from Eq. (6.26):

$$U_b = \frac{Re_b \nu}{D} \approx 13.62 \text{ m.s}^{-1} \quad (6.98)$$

The value of Re_b gives also access to the friction factor by solving Eq. (6.31) which gives 0.03 in this case. Consequently, friction velocity can be computed from Eq. (6.30) ending up with 0.830 m.s^{-1} . The associated friction Reynolds number can be found using Eq. (6.17) with R instead of h for pipes:

$$Re_\tau = \frac{u_\tau R}{\nu} = 357 \quad (6.99)$$

which is extremely close to the value of 360 mentioned in El Khoury *et al.* [247]. Finally, the viscous length scale δ_ν is obtained from Eq. (6.15) to get 1.78×10^{-5} m. The mesh is designed to have the following characteristics:

$$r^+ = 1.25, R\theta^+ = 9 \text{ and } z^+ = 11 \quad (6.100)$$

A view of a quarter-section of the mesh elements (no SP are shown) is represented in Figure 6.23b where the mesh refinement in the wall normal direction (radial direction) can be seen. This cross-section is extruded along pipe direction using 130 uniformly distributed segments between $z = 0$ and $z = L$. The mesh was built for a polynomial degree set to $p = 3$ (order 4) in every elements. The final mesh is composed of 108290 second-order (to better capture the curvature of pipe walls) hexahedral elements which gives a total number of DOF around 6.9×10^6 when $p = 3$. All the simulations were run first for 3 convective times, based on L and U_b of this case, which corresponds to 22.38 ms of physical time. Then, from the restart solution at 22.38 ms, time-averages variables are collected during 8 convective times corresponding to 44.76 ms. These 8 convective times were sufficient for the statistics to be converged. The post-treatment of these statistics is done as follows:

1. The code again provides \bar{u} , \bar{v} , \bar{w} , $\overline{u^2}$, $\overline{v^2}$, $\overline{w^2}$, \overline{uv} , \overline{vw} and \overline{uw} at SP.
2. R^{cart} is computed at all SP using Eq. (6.93).
3. \bar{u} , \bar{v} , \bar{w} and R^{cart} at each SP are interpolated on a cylindrical mesh where the discretization is uniform along $r \in [0, R]$, $\theta \in [0, 2\pi]$ and $z \in [0, L]$.
4. Azimuthal average is performed on the uniform cylindrical mesh considering n_{cut} (defined by the user) cuts at different θ in $[0, 2\pi]$ where \bar{u}_r , \bar{u}_θ , \bar{u}_z and R^{cyl} are computed using Eq. (6.33) and Eq. (6.35).
5. At the end of the azimuthal averaging process, an average meridional plane of the pipe is obtained on which \bar{u}_r , \bar{u}_θ , \bar{u}_z and R^{cyl} have been azimuthally averaged and will be used to plot velocity profiles along r direction at different axial positions on the pipe.

The remaining numerical parameters for Riemann solver, diffusion scheme and time-integration procedure used for all the simulations are still the same as in Table 6.3.

6.8.2. Injection of HIT only at the inlet

In this paragraph, HIT is injected at the inlet of the pipe and superimposed to an injected axial mean flow given by Eq. (6.40) with $n = 6.86$ and $U_{cl} = 16.74 \text{ m.s}^{-1}$ both obtained by solving the system of equations provided by Eq. (6.41), Eq. (6.42) and Eq. (6.43) for $Re_b = 11700$. As for the wall-bounded channel flow case, the objective is to see the capability of HIT injection only to reproduce, after a certain distance from the inlet of the pipe, a turbulent pipe flow in the presence of walls. The reference turbulent pipe flow from which results will be compared is the DNS made by El Khoury *et al.* [247] at the same bulk Reynolds number. The turbulent parameters that have been selected for the injected HIT are summed up in Table 6.8. In this case, N was set to 300 since the mesh was

| Energy spectrum | N [-] | Mode distribution | L_e [mm] | L_{ref} [mm] | L_c [mm] | u_{rms} [m.s ⁻¹] |
|-----------------|---------|-------------------|------------|----------------|------------|--------------------------------|
| Passot-Pouquet | 300 | Logarithmic | 3.175 | 6.35 | 1.2 | 1.5 |

Table 6.8. – Turbulent parameters used for the HIT injection in a turbulent wall-bounded pipe flow.

composed of approximately two times more DOF than in the wall-bounded channel flow. However, in paragraph 6.6.4, $N = 300$ was shown sufficient to have a satisfactory HIT field at the inlet at moderate computational cost. Regarding the u_{rms} value, according to DNS results, RMS velocities at channel centers were around 0.83 m.s^{-1} for $\sqrt{w'^2}$ and around 0.62 m.s^{-1} for $\sqrt{u_r'^2}$ and $\sqrt{u_\theta'^2}$ so that a value

of u_{rms} around 0.9 m.s^{-1} or 1.0 m.s^{-1} was firstly expected to work. Nevertheless, after some tests it appeared that $u_{rms} = 1.5 \text{ m.s}^{-1}$, so around 11% of U_b , was a good choice. It can be due to the lower value of Re_b in this case: 11700 instead of 20000 for the wall-bounded channel flow so that the flow is harder to trigger to become turbulent. The smaller value of N might entail that the injected turbulence decays faster in the presence of walls compared to the same case but with $N \approx 1000$. This is something that could be investigated for future works. Characteristic sizes were chosen as follows:

- similarly to the wall-bounded channel case and based on previous studies using PP spectrum with walls [286, 287], L_e was set equal to $D/4$.
- again, similarly to the wall-bounded channel case, L_{ref} was set equal to R to have structures with a size of at most the pipe radius.
- L_{cut} was set equal to $6\Delta z$ since z direction has the coarser discretization and to have at least 6 SP in the smallest structures.

Radial profiles at different axial positions z are represented in Figure 6.24 and 6.25 respectively for the mean axial flow and the RMS velocities in cylindrical coordinates. DNS values are also represented for comparison. It should be mentioned that RMS velocities $\overline{u'_r u'_\theta}$ and $\overline{u'_\theta u'_z}$ are not shown since they are almost zero according to DNS results. As for the wall-bounded channel flow, a turbulent pipe flow is obtained after approximately $6D$ or $7D$ of pipe length using turbulent injection, explained in Section 6.4, in JAGUAR. The near-wall turbulence is well captured but there are still some differences with the DNS in the transition zone between the pipe center and the near-wall region. Like in the wall-bounded channel flow case, artificial tripping and anisotropic turbulent injection will be used in the next two paragraphs to see if they can better predict this transition zone.

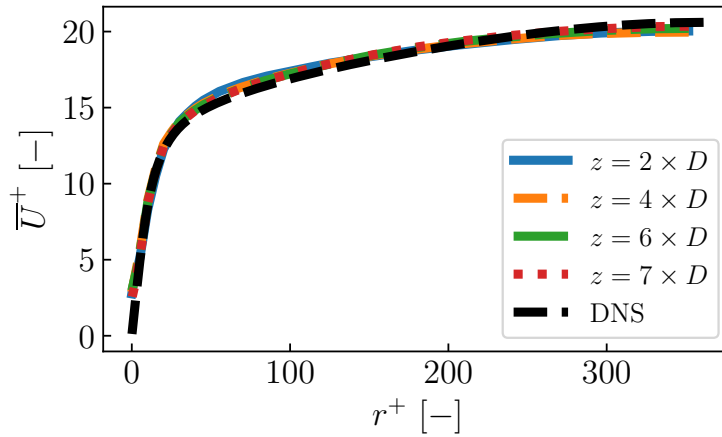


Figure 6.24. – Evolution of $\overline{U}^+ = \overline{w}/u_\tau$ as a function of r^+ at different axial positions in the pipe. DNS results of El Khoury *et al.* [247] are also represented for comparison.

6.8.3. Injection of HIT at the inlet coupled with artificial tripping

The same situation as the one shown in paragraph 6.8.2 is considered with the same injected mean flow and same turbulent injection parameters. In order to better reproduce the DNS results, especially in the transition zone between the pipe center and the near-wall region, a cylindrical artificial step, like the one shown in Figure 6.5b, will be added to the simulation. Similarly to what was done for the Cartesian steps in paragraph 6.7.3, the cylindrical artificial step is set as follows:

- it is located between $z = R$ and $z = R + 5\Delta_z^{\text{DOF}}$ to have at least 5 SP inside the steps axially.
- it has a height of 8% of R so that it will be applied for $r \in [0.92R, R]$.

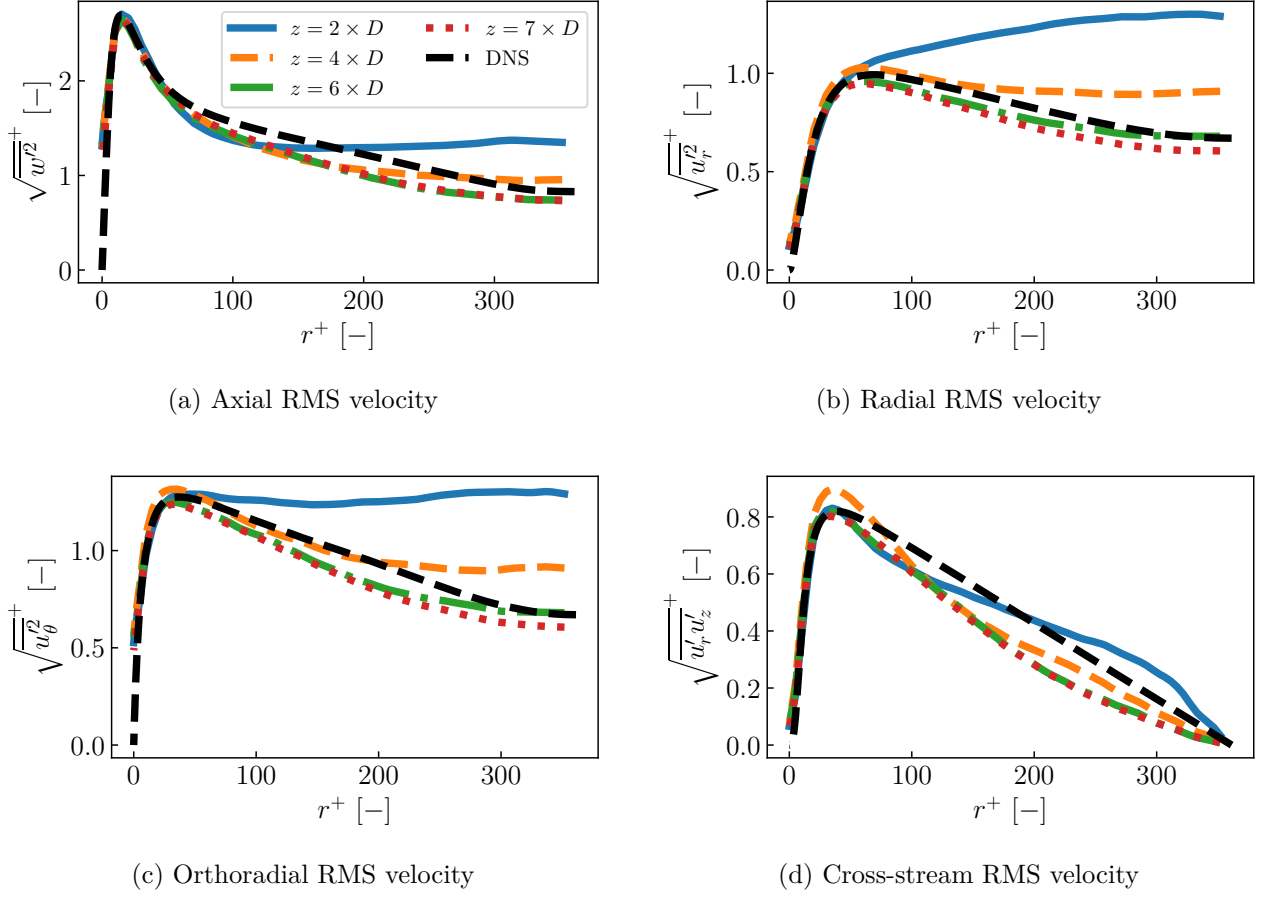


Figure 6.25. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at different axial positions in the pipe when only HIT is injected at pipe inlet. DNS results of El Khoury *et al.* [247] are also represented for comparison.

Thus, the only parameter remaining is the value of C_D introduced in Eq. (6.90) which needs to be calibrated. Thus, as for the wall-bounded channel flow, simulations have been performed using three different values of C_D : 1, 1.5 and 2. Results at $z = 7D$ are compared in Figures 6.26 and 6.27 with the ones obtained in paragraph 6.8.2 with only HIT injection. As for the wall-bounded channel flow, only RMS velocities are impacted by the artificial tripping treatment: the mean axial flow is almost the same according to Figure 6.26. These RMS velocities are significantly improved when using the artificial step for every value of C_D tested here. Therefore, like for the wall-bounded channel flow case, the use of artificial tripping, as a complement to HIT injection, helps to better reproduce a turbulent pipe flow.

6.8.4. Injection of non-homogeneous and anisotropic turbulence at the inlet

This paragraph studies the impact of using the injection of non-homogeneous and anisotropic turbulence based on the DNS profiles of El Khoury *et al.* [247] and its capability to reproduce a turbulent pipe flow around $z = 6 - 7D$. The same methodology as the one described in paragraph 6.5.1, and already applied in paragraph 6.7.4 for the wall-bounded channel flow, will be used. The only thing that differs is the computation of the Cholesky matrix which is slightly different because the DNS data are in cylindrical coordinates. It is computed as follows:

1. The DNS of El Khoury *et al.* [247] provides r^+ , $\overline{U}^+(r^+)$ and $R^{cyl,+}(r^+)$ where $R^{cyl,+} = R^{cyl}/u_\tau^2$.

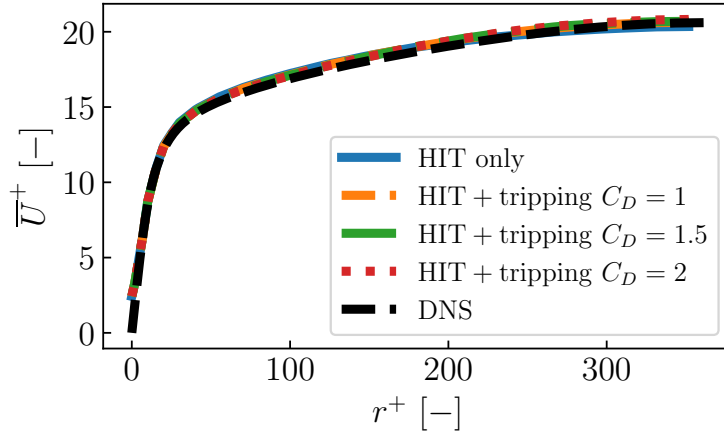
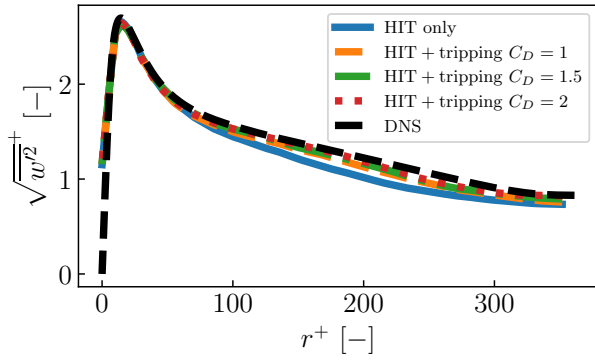
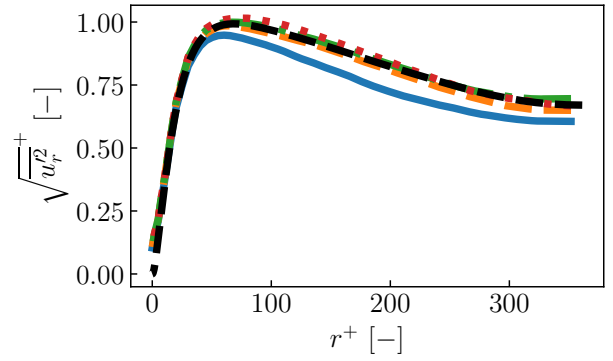


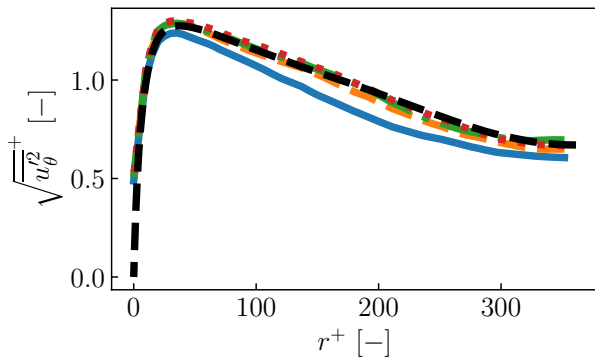
Figure 6.26. – Evolution of $\bar{U}^+ = \bar{w}/u_\tau$ as a function of r^+ at $z = 7D$ for three values of C_D . DNS results of El Khoury *et al.* [247] and results with only HIT injected at the inlet are also represented for comparison.



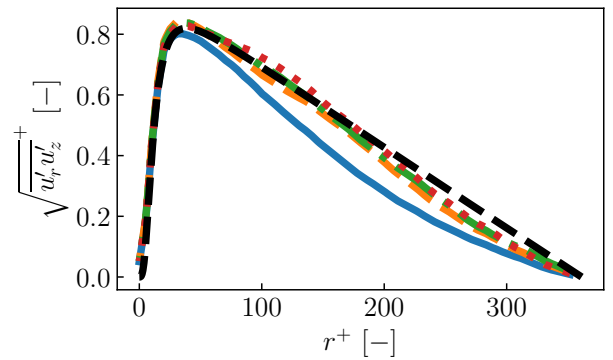
(a) Axial RMS velocity



(b) Radial RMS velocity



(c) Orthoradial RMS velocity



(d) Cross-stream RMS velocity

Figure 6.27. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at $z = 7D$ for three values of C_D . DNS results of El Khoury *et al.* [247] and results with only HIT injected at the inlet are also represented for comparison.

2. A python script with input data D, μ, R_{gas}, P, T and Re_τ converts $r^+, \bar{U}^+(r^+)$ and $R^{cyl,+}(r^+)$ into $r, \bar{w}(r)$ and $R^{cyl}(r)$.
3. At each inlet FP, $R^{cyl}(r)$ is interpolated and Eq. (6.35) is inverted to get R^{cart} at these FP.
4. Finally, Cholesky matrix A is computed from R^{cart} at each FP using Eq. (6.85).

The injected mean flow is the one obtained from DNS data and turbulent parameters are the same as in Table 6.8 except that three different values of L_e have been tested to see their impact on the results: $D/3$, $D/4$ (the value in Table 6.8) and $D/5$. Results at $z = 7D$ are summed up in Figures 6.28 and 6.29. As in the wall-bounded channel case with anisotropic injection, the mean flow is not very different. It can be noticed that with smaller values of L_e , the centerline velocity is slightly over-predicted. About the RMS velocities shown in Figure 6.29, $L_e = D/4$ gives the best results whereas for $L_e = D/3$ and $L_e = D/5$ RMS velocities are respectively over-predicted and under-predicted. Thus, in the case of the wall-bounded pipe, no trade-off has to be found on L_e since changing its value impact all the RMS profiles in the same manner. This is not a result that can be generalized since it probably depends on the value of Re_b and also on the turbulent spectrum that is used. However, for the situation considered here with a PP spectrum injected in a wall-bounded pipe at $Re_b = 11700$, it seems that $L_e = D/4$ is an optimal choice. It is able to reproduce a turbulent pipe flow after a distance around $7D$ and gives similar results to the injection of HIT coupled with artificial tripping using $C_D = 1.5$.

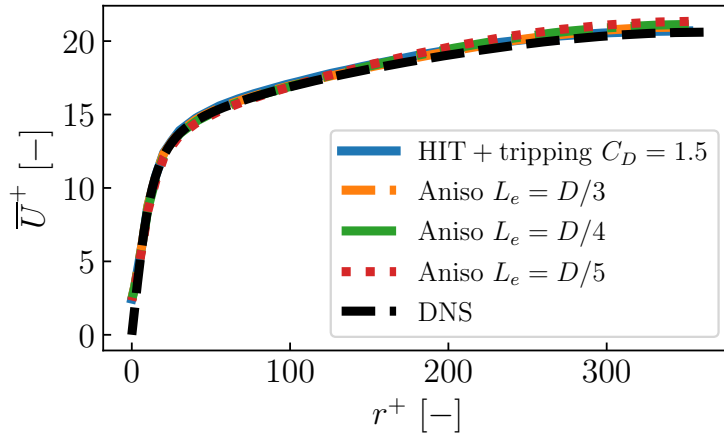


Figure 6.28. – Evolution of $\bar{U}^+ = \bar{w}/u_\tau$ as a function of r^+ at $z = 7D$ for three values of L_e when injecting non-homogeneous and anisotropic turbulence. DNS results of El Khoury *et al.* [247] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison.

6.8.5. Axial pressure gradient and acoustics in the pipe

As it was done for the wall-bounded channel flow in paragraph 6.8.5, the pressure gradient due to friction at walls can be investigated in the pipe to see if it is correctly reproduced. It can be shown that the mean axial pressure gradient at pipe center follows [150]:

$$\frac{\partial \bar{P}}{\partial z} = -2 \frac{\rho u_\tau^2}{R} \iff \bar{P}(z, 0) = P_{out} + 2(L_z - z) \frac{\rho u_\tau^2}{R} \quad (6.101)$$

where $P_{out} = 10^5$ Pa here. Figure 6.30a shows the evolution of $\bar{P}(z, 0)$ for three different injection cases previously studied: HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous

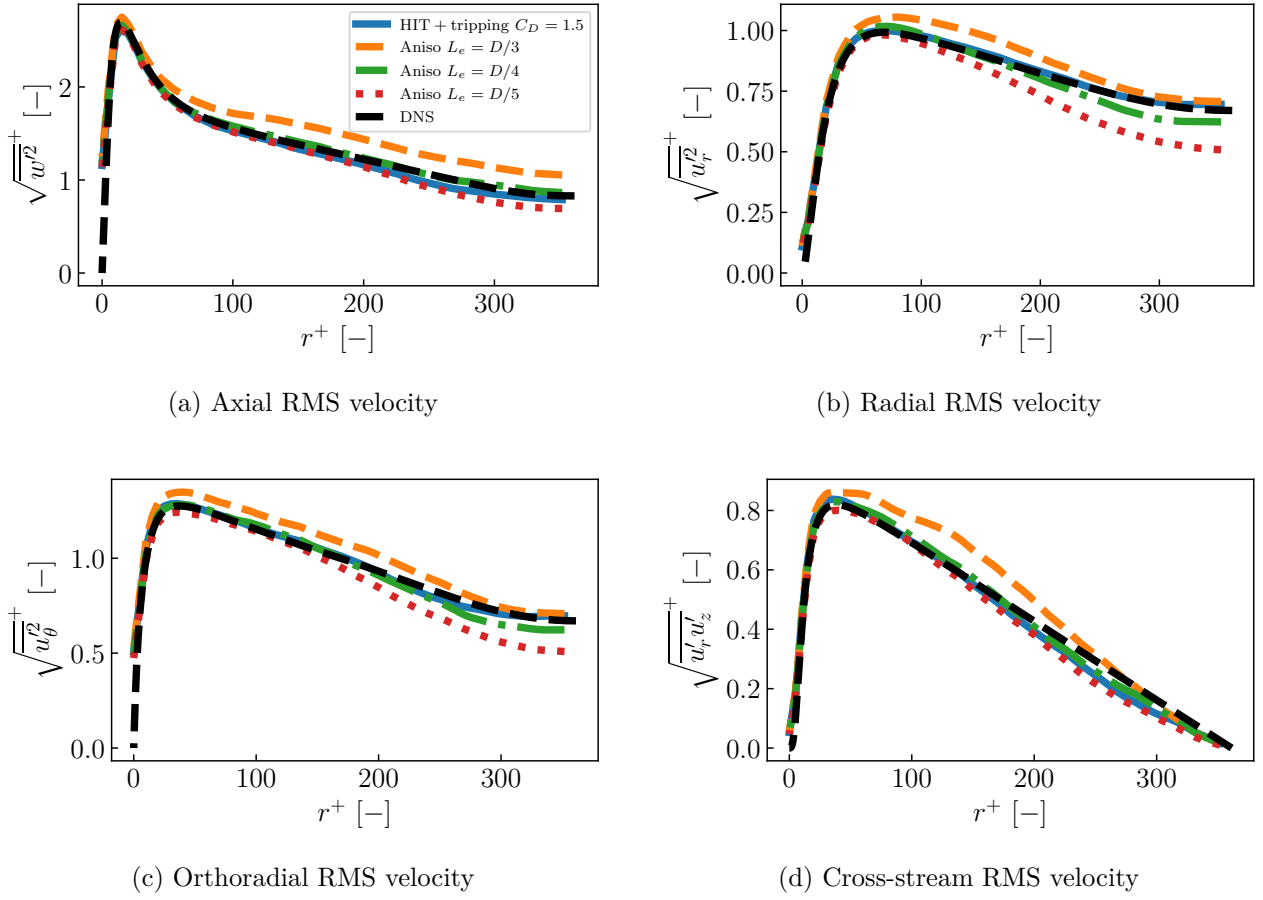
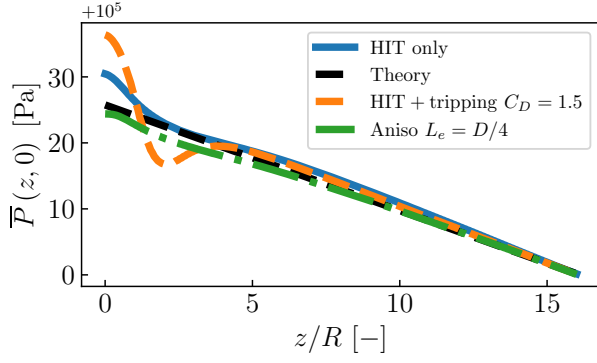
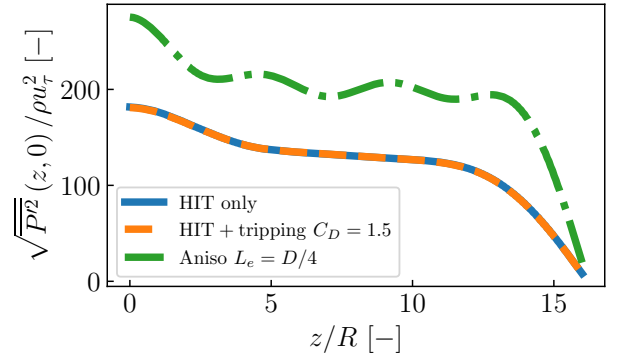


Figure 6.29. – Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at $z = 7D$ for three values of L_e when injecting non-homogeneous and anisotropic turbulence. DNS results of El Khoury *et al.* [247] and results with only HIT injected at the inlet are also represented for comparison.

anisotropic turbulence with $L_e = D/4$. The theoretical curve corresponds to Eq. (6.101) computed with $u_\tau = 0.830 \text{ m.s}^{-1}$ which is the expected value for $\text{Re}_\tau = 360$. In every cases, the expected evolution of $\bar{P}(z, 0)$ is retrieved very rapidly downstream of the inlet plane. As in the wall-bounded channel flow, the artificial tripping induces a deviation of the mean pressure around its location in $x = R$ but the flow rapidly goes back to the expected mean pressure downstream. Figure 6.30b presents the non-dimensionalized RMS of pressure along $r = 0$ within the channel giving an estimation of the noise generated by the injection of turbulence in the three cases: HIT only, HIT coupled with artificial tripping and non-homogeneous anisotropic turbulence. Like in the wall-bounded channel flow, all the three methodologies generate noise mostly at the inlet but this noise is at well higher levels than in the channel flow case. This time the noise is more important in the case of non-homogeneous and anisotropic turbulence which is to be expected since Eq. (6.74) is not satisfied in this case. Therefore, the wall-bounded pipe also shows that more work is needed to investigate the origin of this noise and how it can be reduced.



(a) Mean pressure along $r = 0$



(b) Non-dimensionalized RMS of pressure along $r = 0$

Figure 6.30. – Evolution of mean pressure and RMS of pressure at channel center ($r = 0$) along pipe axis for HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous anisotropic turbulence with $L_e = D/4$ mm cases.

6.9. Summary of this chapter

In this chapter, the general concepts of turbulent channel and pipe flows are first introduced showing that they are characterized by different flow regions from the viscous sublayer near the walls to the outer layer near channel or pipe center. In between, the buffer layer and the log-law regions lack of modeling and are still widely studied. Different methods exist in CFD to reproduce these kinds of flows and among them the injection of turbulence at the inlet of the computational domain is very efficient.

Several approaches for injecting turbulence are proposed in the literature. The one which is considered in this work is the synthetic random Fourier approach where turbulent fluctuations are generated based on a sum of random Fourier modes. This methodology was originally developed by Kraichnan [263] ending up with a divergence-free synthetic HIT field where turbulent modes are chosen stochastically and indirectly to satisfy a given energy spectrum. In this work, the extension of Kraichnan's work by Karweit *et al.* [269] and Bechara *et al.* [264], where turbulent modes are chosen deterministically and directly from a given energy spectrum, is employed since it is adapted to any energy spectrum function. The synthetic random Fourier method can be extended to inject non-homogeneous and anisotropic fluctuations to better reproduce near-wall turbulence. The addition of an artificial tripping approach can also be used for that purpose.

The injection of turbulence is first validated on a channel flow with an inlet and an outlet as axial boundary conditions whereas channel sides are set as periodic boundary conditions in order to be in HIT hypotheses where there are no losses due to wall friction. The expected injected HIT field was retrieved within the channel for two different energy spectrum functions and a study on the influence of the number of turbulent modes N , and of their distribution, on the results and computational time was carried out. It shows that for $N \geq 600$, the isotropic property of the HIT field is conserved, for the two energy spectra considered, but there is a trade-off with computational time which strongly increases when $N \geq 600$. The injection of turbulence is then tested for reproducing turbulent wall-bounded channel and pipe flows. In both cases, it performs well to reproduce turbulence at channel or pipe centers whereas near the walls and in the transition zone between walls and channel/pipe enters, turbulence is usually under-predicted. Therefore, either the extension of synthetic random Fourier method to inject non-homogeneous and anisotropic turbulence or the addition of an artificial tripping were tested and show improvement of the results in the near-wall region. The length required to reach

an established turbulent flow using this turbulent injection methodology in JAGUAR, was found to be around $6 \times 2h$ and $6 \times D$ respectively for turbulent channel and pipe flows. Finally, although the results for aerodynamic fields were very satisfying, both channel and pipe flows generate too much noise compared to what DNS results have shown. Therefore, more work is needed to reduce this noise which can come from many sources such as the boundary conditions or the discretization of turbulent modes due to the mesh.

Part III.

The Spectral Difference method and its application to a multi-species and reacting gas mixture

Extension of the Spectral Difference method to simulate a multi-species and reacting gas mixture

Overview

| | |
|---|------------|
| 7.1. Handling contact discontinuities for a thermally perfect gas with the SD method | 150 |
| 7.1.1. Discretization of a contact discontinuity with the SD method | 150 |
| 7.1.2. Simulations of a contact discontinuity using the SD method | 152 |
| 7.1.3. Discussion on the impact of using TUPY approach instead of CONS | 154 |
| 7.2. One-dimensional laminar premixed flames | 155 |
| 7.2.1. One-dimensional methane/air flame using a two-reactions chemistry | 155 |
| 7.2.2. One-dimensional methane/air flame using Analytically Reduced Chemistry . . | 157 |
| 7.2.3. One-dimensional hydrogen/air flame using a skeletal mechanism | 159 |
| 7.2.4. Conclusion on one-dimensional laminar premixed flames results | 161 |
| 7.3. Two-dimensional laminar premixed flames | 162 |
| 7.3.1. Two-dimensional methane/air burner computed with a two-reactions chemistry | 162 |
| 7.3.2. Two-dimensional methane/air circular flame computed with a two-reactions chemistry | 164 |
| 7.3.3. Conclusion on two-dimensional laminar premixed flames results | 167 |
| 7.4. Evaluation of p-refinement on combustion cases | 167 |
| 7.4.1. Context of the study | 168 |
| 7.4.2. Test case presentation and reference solution | 168 |
| 7.4.3. Influence of p on the accuracy of the solution in uniform p cases | 169 |
| 7.4.4. Influence of p on the computational cost in uniform p cases | 171 |
| 7.4.5. Local p -adaptation | 173 |
| 7.4.6. Conclusions of the study | 175 |
| 7.5. Comparison between original and SDLIFT formulations for the diffusion scheme on 1D and 2D premixed flames using coarse meshes | 176 |
| 7.5.1. 1D methane/Air premixed flame with four points inside the flame front using the original formulation of the diffusion scheme | 176 |
| 7.5.2. 1D methane/Air premixed flame with four points inside the flame front using SDLIFT formulation for the diffusion scheme | 178 |
| 7.5.3. Impact of the SDLIFT formulation with ARC schemes | 178 |
| 7.5.4. Impact of the SDLIFT formulation on a 2D methane/air circular flame | 179 |
| 7.6. Summary of this chapter | 181 |

This chapter is dedicated to the simulation of one-dimensional and two-dimensional laminar premixed flames using the Spectral Difference (SD) method. Firstly, the classical SD algorithm is shown to be unstable in the presence of a thermally multi-species perfect gas. A modification of the SD algorithm is then proposed to correct this non-physical behavior. This allows to simulate several laminar one-dimensional and two-dimensional premixed flames, with different chemical schemes, as validation test cases. A first evaluation of the interest and the potential of using the SD method in combustion is also investigated on these test cases. Finally, the impact of using the SDLIFT formulation for diffusion schemes is studied on very coarse discretizations of several flames.

7.1. Handling contact discontinuities for a thermally perfect gas with the SD method

In paragraph 1.3.1, it was explained that multi-species mixtures, with thermodynamic functions depending on T and Y_k may lead to numerical instability, in particular in reacting flows where the energy equation is required. This issue has been observed in previous works with FV [75], DG [25, 27, 85] and also FR [92] methods, and is therefore also expected with SD. The objective of this section is to evaluate and solve this problem with SD.

7.1.1. Discretization of a contact discontinuity with the SD method

Consider the case of a contact discontinuity as described in [75] where the 1D Euler equations for a multi-species gas (1D case of Eq. (F.1)) are solved with a flow initially at constant pressure P_c and constant velocity u_c but with a temperature or/and composition gradient in the domain. In this situation, in theory the pressure and velocity remain constant. Denoting $H_a(t)$ the theoretical solution at time instant t , verified at SP, the question is to determine at which conditions H_a is verified at SP at the next time instant $t + \Delta t$. The conservative variables at SP are advanced in time using Eq. (4.21) which is recalled here for clarity:

$$\frac{d\widehat{\mathbf{U}}}{dt}(\xi_j) + \sum_{k=1}^{N_{FP}^{1d}} \mathbf{E}_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = \mathbf{0}, \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (7.1)$$

where \mathbf{E}_k here replaces $\widehat{\mathbf{E}}_k$ since $|J|J^{-1} = 1$ in 1D. As \mathbf{E}_k is computed from values at FP, these must preserve H_a . In other words, the interpolation from SP to FP must preserve constant pressure and velocity. One easy way to guarantee this is to interpolate directly P and u instead of ρu and ρE . For the two remaining variables, Y_k is interpolated instead of ρY_k and ρ can be interpolated as usual - or the temperature can be taken instead. Thus, H_a is verified also at FP and writing mass, momentum and energy equations from Eq. (7.1) gives:

1. Conservation of mass

$$\frac{d\widehat{\rho}}{dt}(\xi_j) + \sum_{k=1}^{N_{FP}^{1d}} \rho_k u_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \implies \frac{d\widehat{\rho}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \rho_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (7.2)$$

since $u_k = u_c$ for $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$.

2. Conservation of momentum

$$\frac{d\widehat{\rho u}}{dt}(\xi_j) + \sum_{k=1}^{N_{FP}^{1d}} (\rho_k u_k^2 + P_k) \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \implies \frac{d\widehat{\rho u}}{dt}(\xi_j) + u_c^2 \sum_{k=1}^{N_{FP}^{1d}} \rho_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) + P_c \sum_{k=1}^{N_{FP}^{1d}} \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (7.3)$$

since $P_k = P_c$ for $k \in \llbracket 1, N_{FP}^{1d} \rrbracket$.

3. Conservation of energy

$$\begin{aligned} \frac{d\widehat{\rho E}}{dt}(\xi_j) + \sum_{k=1}^{N_{FP}^{1d}} u_k (\rho_k E_k + P_k) \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \implies \frac{d\widehat{\rho E}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \rho_k E_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) \\ + u_c P_c \sum_{k=1}^{N_{FP}^{1d}} \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \end{aligned} \quad (7.4)$$

Since $\sum_{k=1}^{N_{FP}^{1d}} l_k^{FP}(\xi) = 1 \quad \forall \xi \in [0, 1] \Rightarrow \sum_{k=1}^{N_{FP}^{1d}} \left[\partial l_k^{FP} / \partial \xi \right](\xi_j) = 0$ for $j \in \llbracket 1, N_{SP}^{1d} \rrbracket$ so that Eq. (7.3) (respectively Eq. (7.4)) turns into Eq. (7.5) (respectively Eq. (7.6)):

$$\frac{d\widehat{\rho u}}{dt}(\xi_j) + u_c^2 \sum_{k=1}^{N_{FP}^{1d}} \rho_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (7.5)$$

$$\frac{d\widehat{\rho E}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \rho_k E_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (7.6)$$

Injecting Eq. (7.2) into Eq. (7.5) ends up with:

$$\frac{d\widehat{\rho u}}{dt}(\xi_j) - u_c \frac{d\widehat{\rho}}{dt}(\xi_j) = 0 \iff \frac{d\widehat{u}}{dt}(\xi_j) = 0, \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (7.7)$$

Eq. (7.7) shows that velocity remains constant at $t + \Delta t$ satisfying half of H_a . Concerning the energy equation in Eq. (7.6), an expression of ρE as a function of P is needed. For a thermally perfect gas, ρE can be expressed following Eq. (A.19) provided that the heat capacity ratio is set to $\tilde{\gamma}$ defined in Eq. (A.24) so that $\rho_k E_k$ is now:

$$\rho_k E_k = \frac{P_k}{\tilde{\gamma}_k - 1} + \rho_k \frac{u_k^2}{2} = \Gamma_k P_c + \rho_k \frac{u_c^2}{2} \quad (7.8)$$

where $\Gamma = 1/(\tilde{\gamma} - 1)$. Using Eq. (7.8) into Eq. (7.6) entails:

$$\frac{d\widehat{\rho E}}{dt}(\xi_j) + \frac{u_c^3}{2} \sum_{k=1}^{N_{FP}^{1d}} \rho_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) + u_c P_c \sum_{k=1}^{N_{FP}^{1d}} \Gamma_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0 \quad (7.9)$$

Thanks to Eq. (A.19), the time derivative of $\widehat{\rho E}(\xi_j)$ is expressed as:

$$\frac{d\widehat{\rho E}}{dt}(\xi_j) = \widehat{\Gamma}(\xi_j) \frac{d\widehat{P}}{dt}(\xi_j) + P_c \frac{d\widehat{\Gamma}}{dt}(\xi_j) + \frac{u_c^2}{2} \frac{d\widehat{\rho}}{dt}(\xi_j) + \underbrace{\rho u_c \frac{d\widehat{u}}{dt}(\xi_j)}_{=0} \quad (7.10)$$

Consequently, putting Eq. (7.10) into Eq. (7.9), the energy equation becomes:

$$\widehat{\Gamma}(\xi_j) \frac{d\widehat{P}}{dt}(\xi_j) + P_c \left[\frac{d\widehat{\Gamma}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \Gamma_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) \right] + \underbrace{\frac{u_c^2}{2} \left[\frac{d\widehat{\rho}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \rho_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) \right]}_{=0} = 0 \quad (7.11)$$

Therefore, to have $\left[d\widehat{P}/dt \right](\xi_j) = 0$ for $j \in \llbracket 1, N_{SP}^{1d} \rrbracket$, $\Gamma(\xi_j)$ must satisfy:

$$\frac{d\widehat{\Gamma}}{dt}(\xi_j) + u_c \sum_{k=1}^{N_{FP}^{1d}} \Gamma_k \frac{\partial l_k^{FP}}{\partial \xi}(\xi_j) = 0, \text{ for } j \in \llbracket 1, N_{SP}^{1d} \rrbracket \quad (7.12)$$

From Eq. (7.12), it can be concluded that:

- a mixture with constant Γ (calorically perfect gas case) always fulfills $\left[d\hat{P}/dt \right] (\xi_j) = 0$ for $j \in \llbracket 1, N_{SP}^{1d} \rrbracket$ as already said in the beginning of this paragraph.
- in the general case where $\Gamma = \Gamma(T, Y_k)$ does not verify Eq. (7.12) and H_a is not satisfied at $t + \Delta t$.

Although changing the interpolation from SP to FP does not strictly allow to verify H_a , it is clear from the above derivations that it will improve the stability of the numerical algorithm, at the cost of a minor modification of the classical SD algorithm. To evaluate this improvement, simulations of a contact discontinuity with the classical SD algorithm and with the modified SD algorithm are presented in the next paragraph.

7.1.2. Simulations of a contact discontinuity using the SD method

Two different approaches for the interpolation of variables from SP to FP are defined:

- Approach **CONS**: $\hat{\mathbf{U}}_{SP}^e \rightarrow \mathbf{U}_{FP}^e$ with $\hat{\mathbf{U}}_{SP}^e$ (respectively \mathbf{U}_{FP}^e) being the conservative variables at SP (respectively FP) in the isoparametric (respectively physical) domain. \mathbf{U}_{FP}^e are used at FP to compute all needed variables for flux computations. This is the usual discretization process in SD.
- Approach **TUPY**: $\hat{\mathbf{U}}_{SP}^e \rightarrow \mathbf{Q}_{SP}^e \rightarrow \mathbf{Q}_{FP}^e$ where $\mathbf{Q} = (T, \mathbf{u}, P, Y_k)^T$ is the vector of primitive variables with T instead of ρ as first variable. \mathbf{Q}_{FP}^e is used at FP to compute all needed variables for flux computations.

These two formulations are tested in a 1D case with air ($Y_{O_2} = 0.23$ and $Y_{N_2} = 0.77$), at $T = 300$ K on the left side and $T = 2010$ K on the right side. Initial pressure and velocity are constant respectively at $P_c = 101325$ Pa and $u_c = 0.2815$ m.s⁻¹. The 1D domain is discretized using $N_e = 96$ elements with $p = 3$ set inside each of them. This case is illustrated on Figure 7.1 where values of T , u and P at SP are shown. Characteristic boundary conditions for a subsonic inlet, imposing $T^{in} = 300$ K, $u^{in} = u_c$, $Y_{O_2}^{in} = 0.23$ and $Y_{N_2}^{in} = 0.77$, and for a subsonic outlet imposing $P^{out} = P_c$ are applied. The

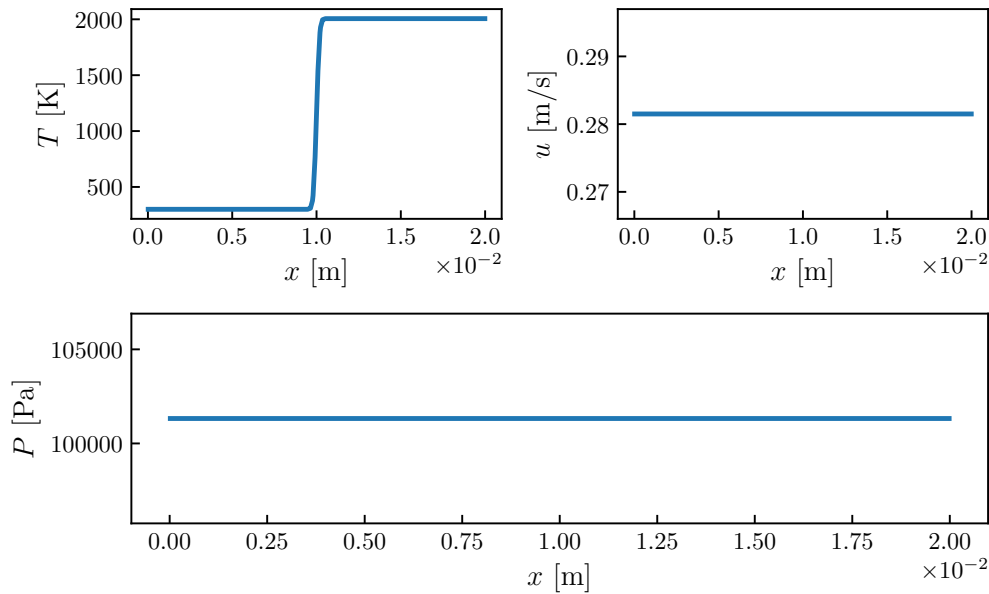


Figure 7.1. – Initial values of T , u and P at SP for both CONS and TUPY approaches.

simulation is carried out without diffusion fluxes and combustion source terms to compute only the 1D multi-species, non-reacting Euler equations. As expected, with the CONS approach important wiggles

appear on the pressure (and consequently on velocity) profile in the temperature transition zone as depicted in Figure 7.2. Other values of N_e and p have been tested to see if the mesh discretization can change this behavior, especially with finer meshes, but observations were the same. On the contrary, such wiggles are not visible with the TUPY approach on the initial discretization with $N_e = 96$ and $p = 3$. For further confirmation, a similar mono-species case was also computed with the CONS approach, showing a stable behavior as expected. Actually, very small pressure oscillations of the order of 0.1 Pa are present with the TUPY approach in the temperature transition zone, as Eq. (7.12) is not strictly verified. These oscillations are however extremely small, and considered acceptable for the targeted combustion applications.

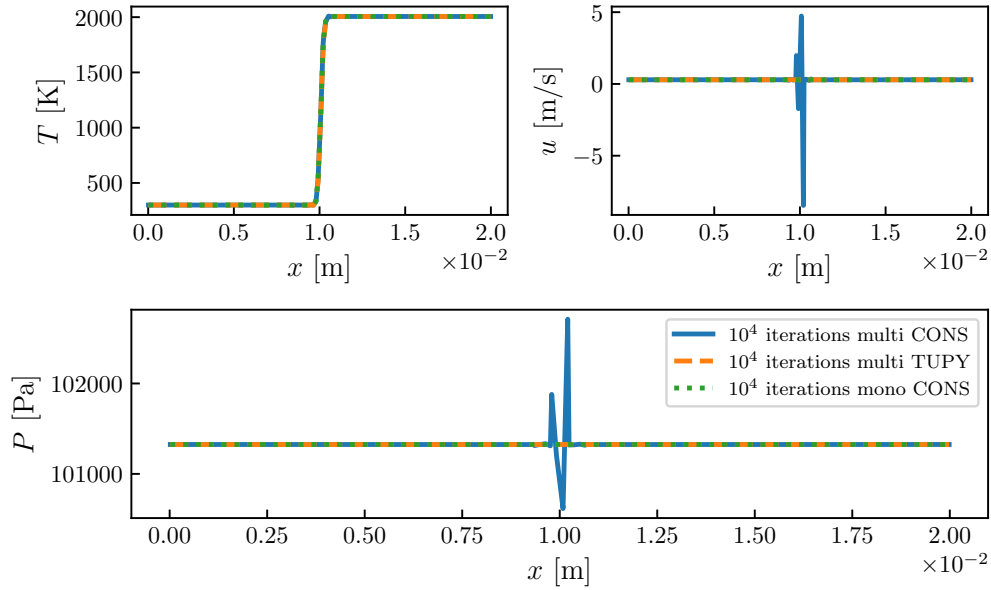


Figure 7.2. – Profiles of T , u and P at SP after 10^4 iterations for multi-species cases with either CONS or TUPY approaches, and mono-species case with CONS approach.

For further understanding, values of T , u and P are represented in Figure 7.3 at initialization on ten points equally spaced per mesh element, called output points (OP), for both the mono-species case and the multi-species case. This allows to see values of T , u and P at other points than SP, evaluated using the continuous polynomials built from SP. Interestingly, wiggles appear on the pressure profile in the transition zone between cold and hot air with the multi-species CONS approach whereas a perfect constant pressure profile is kept with the two other cases.

These results confirm that for the multi-species case, interpolating and building polynomials with conservative variables create pressure oscillations, whereas working with P and T at SP does not. Nevertheless, the case presented here has a low velocity so that the advection term in Eq. (7.12) is quite small. For larger values of u_c , larger pressure oscillations are expected even for the TUPY approach, as illustrated on Figure 7.4 for the exact same case with $u_c = 20 \text{ m.s}^{-1}$. In that case, additional numerical treatments will be necessary such as those already mentioned in paragraph 1.3.1 [75–84].

For a flame, u_c corresponds to the speed at which the flame front propagates which is the laminar flame speed S_L^0 . For most hydrocarbons: $0.1 \text{ m.s}^{-1} \leq S_L^0 \leq 2 \text{ m.s}^{-1}$. Thus, the TUPY approach without additional numerical treatment should be sufficient for the targeted applications of this work. It was the choice made here and in all simulated cases it did not lead to strong stability issues probably due to the small values of S_L^0 used. Finally, it should be mentioned that another approach interpolating ρ instead of T at FP, called ρ UPY was also tested and found much more stable than the CONS approach and with similar results than the TUPY approach. Some results using the ρ UPY

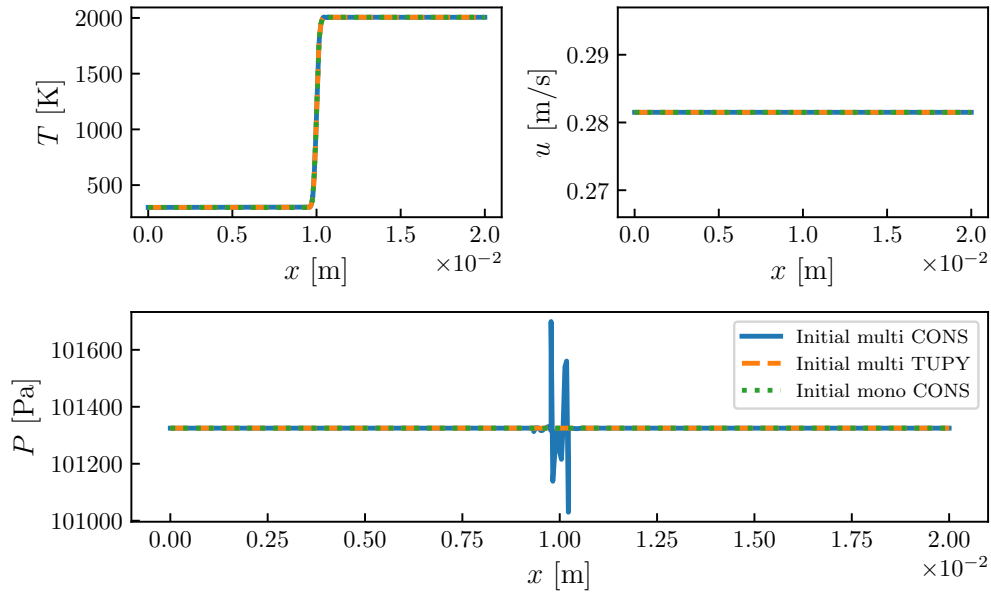


Figure 7.3. – Initial profiles of T , u and P at OP for multi-species cases with either CONS or TUPY approaches and mono-species case with CONS approach.

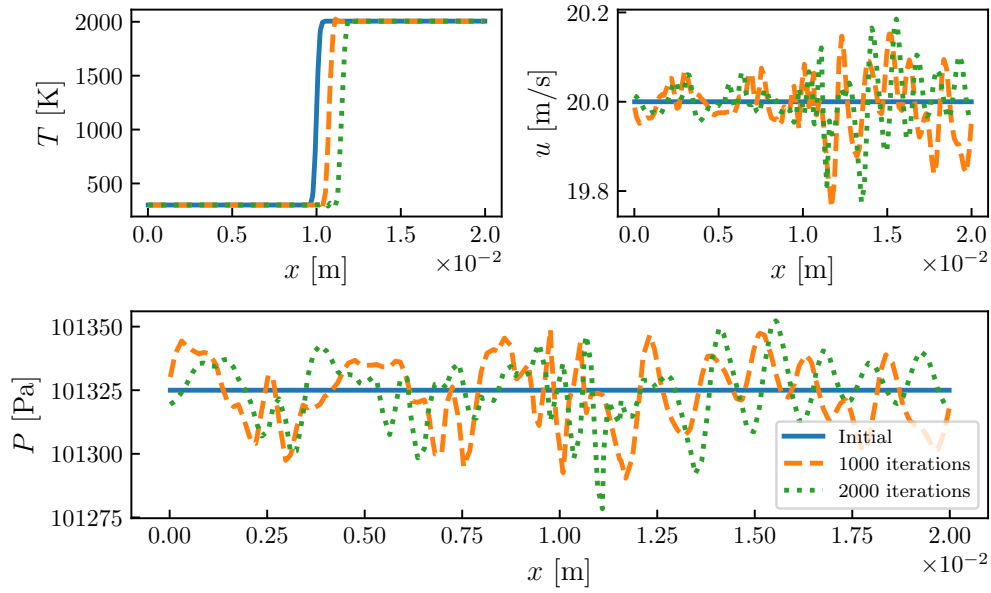


Figure 7.4. – Profiles of T , u and P at SP at 0, 1000 and 2000 time iterations for the multi-species case with the TUPY approach when $u_c = 20 \text{ m.s}^{-1}$.

approach can be found in paragraph 7.5.1.

7.1.3. Discussion on the impact of using TUPY approach instead of CONS

The TUPY approach is well more stable when considering a contact discontinuity simulation with a thermally perfect gas compared to the CONS approach. A first thing to notice is that, in both approaches, the conservative variables are still stored and advanced in time at SP. It is essential to

ensure the conservativity of the discretization so that Eq. (4.25) still holds. However, when employing the TUPY approach there are still open questions such as:

1. By construction \mathbf{U}_{SP} is a p -degree polynomial.
 $\implies \mathbf{Q}_{SP}$ is not probably a polynomial due to non-linear relations between \mathbf{U} and \mathbf{Q} . Nevertheless, \mathbf{Q}_{SP} is used to build a p -degree polynomial for \mathbf{Q} at SP used after to evaluate \mathbf{Q}_{FP} .
 $\implies \mathbf{Q}_{FP}$ is obtained from a function that is directly a polynomial which is not the case in CONS approach since \mathbf{Q}_{FP} is deduced from \mathbf{U}_{FP} .
 It is possible that these considerations does not play a big role because at the end: \mathbf{Q}_{FP} is used to compute fluxes \mathcal{F} at FP, \mathcal{F}_{FP} , which are used to build a $(p + 1)$ -degree polynomial for fluxes in both approaches.
2. For viscous fluxes, $\nabla \mathbf{Q}$ is needed at FP. TUPY computes $\nabla \mathbf{Q}$ directly from \mathbf{Q}_{FP} whereas CONS computes $\nabla \mathbf{U}$ and reconstruct $\nabla \mathbf{Q}$.
 \implies These differences of algorithm were shown to not being equivalent in DG and FR frameworks although the differences on the results remain small [289]. It might also be the case for the SD discretization.

7.2. One-dimensional laminar premixed flames

To validate the implementation of reacting flow equations in JAGUAR, simulations of 1D premixed flames have been performed. In this section, JAGUAR results are compared with results obtained using the reference solver AVBP [283, 284] developed by CERFACS which solves exactly the same reacting NSE for a multi-species gas with the same transport and chemistry models. All AVBP simulations were carried out using the TTGC scheme [179] for convective fluxes without artificial viscosity for a fair comparison with JAGUAR. For diffusive fluxes, AVBP uses a finite element scheme of order 2 [179]. 1D cases are also compared to the reference chemistry code CANTERA [126]. The objective of this section is to validate in JAGUAR:

- The combustion source terms and species transport computations with a two-reactions scheme.
- The implementation of JANAF thermochemical tables.
- The inlet and outlet NSCBC in a 1D multi-species case.

For all JAGUAR simulations presented in this section, the average approach for the diffusion scheme (see paragraph 4.4.1) was employed along with the SSP-ERK(3,3) scheme of Gottlieb and Shu [222] for time integration (see Section 4.6). The left boundary condition is a NSCBC subsonic inflow imposing $u^{in} = S_L^0$, $T^{in} = T^f$ and Y_k^{in} such that $\phi^{in} = \phi$ at the inlet. This inlet boundary condition leads to a steady flame as the inlet velocity is exactly equal to the flame speed S_L^0 . The right boundary is a NSCBC subsonic outflow imposing a static pressure P^{out} . Finally, all JAGUAR and AVBP simulations were initialized using a CANTERA solution and were run over a physical time of 0.1 seconds for methane-air flames and 0.01 seconds for the hydrogen-air flame. These simulation times ensured to reach a steady state.

7.2.1. One-dimensional methane/air flame using a two-reactions chemistry

7.2.1.1. Presentation of the case and numerical setup

A 1D CH_4/Air premixed flame is considered. The chemical scheme is the two-reactions CH_4/Air -2S-BFER scheme developed by Franzelli *et al.* [130] and already introduced in paragraph 2.3.2. The characteristics of this flame are given in Table 7.1 where S_L^0 and δ_L^0 have been obtained from the CANTERA solution at $\phi = 0.8$. In particular, δ_L^0 was obtained from the temperature profile of the CANTERA solution using Eq. (2.33). The computational domain is a 1D segment of length $L_x = 0.02$ m. To allow a fair comparison between JAGUAR and AVBP, all calculations are performed

| ϕ [-] | T^f [K] | T^b [K] | P [Pa] | δ_L^0 [m] | S_L^0 [cm.s ⁻¹] |
|------------|-----------|-----------|----------|-----------------------|-------------------------------|
| 0.8 | 300 | 2010 | 101325 | 4.30×10^{-4} | 28.15 |

Table 7.1. – Characteristics of the 1D methane-air premixed flame using the CH₄/Air-2S-BFER scheme. T^b , S_L^0 and δ_L^0 are the values given by CANTERA.

with similar number of points, i.e., same number of DOF. In AVBP, a DOF corresponds to a cell node whereas in JAGUAR, it corresponds to a SP. In space dimension d for hexahedral elements at the same degree p , the number of DOF is given by Eq. (4.74) for JAGUAR. Here the number of DOF is set around 400 in order to have a sufficient number of DOF in δ_L^0 , around 9. While this imposes a fixed number of 400 nodes in AVBP, JAGUAR has the possibility to use different values of N_e depending on the polynomial degree p to keep the DOF value close to 400 as summed up in Table 7.2.

| p | 3 | 4 | 5 | 6 |
|-------|-----|----|----|----|
| N_e | 100 | 80 | 67 | 57 |

Table 7.2. – Values of N_e for each degree p to keep the DOF number around 400.

7.2.1.2. Results

After a transient phase due to the transition from a constant pressure (CANTERA solution) to a compressible solution, convergence is reached and final profiles of density, velocity, pressure and temperature are represented in Figure 7.5 for CANTERA, AVBP and JAGUAR with $p = 4$ and $p = 6$. Major species mass fractions are plotted in Figure 7.6 for CANTERA and JAGUAR at $p = 4$ only since JAGUAR at $p = 6$ and AVBP show the same results.

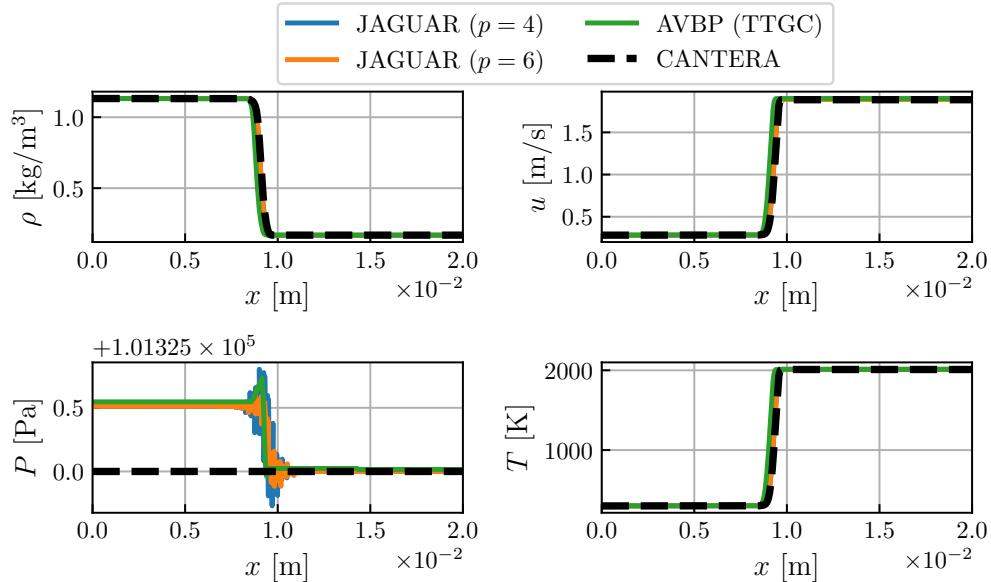


Figure 7.5. – Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH₄/Air-2S-BFER chemical scheme at $\phi = 0.8$.

All 4 profiles are in excellent agreement. The pressure jump through the flame front is captured by both JAGUAR and AVBP while CANTERA runs at constant pressure. Theoretically, this pressure

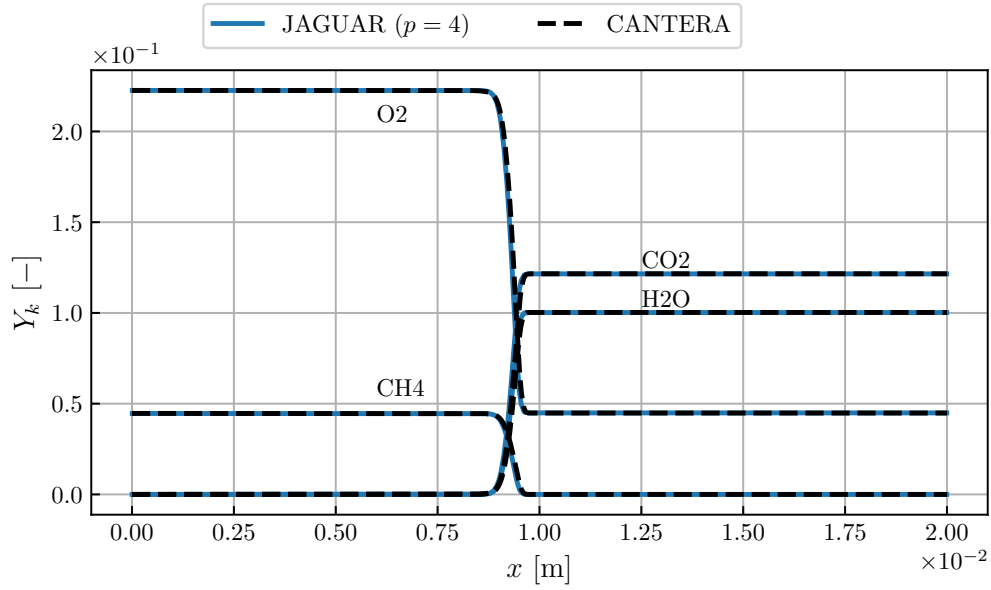


Figure 7.6. – Comparison of major species mass fractions profiles between CANTERA and JAGUAR at $p = 4$ for a 1D methane-air premixed flame using the CH4/Air-2S-BFER chemical scheme at $\phi = 0.8$.

drop is given by [3]:

$$P^b - P^f = \rho^f (S_L^0)^2 \left(1 - \frac{T^b}{T^f} \right) \quad (7.13)$$

where P^b and P^f are respectively the pressure in burnt and fresh gases. Using values of Table 7.1, Eq. (7.13) leads to a pressure drop of -0.511 Pa, i.e., very close to the value of -0.512 Pa measured from JAGUAR and AVBP solutions. Finally flame speeds based on the consumption speed (see Eq. (2.29)), are compared in Table 7.3 for both AVBP and JAGUAR solutions where ϵ_{rel} is the relative error compared to the CANTERA reference value given in Table 7.1. Flame speeds estimated by JAGUAR are in very

| Code | AVBP | JAG ($p = 3$) | JAG ($p = 4$) | JAG ($p = 5$) | JAG ($p = 6$) |
|-------------------------------|-------|-----------------|-----------------|-----------------|-----------------|
| S_L^0 [cm.s ⁻¹] | 28.31 | 28.17 | 28.16 | 28.16 | 28.15 |
| ϵ_{rel} in % | 0.568 | 0.071 | 0.036 | 0.036 | 0.0 |

Table 7.3. – Comparison of flame speed obtained with AVBP and JAGUAR (abbreviated by JAG) at different polynomial degrees p for a 1D methane-air premixed flame using the CH4/Air-2S-BFER chemical scheme at $\phi = 0.8$.

good agreement with the value given by CANTERA which demonstrates its capability to well-capture the flame front. Note that the error is always smaller with JAGUAR than with AVBP, and decreases for higher p values, showing the benefit of high-order methods.

7.2.2. One-dimensional methane/air flame using Analytically Reduced Chemistry

7.2.2.1. Presentation of the case and numerical setup

The objective of this test case is to simulate the previous 1D premixed flame with JAGUAR using a more detailed chemistry, including more species to transport than the two-reactions mechanism employed in paragraph 7.2.1. The chemical scheme considered here is the CH4_16_250_10_QC

ARC mechanism already mentioned in paragraph 2.3.3. It is composed of 16 transported species, 250 chemical reactions and 10 species in QSS. These QSS species have a very small characteristic timescale so that their net chemical source term is considered equal to zero. Their concentration is then computed from the concentrations of the other species, without solving a transport equation which reduces the computational time. The characteristics of this flame are given in Table 7.4. Note that for this case the equivalence ratio is $\phi = 1$. The flame thickness is almost the same as the one

| ϕ [-] | T^f [K] | T^b [K] | P [Pa] | δ_L^0 [m] | S_L^0 [cm.s ⁻¹] |
|------------|-----------|-----------|----------|-----------------------|-------------------------------|
| 1.0 | 300 | 2210 | 100000 | 4.39×10^{-4} | 37.86 |

Table 7.4. – Characteristics of the 1D methane-air premixed flame using an ARC mechanism. T^b , S_L^0 and δ_L^0 are the values given by the CANTERA solution.

obtained with the CH4/Air-2S-BFER at $\phi = 0.8$ already simulated in paragraph 7.2.1. Therefore the same 1D segment of length $L_x = 0.02$ m is used and discretized with the same number of DOF, here with $N_e = 80$ elements and a polynomial degree set to $p = 4$. Note that in the general case, ARC schemes may be stiffer than global chemical schemes due to the presence of fast intermediate species. This is however not the case for methane-air combustion.

7.2.2.2. Results

Final profiles of density, velocity, pressure and temperature are represented in Figure 7.7 and some species mass fractions profiles are represented in Figures 7.8 and 7.9. JAGUAR results are very close to AVBP and CANTERA results which shows its capability to simulate more complex chemistry. The pressure profile of the JAGUAR solution has very small pressure oscillations of the order of 0.1 Pa like in the contact discontinuity case of paragraph 7.1.2. It comes again from Eq. (7.12) that is not verified but this does not impact the flame structure in this case. Moreover, the flame speeds predicted by JAGUAR (38.06 cm.s⁻¹) and AVBP (38.09 cm.s⁻¹) are almost identical meaning that JAGUAR gives comparable results to a well established combustion solver.

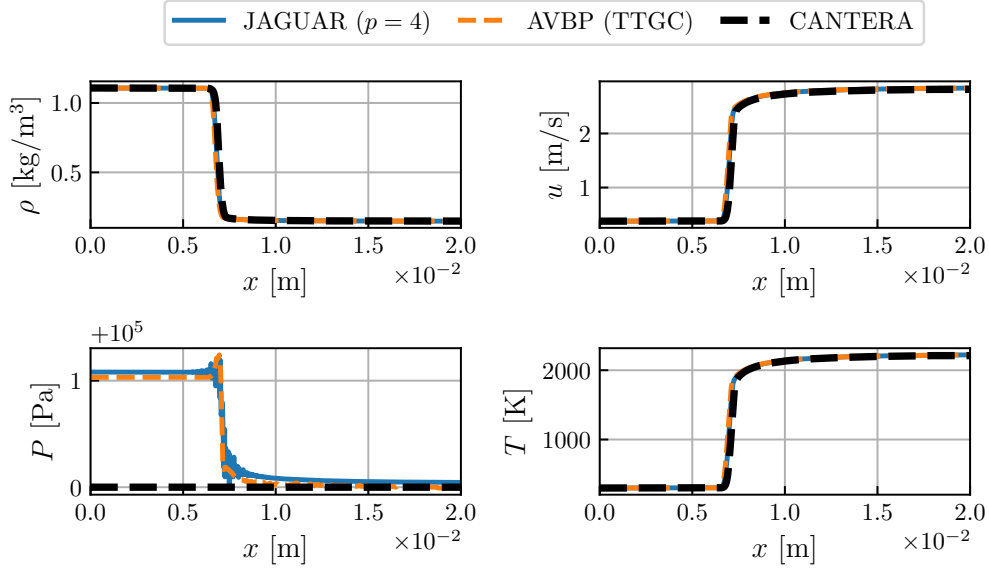


Figure 7.7. – Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$.

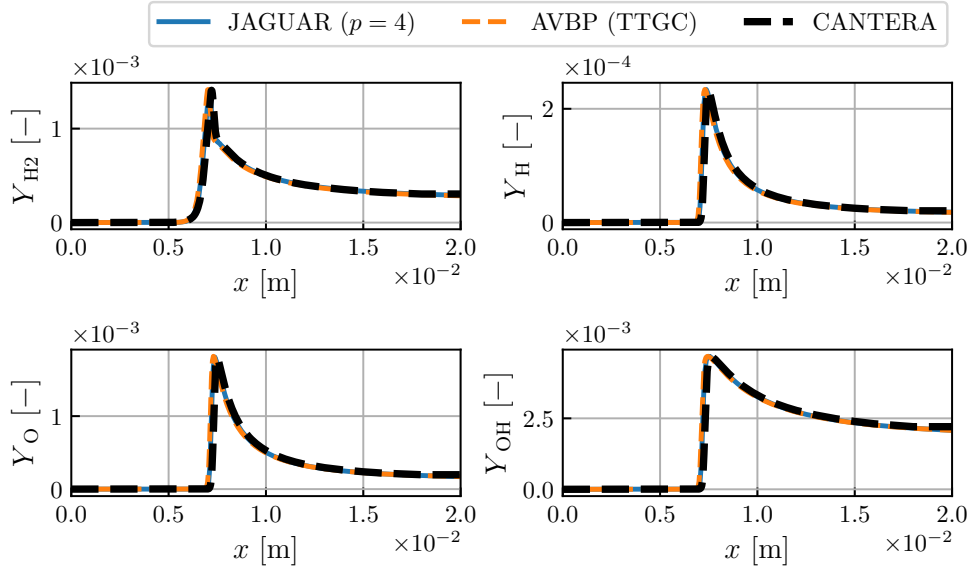


Figure 7.8. – Comparison of species mass fractions profiles of H_2 , H , O and OH between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$.

7.2.3. One-dimensional hydrogen/air flame using a skeletal mechanism

7.2.3.1. Presentation of the case and numerical setup

H_2/Air flames have recently gained more attraction in the low-carbon energy context. They differ significantly from hydrocarbon flames, due to the light and very reactive hydrogen molecule, and it

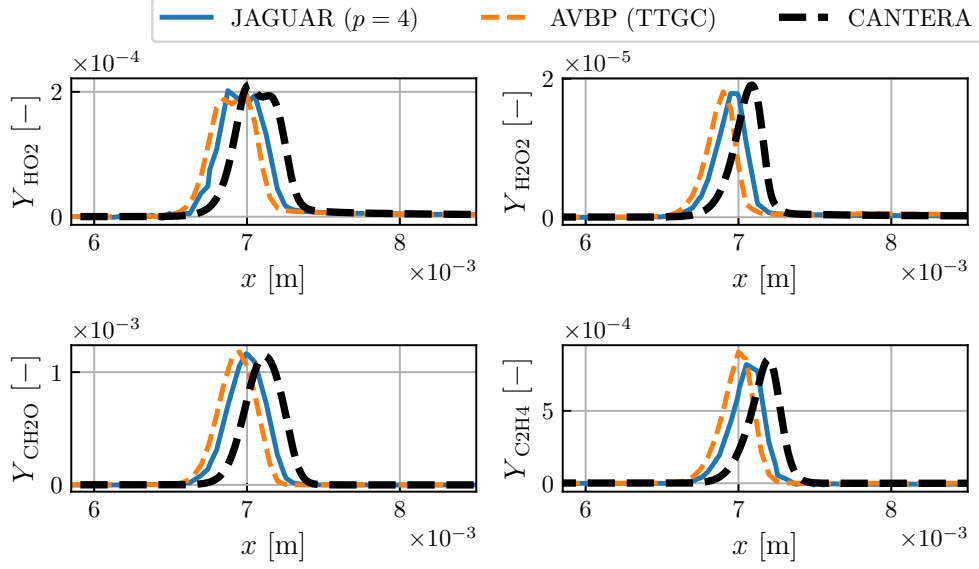


Figure 7.9. – Comparison of species mass fractions profiles of HO_2 , H_2O_2 , CH_2O and C_2H_4 between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$. Profiles are shown between $x_{\min} = 5.85$ mm and $x_{\max} = 8.5$ mm for clarity.

is interesting to evaluate JAGUAR also on such flames. The chemical scheme considered here is the so-called San Diego skeletal mechanism [136], named here as H2AIR_9_21_0_SD. It is composed of 9 transported species reacting through 21 chemical reactions. The characteristics of this flame are given in Table 7.5.

| ϕ [-] | T^f [K] | T^b [K] | P [Pa] | δ_L^0 [m] | S_L^0 [m.s ⁻¹] |
|------------|-----------|-----------|----------|-----------------------|------------------------------|
| 1.0 | 300 | 2361 | 101325 | 3.11×10^{-4} | 2.24 |

Table 7.5. – Characteristics of the 1D hydrogen-air premixed flame using a skeletal mechanism. T^b , S_L^0 and δ_L^0 are the values given by the CANTERA solution.

As it can be seen, H_2/Air flames have a much higher flame speed than hydrocarbon/air flames: S_L^0 is almost ten times higher than for CH_4/air flames shown in paragraphs 7.2.1 and 7.2.2. The flame thickness δ_L^0 is also much reduced for H_2/Air flames, around 30% smaller than the thickness of the CH_4/Air flames of paragraphs 7.2.1 and 7.2.2. The computational domain is here of length $L_x = 0.01$ m and is discretized with $N_e = 65$ elements and $p = 4$ ending up with 325 DOF. Based on the value of δ_L^0 in Table 7.5, this leads to approximately ten points inside the flame front.

7.2.3.2. Results

Final profiles of density, velocity, pressure and temperature are represented in Figure 7.10 and all species mass fractions profiles, except for N_2 , are represented in Figures. 7.11 and 7.12. Again, JAGUAR results are similar to the ones of AVBP and CANTERA showing that it is also able to simulate hydrogen combustion. Only small differences appear between JAGUAR and AVBP profiles of HO_2 and H_2O_2 : the shapes are the same but the AVBP profiles are slightly shifted to the left. Finally, flame speeds are almost the same for both codes: 2.136 m.s^{-1} for JAGUAR and 2.144 m.s^{-1} for AVBP.

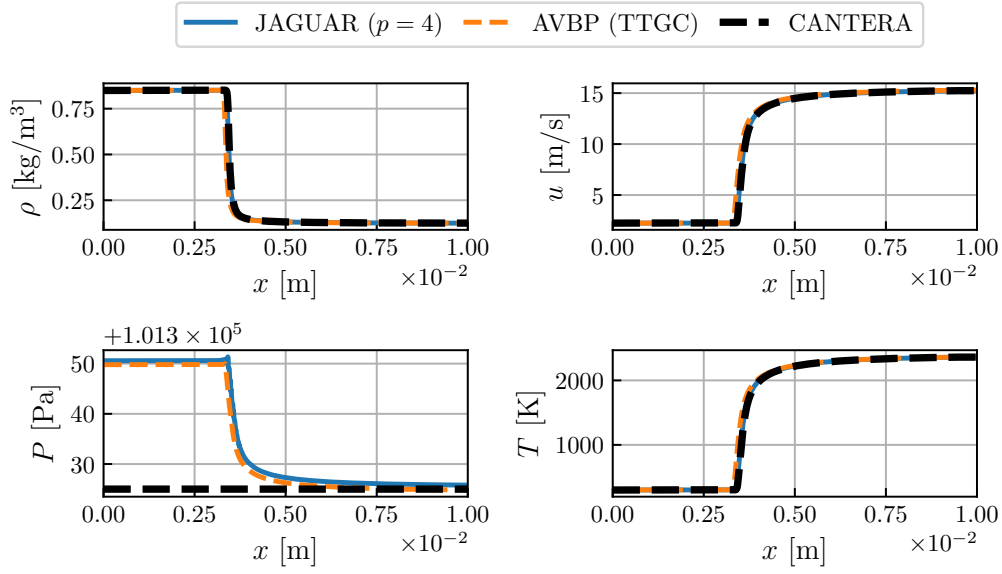


Figure 7.10. – Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$.

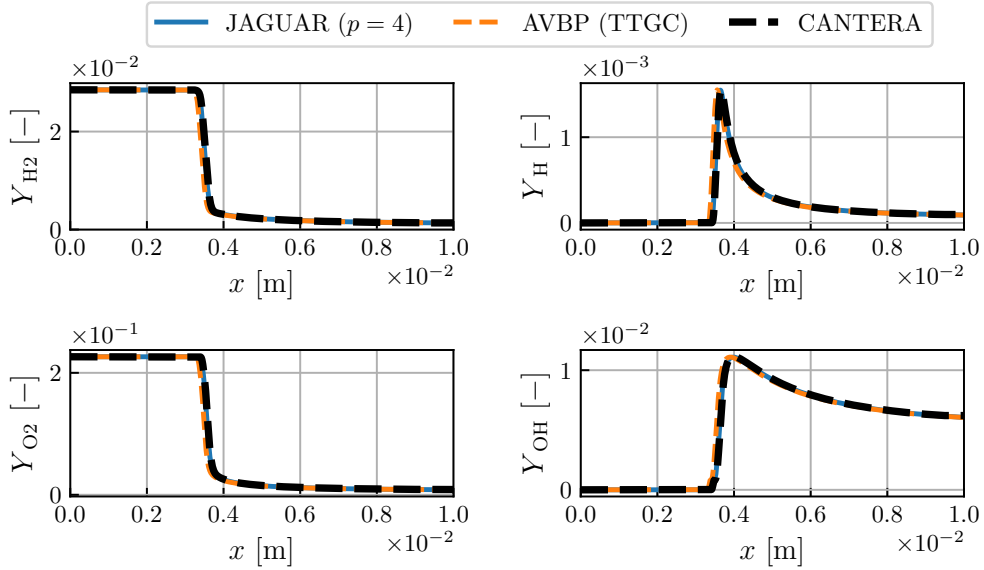


Figure 7.11. – Comparison of species mass fractions profiles of H_2 , H , O_2 and OH between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$.

7.2.4. Conclusion on one-dimensional laminar premixed flames results

The results obtained for all 1D laminar premixed flames presented above validate the implementation of multi-species transport and combustion thermochemistry in JAGUAR, including the combustion source terms computation, the resolution of species transport equations, the implementation of JANAF

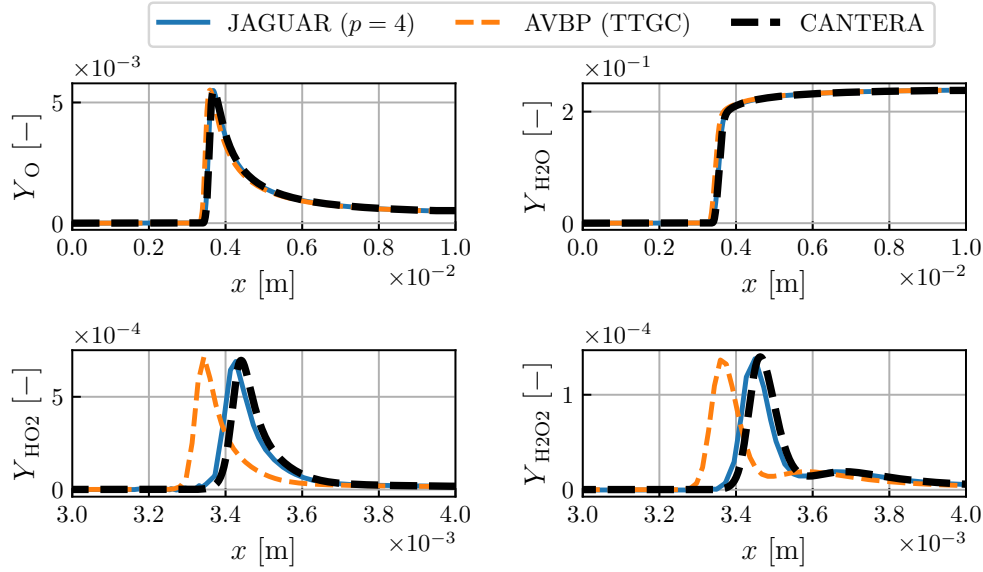


Figure 7.12. – Comparison of species mass fractions profiles of O, H₂O, HO₂ and H₂O₂ between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$. Profiles of HO₂ and H₂O₂ are shown between $x_{min} = 3$ mm and $x_{max} = 4$ mm for clarity reasons.

thermochemical tables and the inlet and outlet NSCBC in 1D multi-species case. Moreover, it also shows that the TUPY approach for interpolating variables from SP to FP, seems sufficient to stabilize the SD method in combustion simulations.

7.3. Two-dimensional laminar premixed flames

To go a step further in the validation, 2D laminar premixed flames have been simulated using JAGUAR. Two configurations are studied: a 2D burner-attached steady flame and a 2D circular propagating flame. The objective of this section is to see if JAGUAR is able to reproduce more realistic flame configurations with other boundary conditions than NSCBC, such as walls and symmetries. In all cases JAGUAR results will be again compared with results obtained with the AVBP solver using the TTGC scheme for convective fluxes and a finite element scheme of order 2 for diffusive fluxes. As for the 1D premixed flames, the JAGUAR simulations use the average approach and the SSP-ERK(3,3) scheme for respectively the interface diffusion fluxes and the time integration.

7.3.1. Two-dimensional methane/air burner computed with a two-reactions chemistry

7.3.1.1. Presentation of the case and numerical setup

The geometry of this test case is described in Figure 7.13. Fresh gases enter the burner axially (x -axis) through a NSCBC subsonic inflow imposing a parabolic profile given by Eq. (7.14):

$$u(y) = U_{cl} \left(1 - \frac{y^2}{h^2} \right) \quad (7.14)$$

where $U_{cl} = 4 \text{ m.s}^{-1}$ and $h = 0.65 \text{ mm}$. The two-steps mechanism is again the CH₄/Air-2S-BFER with equivalence ratio and fresh gas temperature also set respectively to 0.8 and 300 K at the inlet

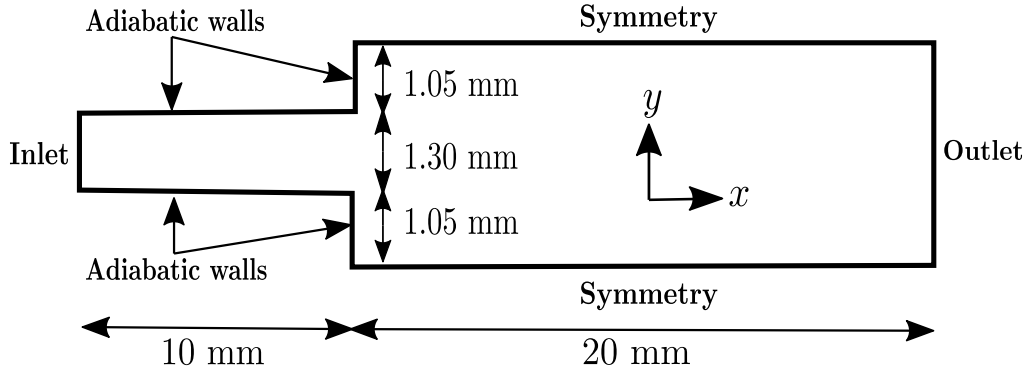


Figure 7.13. – Computational domain and boundary conditions for the 2D burner case.

so that flame characteristics are the ones in Table 7.1. At the outlet, a subsonic outflow imposing $P = 101325$ Pa is employed. From $x = 0$ to $x = 10$ mm (injection tube) side walls are adiabatic and from $x = 10$ mm to $x = 30$ mm (combustion chamber) symmetry boundary conditions are applied. The way symmetry and adiabatic no-slip walls are set using the SD method is explained in Section 5.2. For the JAGUAR simulation, the domain is discretized using 1216 uniform quadrilateral elements of characteristic size $\Delta_e = 2.27 \times 10^{-4}$ m. The polynomial degree within each element is set to $p = 4$ ending up with a total of 30400 DOF and eight points inside the flame front. The AVBP computational domain is discretized with almost the same number of DOF (31574 nodes in that case) for a fair comparison.

7.3.1.2. Results

JAGUAR and AVBP calculations are initialized using the same procedure which consists in setting hot burnt gases in the region between $x = 10$ mm and $x = 30$ mm whereas cold fresh gases are kept for $x < 10$ mm. Figure 7.14 shows the 2D heat release rate field obtained with JAGUAR, showing an attached flame of conical shape. The theoretical flame length can be calculated using Eq. (7.15) [290]:

$$L_f^{th} = \frac{h}{\tan(\alpha)} \text{ with } \alpha = \sin^{-1} \left(\frac{S_L^0}{U_b} \right) \text{ and } U_b = \frac{1}{2h} \int_{-h}^h u(y) dy = \frac{2}{3} U_{cl} \quad (7.15)$$

In Eq. (7.15), $U_b = 2.667$ m.s⁻¹, $\alpha = 0.106$ (computed with $S_L^0 = 0.2815$ m.s⁻¹) so that $L_f^{th} = 6.11$ mm which is very close to the value of 6 mm obtained with JAGUAR and measured on Figure 7.14. Vertical profiles of both JAGUAR and AVBP solutions at $x = 10.1$ mm (close to injector outlet) and at $x = 12$ mm in Figures 7.15 and 7.16 are merely identical. Horizontal profiles along the centerline $y = 0$ are also represented in Figure 7.17 for completeness showing also excellent agreement between the two solvers. Finally, Table 7.6 shows that both codes predict the same burnt gas temperature T^b and almost the same maximum heat release rate. CANTERA results obtained for the equivalent 1D flame are also added as reference values.

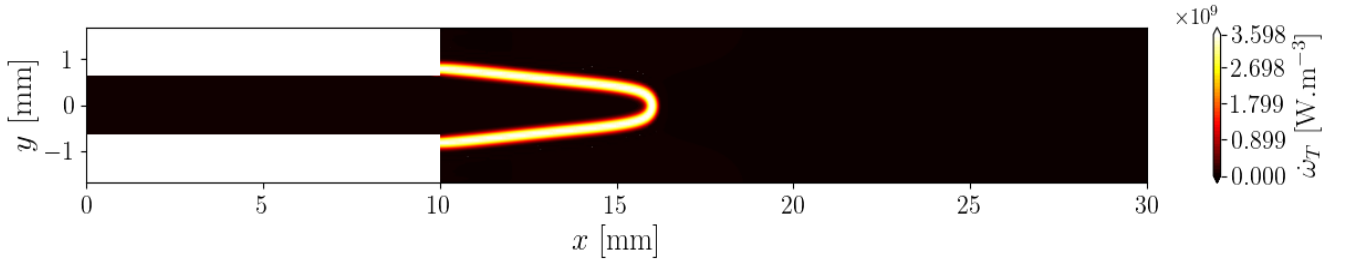


Figure 7.14. – 2D heat release rate field obtained with JAGUAR for the 2D burner case.

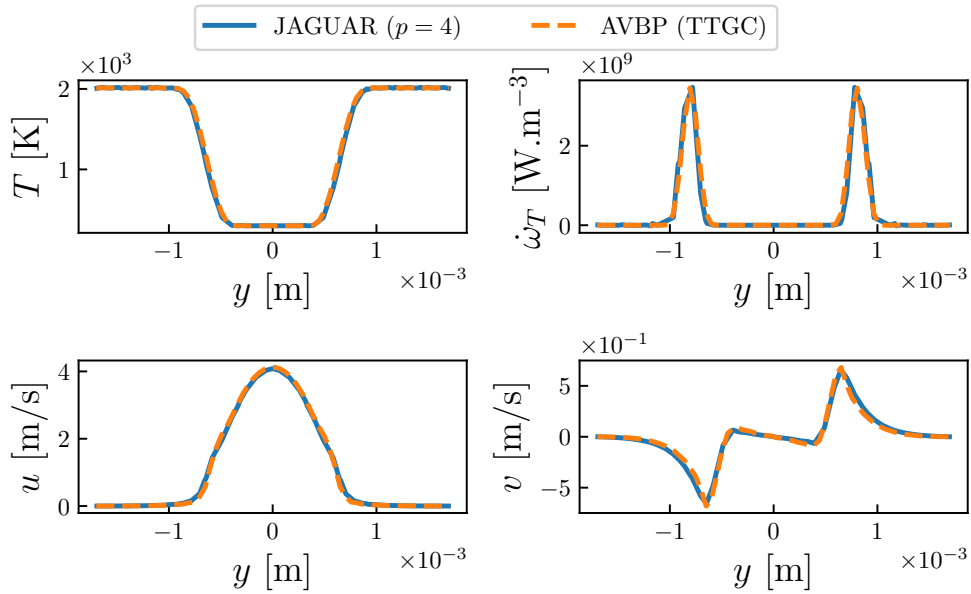


Figure 7.15. – Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $x = 10.1$ mm along y -axis for the 2D burner case.

| Code | T^b [K] | $\dot{\omega}_T^{max}$ [W.m ⁻³] |
|---------|-----------|---|
| JAGUAR | 2012 | 3.69×10^9 |
| AVBP | 2012 | 3.60×10^9 |
| CANTERA | 2010 | 3.71×10^9 |

Table 7.6. – Comparison between JAGUAR and AVBP of burnt gas temperatures T^b and maximum heat release rates $\dot{\omega}_T^{max}$ obtained in the 2D burner case using the CH₄/Air-2S-BFER scheme at $\phi = 0.8$. CANTERA values obtained on a 1D flame in the same conditions are also shown for reference.

7.3.2. Two-dimensional methane/air circular flame computed with a two-reactions chemistry

In this paragraph, an unsteady propagating flame is considered. This is a more challenging test case, allowing to evaluate the performance of JAGUAR in transient situations.

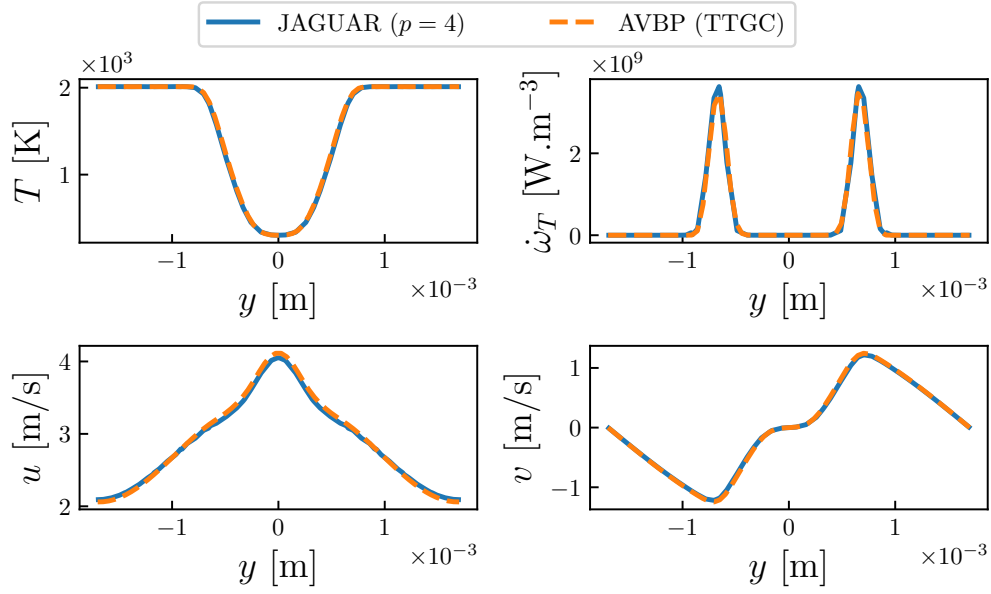


Figure 7.16. – Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $x = 12$ mm along y -axis for the 2D burner case.

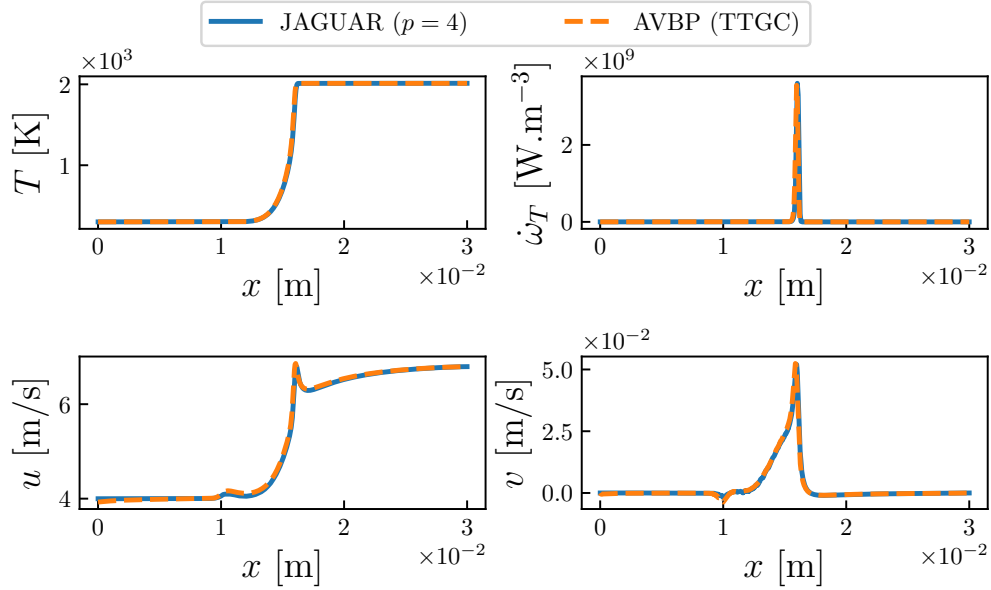


Figure 7.17. – Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $y = 0$ mm along x -axis for the 2D burner case.

7.3.2.1. Presentation of the case and numerical setup

The considered case is a 2D circular flame, as introduced in Bonhomme *et al.* in the context of flame speed measurements [291]. The geometry and the boundary conditions are represented in Figure 7.18. The computational domain is a square of size $L_x = L_y = 10$ cm with NSCBC outlet on all sides maintaining $P = 101325$ Pa. A disk of burnt gases is initialized at the domain center with a radius $r_0 = L_x/100 = 1$ mm as in Bonhomme *et al.* [291]. Fresh gases are surrounding this disk for

$r > r_0$. More precisely, at $t = 0$, temperature and species mass fractions in the whole domain are set following Eq. (7.16).

$$T(r, t = 0) = T^f + (T^b - T^f) \exp\left(\frac{-r^2}{r_0^2}\right), Y_k(r, t = 0) = Y_k^f + (Y_k^b - Y_k^f) \exp\left(\frac{-r^2}{r_0^2}\right) \text{ for } k = 1, N_s \quad (7.16)$$

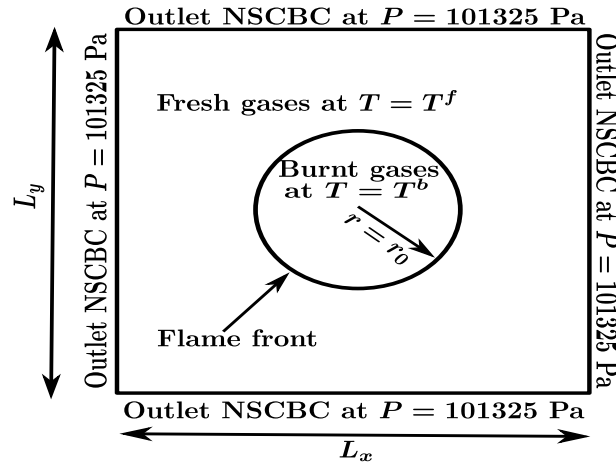


Figure 7.18. – Computational domain and boundary conditions for the 2D circular flame case.

With such initialization, a propagating flame forms at the disk contour and the disk of burnt gases grows in time. According to previous studies [291, 292] on the same configuration:

- once r becomes higher than 5.5 mm, the flame has grown enough and is not anymore influenced by the initial condition. This corresponds approximately to $t = 2$ ms here.
- The flame remains perfectly spherical without any impact of the boundary conditions until $r < 26.5$ mm.

That is why, the physical time simulated for all simulations is set to 6 ms since at that time $r \approx 12.5$ mm. Again, the two-steps mechanism is the CH₄/Air-2S-BFER with $\phi = 0.8$ and $T^f = 300$ K then flame characteristics are the ones shown in Table 7.1. JAGUAR simulations have been carried out using a mesh composed of $N_e = 16000$ uniform quadrilateral elements (400 1D elements in each direction) with $p = 4$. It ends up with 4×10^6 DOF. The AVBP grid is also composed of 4×10^6 DOF distributed in a uniform Cartesian mesh with 2000 nodes along each direction. Thus, based on the value of δ_L^0 in Table 7.1, JAGUAR and AVBP discretizations have eight points inside the flame front. This number of points was shown sufficient in paragraphs 7.2.1 and 7.3.1 to well resolve the flame front for both codes.

7.3.2.2. Results

Figure 7.19 shows the temperature fields and a longitudinal cut of temperature at $y = 0$ along x -axis, for both JAGUAR and AVBP simulations, at $t = 2$ ms, $t = 4$ ms and $t = 6$ ms. The fields were plotted for $(x, y) \in [-2 \text{ cm}, 2 \text{ cm}]^2$ for clarity since the complete domain is for $(x, y) \in [-5 \text{ cm}, 5 \text{ cm}]^2$. The solutions are almost identical for both codes as highlighted on the longitudinal cut of temperature at $y = 0$ along x -axis. The cylindrical shape of the flame is kept in time by both solvers.

An interesting quantity to look at is the evolution of the consumption speed with time. It is represented in Figure 7.20 for both JAGUAR and AVBP calculations. Both solvers actually predict the same variation of S_c with time until $t < 3$ ms. After this time AVBP starts to burn slightly faster than JAGUAR but the difference is very marginal.

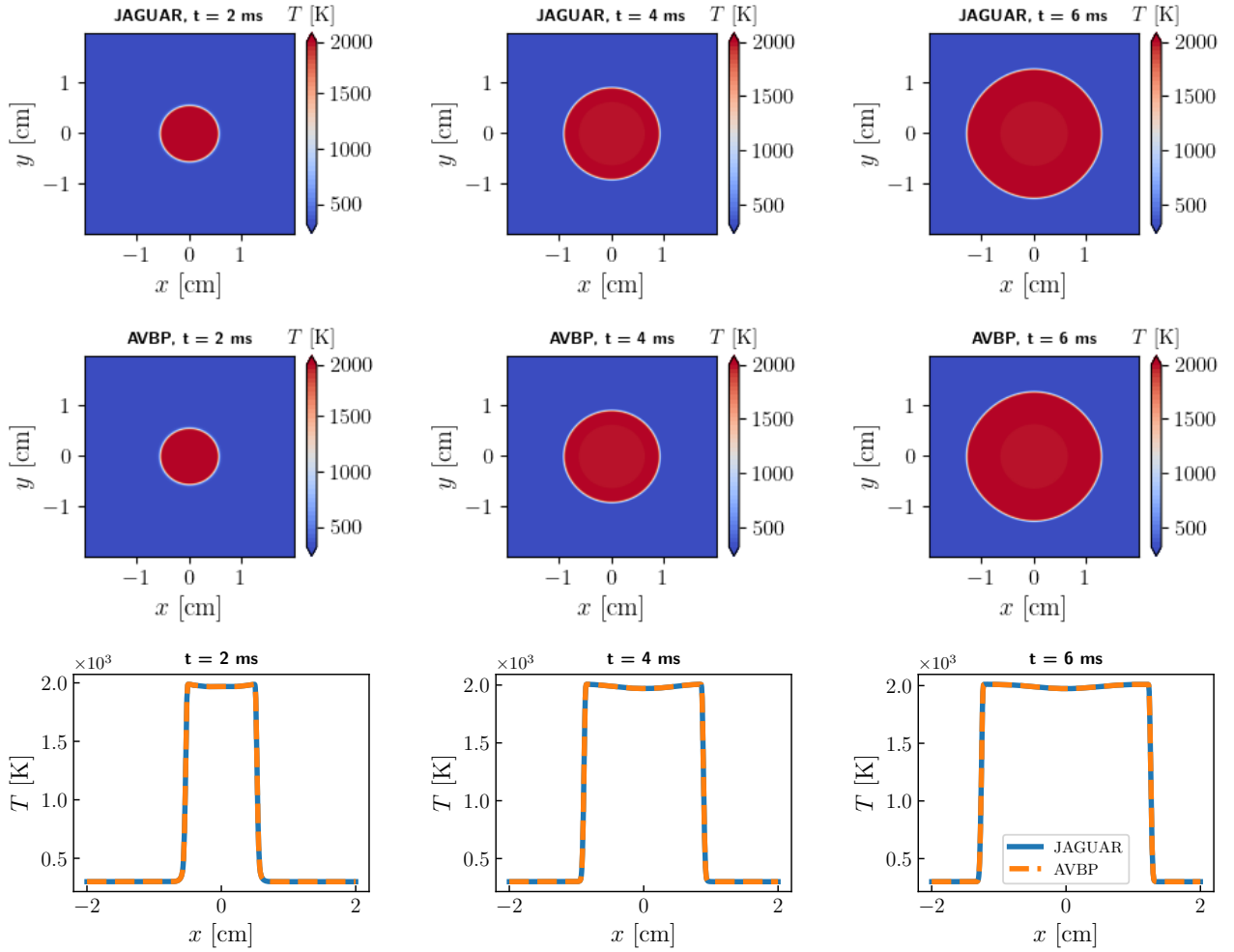


Figure 7.19. – 2D temperature fields obtained with JAGUAR (top) and AVBP (middle), and comparison of a longitudinal cut of temperature at $y = 0$ along x -axis, at $t = 2$ ms, $t = 4$ ms and $t = 6$ ms.

7.3.3. Conclusion on two-dimensional laminar premixed flames results

In both 2D steady and unsteady laminar premixed flames cases, using the CH₄/Air-2S-BFER chemical scheme [130], JAGUAR gives the same results as AVBP similarly to the 1D premixed flames shown in Section 7.2. This validation allows to further investigate the use of SD for combustion, and in particular the impact of the hp -refinement on both the solution accuracy and the computational cost. This is done in the next section.

7.4. Evaluation of p -refinement on combustion cases

In Sections 7.2 and 7.3, the possibility to simulate 1D and 2D laminar premixed flames with JAGUAR has been demonstrated. Before going towards more complex cases, the potential of the SD method in the field of combustion is studied in this Section. In particular the impact of increasing the polynomial degree on the quality of results and the associated computational cost is evaluated, as well as local polynomial adaptation which is particularly attractive for combustion. Results of this Section have been accepted for presentation at the 39th international Symposium of combustion in Vancouver in July 2022, with the title Efficiency of the high-order Spectral Difference method in

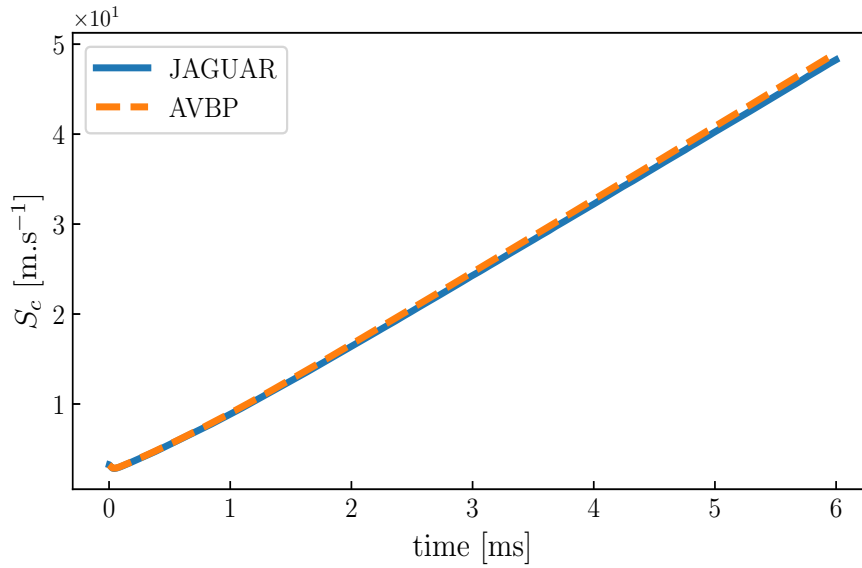


Figure 7.20. – Evolution with time of the consumption speed for JAGUAR and AVBP simulations.

combustion.

7.4.1. Context of the study

As already mentioned in paragraphs 1.3.2 and 1.3.3, only few combustion applications have been studied using HO discontinuous methods such as DG, SD or FR although there are currently well established for aerodynamics [96]. Almost all previous publications focused on the development of stable algorithm, either for DG, SD or FR, for simulating a multi-species reacting flow that handles pressure oscillations as introduced in paragraphs 1.3.1 and in Section 7.1. However, the gain of applying HO methods to combustion in comparison to existing methods has not been evaluated. According to previous studies on aerodynamic applications [112, 113], and compared to other HO methods, the SD method makes a good compromise between computational cost and accuracy. This conclusion may not hold for reacting flows as the number of equations to be solved is increased and additional requirements in terms of accuracy have to be considered. In addition to their high accuracy, HO discontinuous methods allow the local increase of the polynomial degree in regions of interest as explained in Section 4.5 for the SD method. This feature is called p -adaptation (or p -refinement) and has been applied in aerodynamic simulations either with the DG [94, 95] or the SD [96] method. Its use in combustion remains to be demonstrated and compared with the AMR technique widely employed in combustion nowadays. To the authors' knowledge, only Johnson and Kercher [27] have published a paper where p -adaptation was applied to combustion simulations in the context of the DG methodology, and nothing was published on p -adaptation using the SD method on combustion applications.

7.4.2. Test case presentation and reference solution

The test case considered here is the 2D burner case simulated in Section 7.3.1, whose geometry is shown on Figure 7.13. The same boundary conditions, the same CH₄/Air-2S-BFER chemical scheme and the same numerical approach (average approach for interface diffusion fluxes and SSP-ERK(3,3) for time integration) are used. A simulation using a very fine mesh is performed to serve as a reference to measure accuracy, since no full analytical solution is available. The computational domain is discretized using 4576 uniform quadrilateral elements and a polynomial degree of $p = 6$

(order 7). Consequently, the total number of DOF in the computational domain is 224224 as given by Eq. (4.74). The quadrilateral elements have all the same characteristic size $\Delta_e = 1.32 \times 10^{-4}$ m which gives a characteristic distance between the DOF of

$$\Delta_{\text{DOF}} = \Delta_e / (p + 1) = 1.88 \times 10^{-5} \text{ m} \quad (7.17)$$

Thus, based on the value of δ_L^0 in Table 7.1, the number of computational points in the flame front is $n_{pts}^{ff} = 23$ which indeed corresponds to a well resolved flame. Two simulations are run with JAGUAR and AVBP (with the TTGC scheme) respectively, using almost the same number of DOF and the same value for n_{pts}^{ff} . A converged steady solution is obtained after a physical time of $t_f = 15$ ms. Figure 7.21 shows the steady 2D field of heat release rate obtained with both solvers, of which Figure 7.22 plots a y -cut at $x = 12$ mm. The results show that both codes reproduce the same flame shape and structure when the space discretization is very fine. Therefore, either the fine AVBP or JAGUAR solution can be used as a reference. Since JAGUAR reference solution is of order 7 in space and AVBP reference solution is only of order 3 in space with the TTGC scheme, it was decided to keep JAGUAR solution as the reference. Using AVBP reference instead did not make any difference in the results.

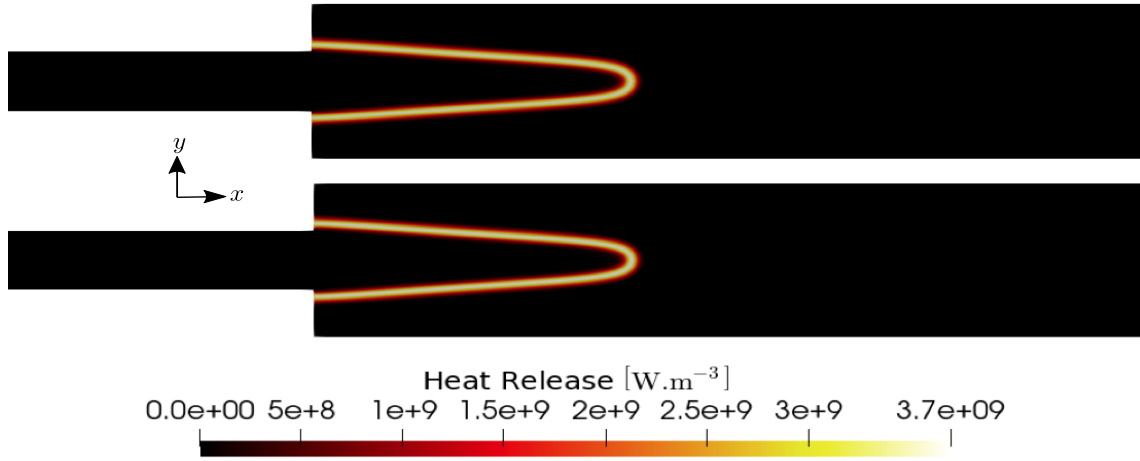


Figure 7.21. – Comparison of the steady 2D heat release rate field between JAGUAR (bottom) and AVBP (top) fine solutions for the 2D burner case.

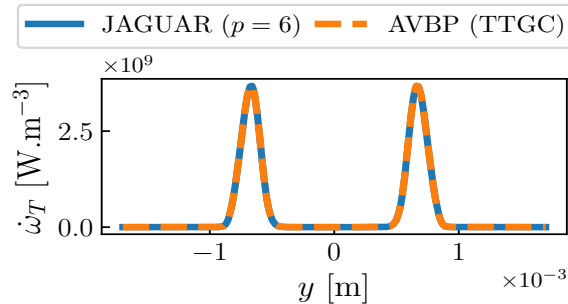


Figure 7.22. – Comparison of the steady heat release rate profiles at $x = 12$ mm between JAGUAR and AVBP fine solutions for the 2D burner case.

7.4.3. Influence of p on the accuracy of the solution in uniform p cases

Typical values of n_{pts}^{ff} in 3D LES combustion cases range between 5 and 10 depending on the stiffness of the chemistry. For a uniform quadrilateral mesh, Eq. (7.17) highlights the fact that for a

SD discretization multiple couples of values of Δ_e and p , or equivalently N_e and p , give almost the same Δ_{DOF} and, hence, the same n_{pts}^{ff} . In other words, a simulation with a small N_e but a large p has approximately the same number of DOF than a simulation with a large N_e but a small p , provided that they verify Eq. (4.74). Thus, the following question arises: for a given number of DOF (or a given n_{pts}^{ff} value), is it better, in terms of accuracy and performance to use large elements with high values of p or small elements with small values of p ? This question has already been answered for aerodynamic cases: low N_e combined with high p values lead to lower computational time and more accurate results [5]. For reacting cases, the question remains open and first answers are given by the analysis below.

As a first step, this question is addressed in terms of the accuracy of the solution. Simulations of the 2D burner case are run using JAGUAR until $t = t_f$ using polynomial degrees from $p = 2$ to $p = 6$. Different values of N_e and p are used in order to keep approximately 30900 and 20100 DOF in the computational domain. Values of N_e , p , Δ_{DOF} and n_{pts}^{ff} are summed up in Tables 7.7 and 7.8 for each case.

One way to compare calculation results is to compute the L^2 -error of the 2D heat release rate field with respect to the reference solution. This error can be defined following the methodology proposed in recent HO workshops [5]:

$$\epsilon^{\dot{\omega}_T} = \left[\sum_{i=1}^{\text{DOF}} \left(\dot{\omega}_{T,i} - \dot{\omega}_{T,i,ref}^{interp} \right)^2 / \text{DOF} \right]^{1/2} \quad (7.18)$$

where $\dot{\omega}_{T,i}$ is the heat release rate value at SP i of the considered solution and $\dot{\omega}_{T,i,ref}^{interp}$ is the heat release rate value at SP i from the reference solution interpolated on the considered solution mesh. This interpolation is done using the inverse distance weighting method. Figure 7.23 shows the evolution of $\epsilon^{\dot{\omega}_T} / \dot{\omega}_{T,ref}^{max}$ with respect to p for the two numbers of DOF considered, where $\dot{\omega}_{T,ref}^{max} = 3.7 \times 10^9 \text{ W.m}^{-3}$ is the maximum of heat release rate in the reference solution. The values obtained with AVBP using the Lax-Wendroff (LW) FV scheme for convective fluxes [199] (second-order in space and time) and the TTGC scheme are also plotted in Figure 7.23.

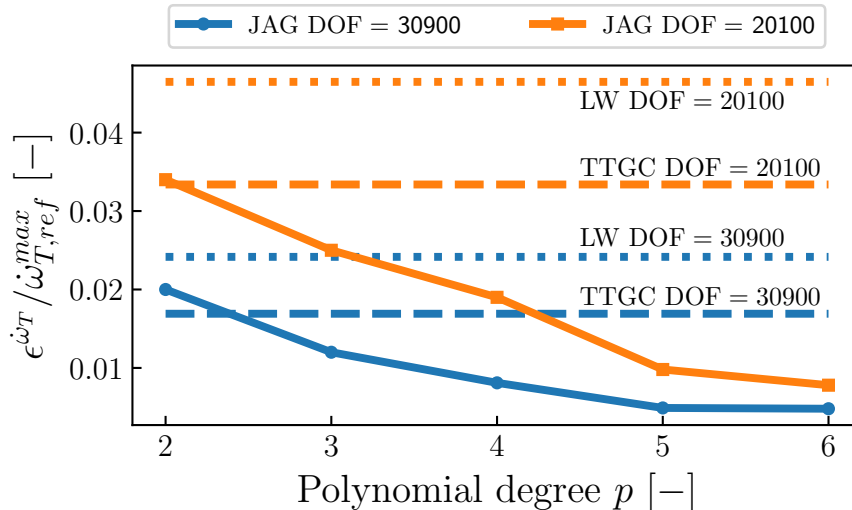


Figure 7.23. – Evolution of $\epsilon^{\dot{\omega}_T} / \dot{\omega}_{T,ref}^{max}$ with respect to p for two numbers of DOF. The values obtained with the LW and TTGC schemes at these numbers of DOF are also shown.

As expected, the error decreases when the number of DOF increases for both JAGUAR and AVBP simulations. For a given number of DOF, the error decreases with the order of the numerical scheme

also for both codes. For JAGUAR, the same error at high number of DOF and low p is recovered with the low number of DOF and high p . This means that for reacting flows as well, large elements with high p give more accurate results. This result is probably due to the polynomial basis that better describes the flow and flame quantities in the cell when p is high, than a piecewise linear function corresponding to low p cases. Note that the error with JAGUAR is always lower than the error with AVBP except at $p = 2$ where the results are comparable showing the gain of using HO methods.

7.4.4. Influence of p on the computational cost in uniform p cases

The computational cost with respect to the polynomial degree is now assessed. More precisely two quantities are of interest:

1. The iteration cost per DOF noted κ .
2. The real time taken to simulate 1 ms of physical time noted τ .

Tables 7.7 and 7.8 show the values of κ and τ for each JAGUAR and AVBP simulations whose errors were computed in paragraph 7.4.3. For both numbers of DOF, τ values of JAGUAR are almost constant when the polynomial degree increases. This indicates that even in reacting multi-species cases, the order of the SD method can increase with almost no change of the iteration cost. It comes from the competition between two main processes in the SD scheme: the interpolation process at FP and the interface treatment at FP (the Riemann solver for convective fluxes and the diffusion scheme for viscous fluxes). In a 2D quadrilateral element, there are $(p+2)(p+1)$ FP per direction so that there are $O(p^2)$ FP inside one element. The interpolation process from SP to FP is done direction per direction which entails that $O(p)$ operations are needed for FP along ξ -direction and also $O(p)$ operations for FP along η -direction. Consequently within one quadrilateral element, the interpolation process at all FP scales as $O(p^3)$ and since there are $\text{DOF}/(p+1)^2$ elements thanks to Eq. (4.74), the overall interpolation process in the whole domain scales like $O(p)$ at constant DOF value. On the other hand, the interface treatment at FP, again if DOF is kept constant, scales as $O(1/p)$ since there are still $\text{DOF}/(p+1)^2$ elements and four edges per quadrilateral with $p+1$ interface FP on each. For low p values, the interface treatment is dominating because there are more interface FP than for high p values if DOF is kept constant and also more MPI communications since there are more interface FP shared by two processors. However, when p increases the interface treatment is less costly and compensates the $O(p)$ increase. At constant DOF, the number of interface FP, noted n_{FP}^I , scales as $O(p)$ and the total number of FP, noted n_{FP}^{tot} , scales as $O(p^2)$ so that $n_{FP}^I/n_{FP}^{\text{tot}}$ varies as $O(1/p)$. This explains the almost 10% decrease of κ between $p = 2$ and $p = 3$ in both Tables 7.7 and 7.8 where $n_{FP}^I/n_{FP}^{\text{tot}} \sim 0.5$ for $p = 2$ whereas $n_{FP}^I/n_{FP}^{\text{tot}} \sim 0.33$ for $p = 3$. For $p = 4$ $n_{FP}^I/n_{FP}^{\text{tot}} \sim 0.25$ so there is still a noticeable improvement on κ compared to $p = 3$. For $p = 5$ and $p = 6$ this gain from the interface treatment becomes less important compared to the $O(p)$ increase of the extrapolation process. Actually, the $O(p)$ increase will not be seen as long as $p < 8$. For comparison, increasing p in the DG method entails the use of more expensive quadrature rules [30] to reduce aliasing errors which can make the calculation unstable for highly turbulent flows. Thus, an increase of p with the DG method often results in an increase of κ even for $p < 8$. The same reasoning applies to the comparison with low-order (LO) methods for which increasing the order of the scheme often results in an increase of the iteration cost. The comparison between the LW and TTGC schemes (order 2 and 3 in space, respectively) of AVBP is a good proof of that: the TTGC scheme is 2.5 times more expensive than the LW scheme [179] as also observed in Tables 7.7 and 7.8. Nevertheless, τ_{TTGC} is still lower than all τ obtained with JAGUAR. This means that the SD method seems more expensive than LO methods. However, it must be noted that JAGUAR is a new code where coding has not been optimized as much as in AVBP.

The iteration cost per DOF is not sufficient to evaluate the computational cost of the simulations as the timesteps used for each simulation may differ. Actually, all HO methods suffer from CFL restrictions that are more constraining than for LO methods. For instance, in the SD method, the

| JAGUAR | | | | | |
|---------------|------|-----------------|------|-----|-----|
| p | 2 | 3 | 4 | 5 | 6 |
| N_e | 3437 | 1930 | 1236 | 860 | 632 |
| κ | 8.5 | 7.7 | 7.4 | 7.3 | 7.1 |
| τ | 361 | 371 | 407 | 471 | 558 |
| AVBP | | | | | |
| κ_{LW} | 2.8 | κ_{TTGC} | 6.3 | | |
| τ_{LW} | 106 | τ_{TTGC} | 239 | | |

Table 7.7. – Values of κ [$\mu s/ite/DOF$] and τ [s] for both AVBP and JAGUAR simulations with different values of N_e and p to keep the number of DOF around 30900 ($\Delta_{DOF} = 5.09 \times 10^{-5}$ m and $n_{pts}^{ff} = 8$).

| JAGUAR | | | | | |
|---------------|------|-----------------|-----|-----|-----|
| p | 2 | 3 | 4 | 5 | 6 |
| N_e | 2230 | 1256 | 804 | 558 | 410 |
| κ | 8.2 | 7.5 | 7.4 | 7.2 | 7.2 |
| τ | 182 | 190 | 216 | 247 | 292 |
| AVBP | | | | | |
| κ_{LW} | 2.8 | κ_{TTGC} | 6.4 | | |
| τ_{LW} | 57 | τ_{TTGC} | 128 | | |

Table 7.8. – Values of κ [$\mu s/ite/DOF$] and τ [s] for both AVBP and JAGUAR simulations with different values of N_e and p to keep the number of DOF around 20100 ($\Delta_{DOF} = 6.31 \times 10^{-5}$ m and $n_{pts}^{ff} = 6$).

CFL limit scales as $(p+1)^{-1}$ [293] and becomes more restrictive as p increases. In this work, for $p = \{2, 3, 4, 5, 6\}$, CFL values were set to 0.36, 0.32, 0.28, 0.24 and 0.20. These CFL values were chosen from the experience in using the SSP-ERK(3,3) scheme of Gottlieb and Shu [222] with JAGUAR and are close to the stability limit for each p . Finding the exact CFL limit is out of the scope of this study and will not change the analysis conducted here. Consequently, this CFL condition has an impact on computational time as can be observed in Tables 7.7 and 7.8 where τ increases with p . For instance, $\tau(p=6)/\tau(p=2) = 1.55$, meaning that the $p=6$ case takes 55% more of real time to simulate 1 ms than the $p=2$ case. Because the TTGC scheme (and also the LW one) is classically used with a CFL of 0.7: $\tau(p=6)/\tau_{TTGC} = 2.3$ meaning that the $p=6$ case takes 2.3 times more than the TTGC scheme to reach the same final simulated time. For $p=4$ (order 5), the ratio $\tau(p=4)/\tau_{TTGC}$ is 1.7. These differences in computational time between the SD method and LO methods are high but probably lower than the differences between the DG method and LO methods since the CFL limit for DG varies as $(2p+1)^{-1}$ [20]. Note that only the CFL limit was considered since the convective operator appears to be the limiting term compared to diffusive and chemical operators. For instance, with CFL = 0.32 for the $p=3$ case at 30900 DOF, the convective timestep is $\Delta t_{conv} = 1.7 \times 10^{-8}$ s (see Eq. (4.137)). This value is well smaller than both the diffusive timestep $\Delta t_{diff} = 4.1 \times 10^{-7}$ s (see Eq. (4.139)) obtained with a Fourier number of 0.1 and the chemical timestep $\Delta t_{chem} = 1 \times 10^{-4}$ s (see Eq. (4.141)) which is actually the chemical timescale of the species CO.

Combined with the accuracy analysis conducted in paragraph 7.4.3, for the same level of error, it is faster to use less DOF with high p rather than more DOF with low p . For instance, Table 7.9 sums up the turnaround times for three different discretizations that have the same level of error ($\epsilon^{\dot{\omega}_T}/\dot{\omega}_{T,ref}^{max} = 0.02$) according to Figure 7.23. It shows that $p=2$ in the 30900 DOF case and $p=4$ in the 20100 case have the same error but the first one has a turnaround time for the results which is 67% higher than the second one. Moreover, $p=4$ in the 20100 case is slightly faster than the TTGC scheme in the 30900 case which also has the same error. Therefore, for $\epsilon^{\dot{\omega}_T}/\dot{\omega}_{T,ref}^{max} = 0.02$,

the optimum choice in terms of accuracy and computational cost is $p = 4$ in the 20100 case. As a

| | $(p = 2)$ and 30900 DOF | $(p = 4)$ and 20100 DOF | AVBP (TTGC) 30900 DOF |
|------------|-------------------------|-------------------------|-----------------------|
| τ [s] | 361 | 216 | 239 |

Table 7.9. – Turnaround times for $\epsilon^{\dot{\omega}_T} / \dot{\omega}_{T,ref}^{max} = 0.02$ with different discretization.

summary, JAGUAR has a κ which is a little bit higher than the one of the TTGC scheme of AVBP but κ does not change when p increases. However, CFL restrictions for HO methods still make their computational cost high compared to LO methods. This computational cost issue can be strongly reduced if local p -adaptation is considered as it is highlighted in paragraph 7.4.5.

7.4.5. Local p -adaptation

In paragraphs 7.4.3 and 7.4.4, the SD method was shown to be more accurate with higher polynomial degree without altering the iteration cost per DOF. However, the overall computational cost is still high compared to other methods because of the CFL constraints. Increasing the CFL limit is difficult but reducing the number of DOF of the computational domain is possible without altering the accuracy of the results by using local p -adaptation presented in Section 4.5. This method consists in placing the DOF in regions of interest only by adapting the polynomial degree in the mesh elements. Therefore, computations are run with a non-uniform polynomial degree that changes according to user-defined criteria.

To illustrate this methodology, the 2D burner case is considered with the mesh composed of $N_e = 1236$ quadrilateral elements and $p = 4$ inside each of them. The objective is now to simulate the same case with $p = p_{max} = 4$ only in near-flame elements and $p = p_{min} = 2$ elsewhere. The polynomial degree p_e within each element is set according Eq. (4.126) where the sensor value $\tilde{\theta}_e \in [0, 1]$ is obtained thanks to Eq. (4.125). As explained in paragraph 4.5.3, a first simulation at uniform $p = 2$ in all elements is performed and used to compute the sensor. This simulation at $p = 2$ on a mesh initially designed for $p = 4$ does not give good results but is sufficient to get acceptable values of $\tilde{\theta}_e$. Figure 7.24 shows the resulting field of $\tanh(\alpha \tilde{\theta}_e)$ where the grid of elements is shown.

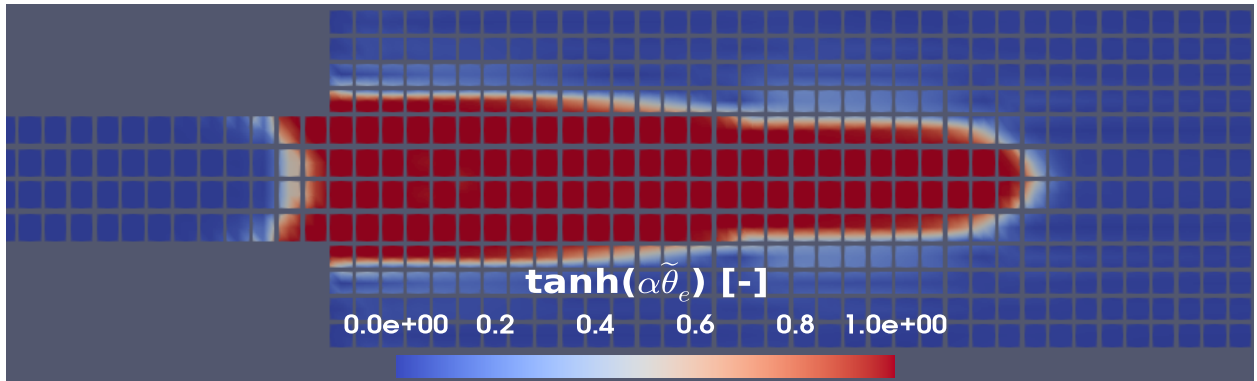


Figure 7.24. – Field of $\tanh(\alpha \tilde{\theta}_e)$ obtained from a uniform $p = 2$ simulation on the mesh with 1236 elements and $\alpha = 100$.

A second simulation is then run using the values of $\tanh(\alpha \tilde{\theta}_e)$ in Eq. (4.126) to set the polynomial distribution illustrated in Figure 7.25. As it can be observed, the flame region is mainly composed of elements with $p_e = 4$ surrounded by a zone of elements at $p_e = 3$ and the remaining elements where source terms and gradients are no more observable are at $p_e = 2$. This polynomial distribution does not well follow the 2D heat release rate field since the sensor is based on $\|\nabla \rho\|_2$ and not on $\dot{\omega}_T$ but it is sufficient to show the possibilities of doing p -adapted simulations in combustion.

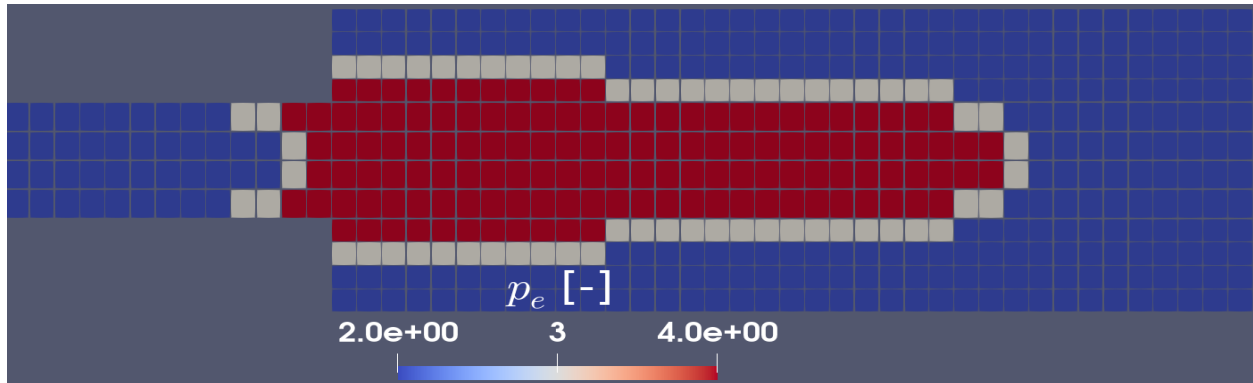


Figure 7.25. – Polynomial degree distribution for a calculation with $p_{min} = 2$ and $p_{max} = 4$ on the mesh with 1236 elements.

Figures 7.26 and 7.27 show respectively the steady 2D heat release rate field and its y -cut at $x = 12$ mm obtained with the p -adapted simulation. Comparison of the profile is done with uniform p simulations respectively at $p = 2$ and $p = 4$ on the same mesh. The flame structure is well retrieved for the p -adapted case. An improvement of the uniform $p = 2$ solution is clearly seen in Figure 7.27: the mesh is not adapted to $p = 2$ in the flame zone where $n_{pts}^{ff} = 4$ whereas locally switching to $p = 4$ in the flame zone gives $n_{pts}^{ff} = 8$ resulting in better results. In terms of accuracy, the normalized L^2 -error of the 2D heat release rate field for the p -adapted simulation is $\epsilon^{\dot{\omega}_T} / \dot{\omega}_{T,ref}^{max} = 8.7 \times 10^{-3}$ and is close to the error value of 8.1×10^{-3} obtained for the uniform $p = 4$ case in Section 7.4.3.

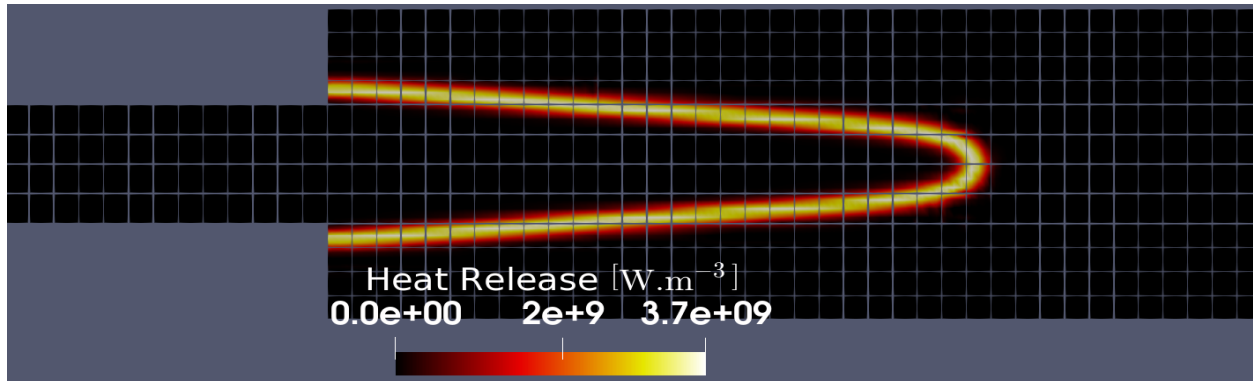


Figure 7.26. – Steady 2D heat release rate field obtained with a polynomial order distribution $2 \leq p \leq 4$ on the mesh with 1236 elements.

As for the computational cost, Table 7.10 shows the values of the number of DOF, κ and τ for the three calculations of Figure 7.27. The gain in the number of DOF in the varying $2 \leq p \leq 4$ case compared to the uniform $p = 4$ simulation is of about 56% for a similar accuracy of the result and the computational time is divided by 2. Even adding the cost of the computation at uniform $p = 2$ (needed for the sensor evaluation), there is still a gain of 33% of the computational time. Similar observations were done in aerodynamic computations either with the DG [94] or SD methods [96]. Of course in that case, JAGUAR was used with load-balancing based on the polynomial distribution per element as stated in paragraph 4.5.2.

These simulations show that the p -adaptation does not solve the issue of CFL restrictions because the varying $2 \leq p \leq 4$ case is run at a CFL of 0.28 which is the CFL used at constant $p = 4$ case. This explains the still important difference in τ between the uniform $p = 2$ case (run at a CFL of 0.36) and the varying $2 \leq p \leq 4$ case, although the DOF are closed. However, p -adaptation significantly

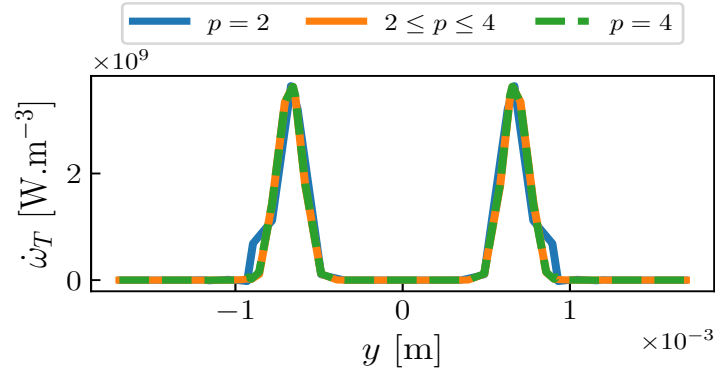


Figure 7.27. – Comparison of the steady heat release rate profiles at $x = 12$ mm, obtained on the same mesh with 1236 elements, between a computation at uniform $p = 2$, varying $2 \leq p \leq 4$ and uniform $p = 4$.

| | $p = 2$ | $2 \leq p \leq 4$ | $p = 4$ |
|----------|---------|-------------------|---------|
| DOF | 10944 | 13444 | 30900 |
| κ | 8.3 | 7.9 | 7.4 |
| τ | 96 | 199 | 407 |

Table 7.10. – Values of DOF, κ [$\mu s/ite/DOF$] and τ [s] for JAGUAR simulations of the 2D burner on the same mesh but with different polynomial distributions within elements.

reduces the number of DOF and, hence, the computational time. These first results demonstrate the capability of the SD method to perform combustion simulations using p -adaptation. Although the CFL constraints are not resolved, the SD method can easily perform local discretization refinement, *i.e.*, to significantly accelerate simulations without losing accuracy or modifying the mesh. It is therefore easy to make it adaptative and to optimize it during one simulation. Note that it was also chosen to not compare the p -adapted simulation in terms of τ with the AVBP case at 30900 DOF using the TTGC scheme since DOF values were not similar. It was not in the scope of this paragraph which was to show how the p -adaptation can really reduce the SD computational time. The comparison between p -adaptation and AMR coupled with a FV scheme is the next step to that and is left for future work.

7.4.6. Conclusions of the study

The present study investigates the potential of the SD method to simulate combustion problems in terms of accuracy of the results and computational cost. The results show that increasing the order of the numerical scheme improves the accuracy without increasing the iteration cost as opposed to LO methods and weak HO discontinuous methods such as the DG method. It is also highlighted that, as for aerodynamic cases, using large elements with high p values is better in terms of accuracy and performance for the same level of error than using small elements with low p values. However, like other HO methods, the SD method suffers from CFL limitations that become more and more restrictive as p increases. Although these limitations are lighter than in the DG method, it entails that the SD method with a uniform degree p in all mesh elements is still between 2 to 3 times more expensive than LO methods. This limitation is overcome with the use of p -adaptation that drastically reduces the number of DOF by setting high values of p only in regions of interest. Even if the time step is identical between the uniform p and the varying p cases, the computational cost is reduced by a factor 2 for the same accuracy of the results. Therefore, with p -adaptation the computational cost of

the SD method is found comparable to LO methods showing the possibility to increase the numerical order and improve the results at no extra computational cost.

To the authors' knowledge, it is the first time that p -adaptation is applied to a reacting flow test case using the SD method with very encouraging results. This work is a first and necessary step of the development of p -adaptation methods before applying it to more realistic configurations such as 3D turbulent flames. Many improvements can still be made, using for example a dynamic sensor which updates the local p distribution in real time during the computation, avoiding pre-calculations and allowing non-steady cases. Another sensor built on heat release rate or temperature gradient, which seem more adapted to combustion phenomena, should also be tested. In this work, only quadrilateral elements have been considered since the use of triangular elements with high p values in the SD framework is very recent [52, 116] and not straightforward to be applied in combustion and with p -adaptation. It is even worst when considering tetrahedral elements since for now in SD, stable schemes are available for $p = 2$ at most. Future studies using this kind of elements within the SD method will allow to fully conclude about the performance of the method. Compared to AMR, also called h -adaptation, p -adaptation allows to perform local refinement more easily than with LO methods as it avoids re-meshing. The optimum use of hp -refinement for combustion is the next challenge and is left for future work.

7.5. Comparison between original and SDLIFT formulations for the diffusion scheme on 1D and 2D premixed flames using coarse meshes

A central question when simulating combustion cases is how many grid points (n_{pts}^{ff}) are needed inside the flame front to well-resolve it. In paragraph 3.3.2.1, it was said that n_{pts}^{ff} values are commonly between 5 to 20 depending on the chemistry. However, this general observation depends on the numerical method and then each CFD code has its own good practices for choosing the appropriate resolution of the flame front. For instance, in AVBP for CH_4/Air chemistries (two-reactions or ARC), experience has shown that $n_{pts}^{ff} \approx 4 - 5$ or higher is sufficient to have good estimations of flame speed. The objective of this section is first to see if $n_{pts}^{ff} \approx 4 - 5$ is also sufficient when using the SD method and secondly if it is possible to go below this value thanks to the increased accuracy. The SSP-ERK(3,3) scheme of Gottlieb and Shu [222] is employed for time integration in all JAGUAR simulations of this section.

7.5.1. 1D methane/Air premixed flame with four points inside the flame front using the original formulation of the diffusion scheme

The same 1D flame introduced in paragraph 7.2.1, whose characteristics are given in Table 7.1, is considered but this time with 185 DOF, instead of 400. The number of points in δ_L^0 using this discretization is $n_{pts}^{ff} \approx 4$. Three different pairs of (N_e, p) values have been considered to keep $\text{DOF} \approx 185$ and are summed up in Table 7.11. Simulations are carried out with the original formulation of

| | | | |
|-------|----|----|----|
| p | 4 | 5 | 6 |
| N_e | 37 | 31 | 27 |

Table 7.11. – Pairs of (N_e, p) values to keep $\text{DOF} \approx 185$.

Sun *et al.* [47] described in paragraphs 4.1.6 and 4.4.1 and results are shown in Figure 7.28a for the HRR between 0.92 cm and 1 cm for clarity and for the CO mass fraction in Figure 7.28b. The AVBP solution obtained with the TTGC convective schemes on the same number of DOF (without employing artificial viscosity) is also added for comparison.

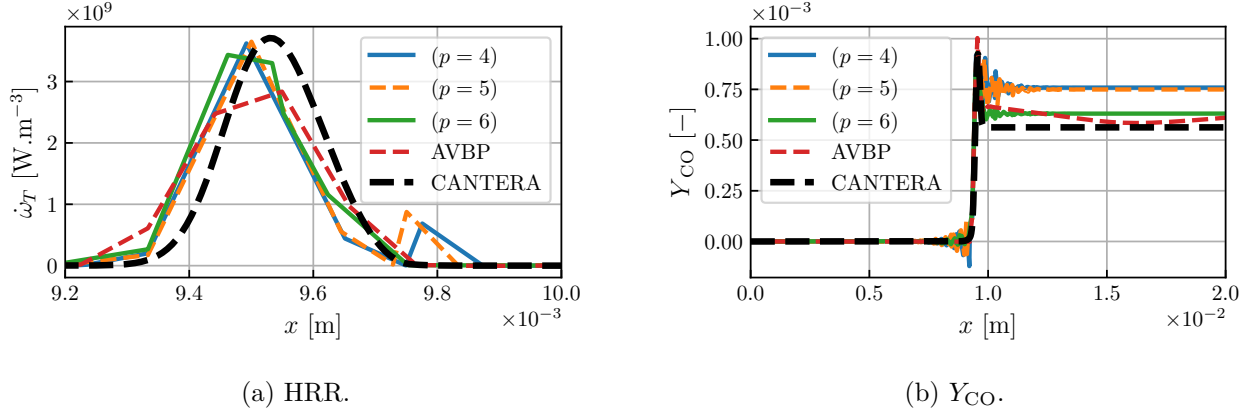


Figure 7.28. – Comparison of HRR and Y_{CO} between JAGUAR calculations at $p = 4$, $p = 5$ and $p = 6$ when $n_{pts}^{ff} \approx 4$ and the original diffusion scheme is employed and the AVBP simulation. CANTERA solution is added as reference.

JAGUAR solutions present strong oscillations, especially on Y_{CO} , for any value of p whereas the AVBP result is not oscillating although it underestimates the HRR. However, as highlighted in Section 7.4, increasing p at constant DOF improves the quality of the solution and in this case reduces the intensity of oscillations. Nevertheless, even at $p = 6$, Y_{CO} remains unstable and is still far from the CANTERA reference value in the burnt gases although it is similar in the AVBP case. The origin of such problem was hard to identify and the first idea was to test ρ UPY approach, mentioned at the end of paragraph 7.1.2, instead of TUPY approach. Figure 7.29 shows Y_{CO} profiles already plotted in Figure 7.28b but now compared with the ones obtained ρ UPY approach at FP. Except for

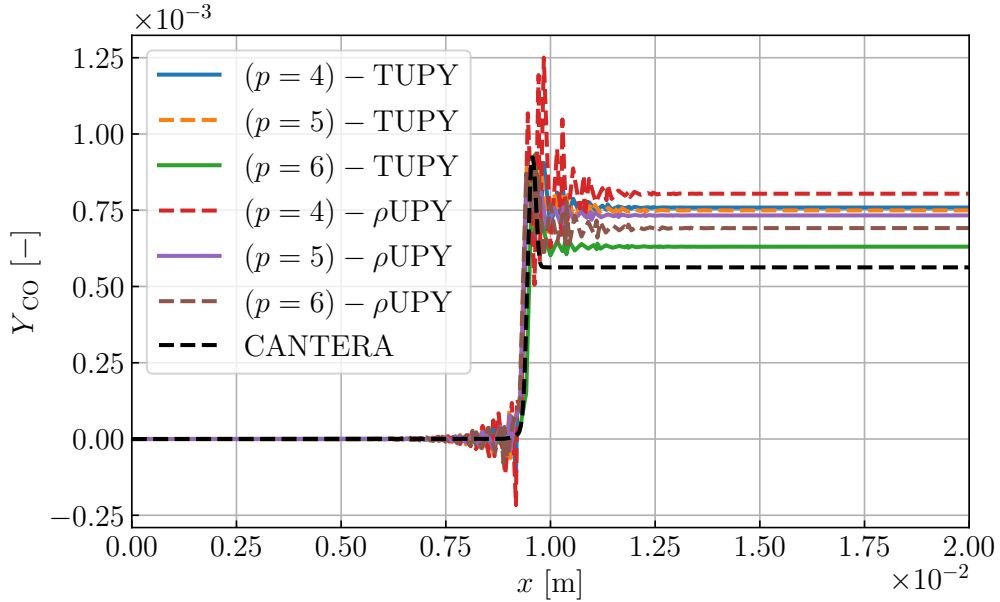


Figure 7.29. – Comparison of the CO mass fraction between calculations at $p = 4$, $p = 5$ and $p = 6$ with either TUPY or ρ UPY approach at FP on a 1D premixed flame with $n_{pts}^{ff} \approx 4$.

$p = 5$ where ρ UPY is slightly better than TUPY, this other approach has a behavior even worst than TUPY approach when the flame front discretization becomes coarser. Therefore TUPY approach was preferred for all calculations during this work.

A second idea was to consider a new formulation for the diffusion scheme, the SDLIFT formulation, which is shown in the next paragraph.

7.5.2. 1D methane/Air premixed flame with four points inside the flame front using SDLIFT formulation for the diffusion scheme

The 1D premixed flame of paragraph 7.5.1 is again considered in the worst case where $N_e = 37$ and $p = 4$, in order to highlight the impact of using the SDLIFT formulation. This new formulation was described in paragraphs 4.1.7 and 4.4.2 respectively for 1D and 3D cases. Three parameters define a SDLIFT process:

1. The chosen diffusion scheme for computing common states and common gradients at interface FP. In this work BR1 (paragraph 4.1.7.2), BR2 (paragraph 4.1.7.3) and IC (paragraph 4.1.7.4) schemes have been introduced and tested.
2. The correction function used to have the correct polynomial degree for the normal derivative in isoparametric space. In this work, two different correction functions, namely g_{DG} and g_{Ga} , have been employed and were explained in paragraph 4.1.7.1.
3. The way tangential derivatives in the isoparametric space are obtained to be used to compute derivatives in the physical space. Two choices were mentioned in this work at the end of paragraph 4.4.2: the classical approach or Huynh's approach. However, in the case of 1D calculations both approach are necessary equivalent since there are no tangential derivatives in the isoparametric space to be used. Thus, this parameter is not consider in this paragraph.

Therefore, there are 3 diffusion schemes that can be coupled with 2 correction functions so that 6 different calculations were simulated. Results are represented in Figure 7.30 for the CO mass fraction because this species was found the most oscillating variable in paragraph 7.5.1. The results for IC-scheme used along with g_{Ga} correction function is not shown for clarity since they are even worst than with the original formulation. The first thing to notice is that BR1- g_{Ga} gives exactly the same result as the original approach like it was stated in paragraph 4.1.7.2. The BR1 diffusion scheme coupled with g_{DG} has strong oscillations but is still better than the original formulation in the burnt gases. The BR2 diffusion scheme, either with g_{Ga} or g_{DG} , is well more stable than the BR1 one with almost no oscillations within the flame front and a value of Y_{CO} in the burnt gases very close to the CANTERA reference value. Finally, the IC- g_{DG} formulation actually gives the best result in this case without oscillations and the closest Y_{CO} value to the CANTERA reference. Therefore, the use of the SDLIFT formulation considerably improves the results when the flame front is discretized with few points. In particular, BR2 and IC diffusion schemes strongly improve the stability of the method. It shows that, when the flame front is badly discretized, a small stencil (BR2 and IC have a 3 elements stencil whereas BR1 has a 5 elements stencil as explained in paragraph 4.1.7) for computing common gradient values is better in terms of stability. Moreover, it also seems that employing g_{DG} , with any diffusion scheme, ends up with more accurate results. This was also expected from the explanation of paragraph 4.1.7.1.

7.5.3. Impact of the SDLIFT formulation with ARC schemes

In paragraph 7.5.2, the use of the SDLIFT formulation for computing diffusive fluxes at interface FP was evaluated for a two-reactions chemical scheme, i.e., without stiff reactions and highly reacting species. The observed improvement may not be so clear when using an ARC scheme with low flame resolution. Keeping the same number of DOF in the flame front, $n_{pts}^{ff} \approx 4$, the 1D premixed flame already simulated in paragraph 7.2.2 using the CH4_16_250_10_QC ARC mechanism is considered here, with $N_e = 37$ and $p = 4$ as in paragraphs 7.5.1 and 7.5.2. As the flame thickness is almost the same for the CH4/Air-2S-BFER flame of Table 7.1 and the CH4_16_250_10_QC flame of Table 7.4, the same mesh leads to similar flame resolution. Among the 16 transported species of

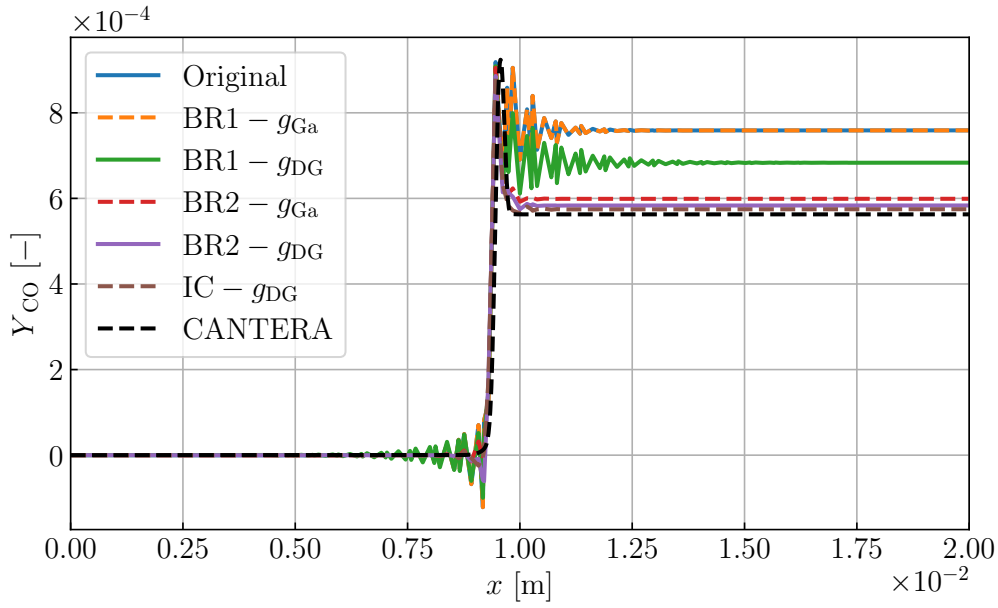


Figure 7.30. – Comparison of the CO mass fraction obtained with different pairs of diffusion schemes and correction functions when using the SDLIFT formulation on a 1D premixed flame with $n_{pts}^{ff} \approx 4$.

the CH4_16_250_10_QC ARC mechanism, several highly reacting species are found: C_2H_4 , HO_2 , CH_2O and H , which are selected to evaluate the SDLIFT formulations. The results are gathered in Figure 7.31 where, for clarity reasons, a zoom was done around the peak for some species.

Again, the use of diffusion schemes from the SDLIFT formulation really improves JAGUAR results: the original formulation is very unstable as it can be seen on Figure 7.31d for HO_2 species whereas the two BR2 schemes and the IC scheme are found very stable. The AVBP solution does not show strong oscillations but some species mass fractions, such as Y_{CH_2O} and $Y_{C_2H_4}$, become negative in the burnt gases like for the original formulation of JAGUAR. However, none of the diffusion schemes of the SDLIFT formulation shows such behavior which advocates for their stability properties even in the presence of a stiff chemical mechanism. Therefore, the SDLIFT formulation seems also well more robust than the original formulation when stiff reactions and highly reacting species are considered.

7.5.4. Impact of the SDLIFT formulation on a 2D methane/air circular flame

The same 2D circular flame simulated in paragraph 7.3.2 is considered here to evaluate the SDLIFT formulation in 2D. In paragraph 7.3.2, it was found that JAGUAR and AVBP give almost the same results on this flame when $n_{pts}^{ff} \approx 8$. The only very small difference was on the evolution of S_c with time (see Figure 7.20) but it remained marginal. As for 1D flames, it is proposed to see how smaller values of n_{pts}^{ff} impact this evolution of S_c with time for both codes. Firstly, it is essential to check that JAGUAR and AVBP give exactly the same results, in terms of S_c , for very fine discretizations before coarsening the meshes. Consequently, the 2D circular flame was computed with JAGUAR and AVBP using this time 16×10^6 DOF in order to have $n_{pts}^{ff} = 16$. JAGUAR simulation was carried out on a mesh with $N_e = 64000$ uniform quadrilateral elements (800 1D elements in each direction) with $p = 4$ inside each of them and for AVBP an uniform Cartesian mesh with 4000 nodes along each direction has been used. Figure 7.32 represents the evolution of $S_c(t)$ for these very fine discretizations showing that the two codes give exactly the same results when $n_{pts}^{ff} = 16$. This solution is then referred to the reference solution for this test case.

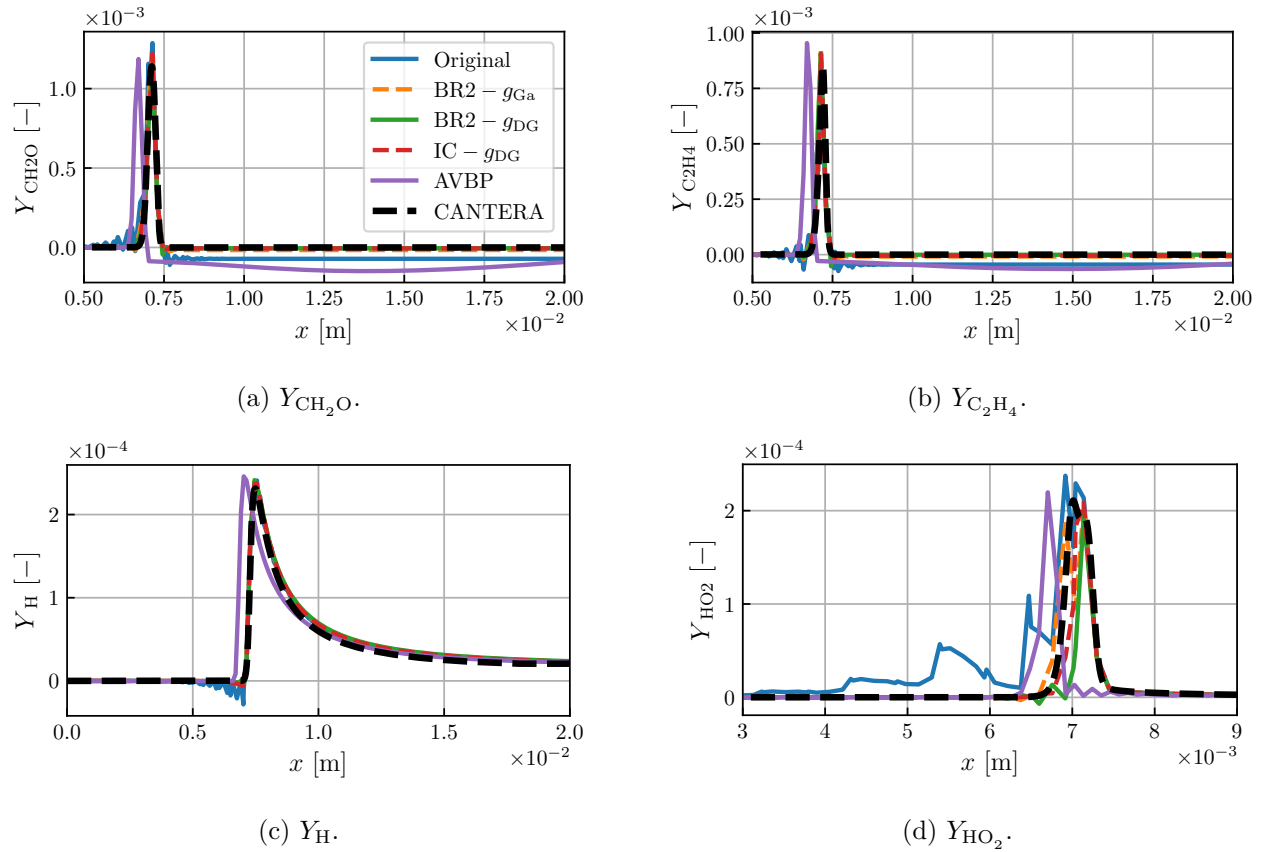


Figure 7.31. – Comparison of $Y_{\text{CH}_2\text{O}}$, $Y_{\text{C}_2\text{H}_4}$, Y_{H} and Y_{HO_2} profiles between original and several SDLIFT formulations on a 1D CH_4/Air premixed flame using an ARC mechanism with $n_{pts}^{ff} \approx 4$. CANTERA solution is added as reference and AVBP solution is also shown for comparison.

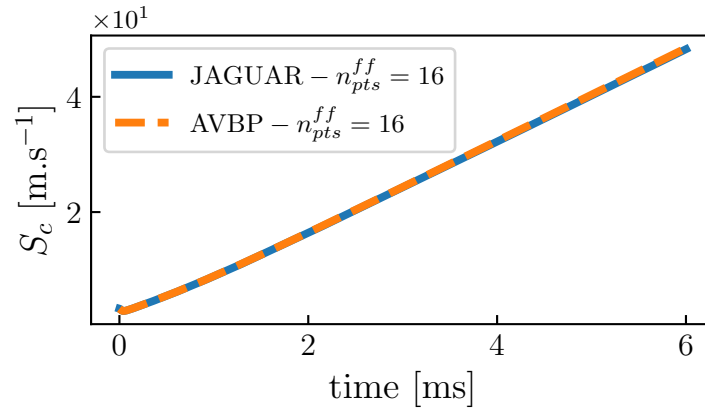


Figure 7.32. – Evolution of $S_c(t)$ for JAGUAR and AVBP simulations when $n_{pts}^{ff} = 16$.

Going now to coarser discretizations, cases with $n_{pts}^{ff} = 4$ and $n_{pts}^{ff} = 2$ corresponding respectively to 1000 and 500 DOF per direction in the computational domain in Figure 7.18, are considered. JAGUAR simulations are done here with $p = 4$ and the IC diffusion scheme coupled with g_{DG} correction function. As already observed in paragraph 7.5.1, this scheme gives a stable solution, shown in Figure 7.33 with

a cut of temperature profile at $y = 0$ along x -axis at $t = 6$ ms. Results obtained with JAGUAR for $n_{pts}^{ff} = 8$ and with AVBP for the same flame resolutions are also reported for comparison. Surprisingly, JAGUAR and AVBP do not show any oscillating behavior whatever the value of n_{pts}^{ff} and without artificial viscosity for AVBP simulations. However, the flame propagation deviates from the reference solution when n_{pts}^{ff} decreases. This deviation is more important with AVBP than with JAGUAR.

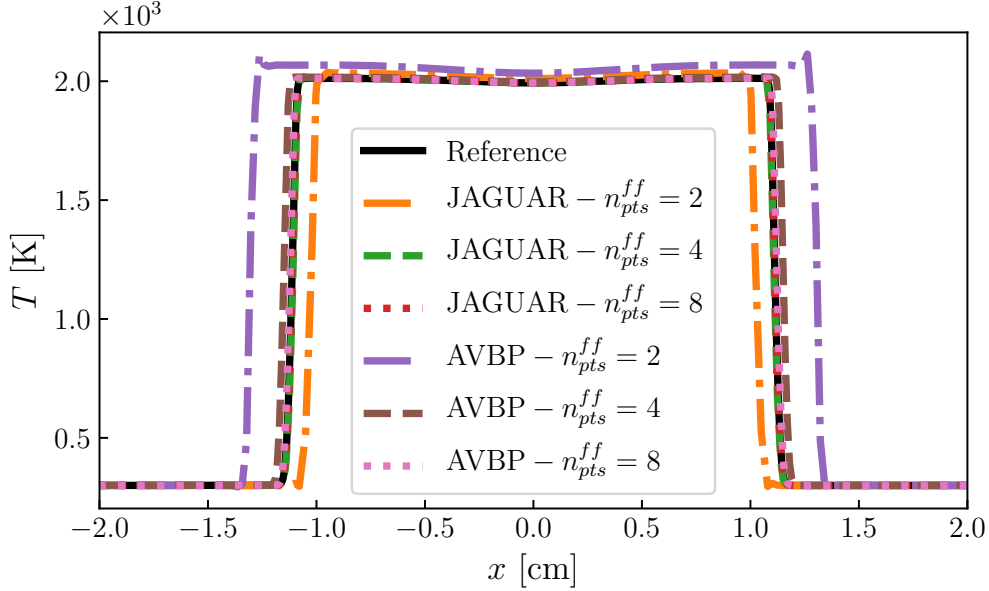


Figure 7.33. – Cut of temperature profile at $y = 0$ along x -axis obtained with JAGUAR and AVBP at $t = 6$ ms when $n_{pts}^{ff} = 2$, $n_{pts}^{ff} = 4$ and $n_{pts}^{ff} = 8$.

These deviations reflect on the evolution of $S_c(t)$ plotted in Figure 7.34 for all cases. For $n_{pts}^{ff} = 4$, the AVBP simulation has a consumption speed higher than the reference whereas the JAGUAR simulation at the same n_{pts}^{ff} value is still following the reference. When $n_{pts}^{ff} = 2$, the AVBP solution propagates well faster than the reference whereas the JAGUAR solution for this n_{pts}^{ff} value underpredicts significantly the flame propagation. However, in absolute value, JAGUAR results are more accurate and seem less sensitive to the value of n_{pts}^{ff} .

7.6. Summary of this chapter

In this chapter, an extension of the SD method for considering a thermally multi-species perfect gas is presented. It is demonstrated that the usual SD algorithm extrapolating conservative variables from SP to FP is unstable for such gas. It appeared that computing primitive variables from conservatives at SP and then extrapolating primitive variables at FP is much more stable and seemed sufficient for the cases considered in this work.

One-dimensional laminar premixed flames using different chemistries of increasing stiffness were firstly carried out to validate chemical source terms computation, implementation of thermodynamics for a multi-species gas and the resolution of species transport equations. Results were in excellent agreement with the ones given by reference combustion solvers AVBP and CANTERA. Going a step further in the validation, two-dimensional laminar premixed flames have been conducted with additional boundary conditions such as walls and symmetries. As for the one-dimensional flames, results were similar to the ones of the AVBP solver.

The interest and the potential of using the SD method in combustion was studied on a two-

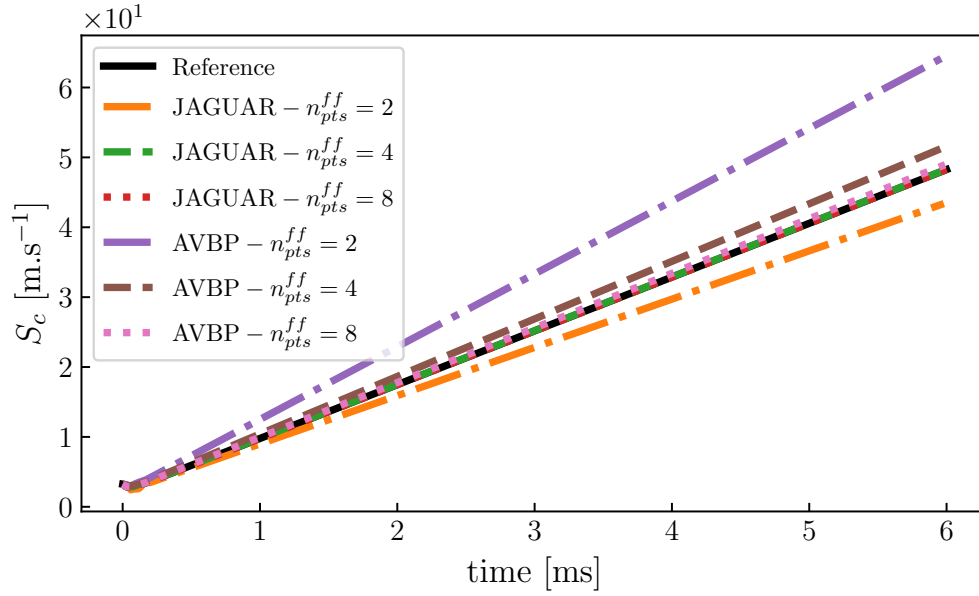


Figure 7.34. – Evolution of $S_c(t)$ for JAGUAR and AVBP simulations when $n_{pts}^{ff} = 2$, $n_{pts}^{ff} = 4$ and $n_{pts}^{ff} = 8$.

dimensional laminar burner case. It was found that the use of small N_e with high p values gives same error level with less DOF than by using large N_e with small p values. Consequently, it is also faster to employ small N_e with high p values to reach the same error level. At a constant DOF value, the iteration cost of the SD method is almost constant with p and is nowadays close to the one of the TTGC convective solver in AVBP. This property is new compared to what was usually observed for classical numerical methods where an increase of the scheme order often results in an increase of the iteration cost. The current issue with the SD method, and more generally with HO methods, is that the timestep has to be decreased with increasing p values for stability reasons. Thus, turnaround times are still higher than with classical numerical methods. However, it was shown that using local polynomial adaptation drastically reduces the number of DOF while keeping the same accuracy level leading to more than a 30% gain in computational time for the same result. It clearly advocates for pursuing the development of local polynomial adaptation for HO methods to remain competitive in terms of computational time compared to classical numerical methods since they already show their capability of obtaining better results for the same resolution.

Finally, laminar one and two-dimensional premixed flames were simulated with very coarse discretizations to see the lower limit in terms of n_{pts}^{ff} for the SD method. These test cases have shown that the original formulation for the diffusion scheme of Sun *et al.* [47] is very unstable when $n_{pts}^{ff} \leq 4-5$. It has motivated the development of the SDLIFT formulation which allows to use other diffusion schemes such as BR2 or IC schemes. They proved to be well more more stable when n_{pts}^{ff} is low compared to the original formulation. This observation is not dependent on the stiffness of the chemical scheme. Moreover, the SD method using SDLIFT formulation better predicts flame propagation for very coarse discretization than the AVBP solver as highlighted in paragraph 7.5.4.

Simulations of turbulent premixed combustion using the Spectral Difference method

Overview

| | |
|---|------------|
| 8.1. TFLES model in the context of the Spectral Difference method | 184 |
| 8.1.1. Implementation of the TFLES model within the SD framework | 184 |
| 8.1.2. One-dimensional Methane/Air flame using a two-reactions chemistry and the thickened flame model | 185 |
| 8.1.3. One-dimensional Methane/Air flame using Analytically Reduced Chemistry and the thickened flame model | 187 |
| 8.1.4. Two-dimensional Methane/Air burner using a two-reactions chemistry and the thickened flame model | 188 |
| 8.1.5. Validation of the subgrid-scale velocity computation | 189 |
| 8.2. Positivity-preserving limiter | 191 |
| 8.3. Simulation of the Cambridge flame burner | 193 |
| 8.3.1. Presentation of the case | 193 |
| 8.3.2. Computational domain | 194 |
| 8.3.3. Boundary conditions | 194 |
| 8.3.4. Mesh discretization | 195 |
| 8.3.5. Chemistry and combustion model | 195 |
| 8.3.6. Other numerical parameters | 195 |
| 8.3.7. Post-treatment | 196 |
| 8.3.8. Results analysis on case SwBc | 196 |
| 8.3.9. Results analysis on case SwB1 | 196 |
| 8.3.10. Computational cost | 199 |
| 8.3.11. Perspectives for future works on case SwB1 using local polynomial adaptation | 202 |
| 8.4. Simulation of the VOLVO configuration | 204 |
| 8.4.1. Presentation of the case | 204 |
| 8.4.2. Mesh discretization | 205 |
| 8.4.3. Boundary conditions | 205 |
| 8.4.4. Chemistry and combustion model | 205 |
| 8.4.5. Other numerical parameters | 206 |
| 8.4.6. Post-treatment | 206 |
| 8.4.7. Results analysis on the non-reacting case | 206 |
| 8.4.8. Results analysis on the reacting case | 210 |
| 8.5. Summary of this chapter | 215 |

This chapter is dedicated to the simulation of turbulent premixed flames using the SD method. The implementation of the TFLES model within the SD formalism is described and first validations for the flame thickening are performed on 1D and 2D laminar flames. Then, details are given on the implementation of a positivity-preserving limiter, needed for stabilizing the 3D turbulent combustion cases presented in this chapter. Finally, two simulations of 3D turbulent premixed flames are presented: the Cambridge flame burner and the VOLVO configuration.

8.1. TFLES model in the context of the Spectral Difference method

The TFLES model was introduced in Section 3.3 in a general way, independently of the numerical method. The objective of this section is to explain the implementation of the TFLES model within a SD discretization. Details are given on how the diffusion coefficients and combustion source terms are modified, how the relaxation flame sensor is calculated and how the SGS velocity used in the efficiency model is evaluated.

8.1.1. Implementation of the TFLES model within the SD framework

As described in paragraph 3.3.3, the TFLES model modifies diffusion coefficients λ and D_k as well as species source terms $\dot{\omega}_k$ following Eq. (3.41). For a SD discretization, λ and D_k are evaluated at FP, because they are needed for the computation of diffusive fluxes, whereas $\dot{\omega}_k$ are evaluated at SP. Thus, values of \mathcal{E} and \mathcal{F} must be available at FP, noted \mathcal{E}_{FP} and \mathcal{F}_{FP} , and at SP, noted \mathcal{E}_{SP} and \mathcal{F}_{SP} . In this work, it was chosen to compute \mathcal{E}_{SP} and \mathcal{F}_{SP} and to interpolate them at FP as it is done for the solution vector. The next two paragraphs explain the calculation of these four quantities in both constant thickening and dynamic thickening cases.

8.1.1.1. Constant thickening case

As mentioned in paragraph 3.3.1, the constant thickening model applies the same thickening value \mathcal{F} everywhere in the computational domain. Consequently, as \mathcal{F}_{FP} and \mathcal{F}_{SP} are both equal to \mathcal{F} no interpolation is required. This is not the case for efficiency \mathcal{E} , which is first evaluated at SP using either Eq. (3.55) (non-saturated Charlette's model) or Eq. (3.57) (saturated Charlette's model) and then interpolated at FP to get \mathcal{E}_{FP} . Note that in the case of saturated Charlette's model with constant β , since $\Delta = \mathcal{F}\delta_L^0$:

$$\mathcal{E}_{FP} = \mathcal{E}_{SP} = \mathcal{F}^\beta = \mathcal{E} \quad (8.1)$$

8.1.1.2. Dynamic thickening case

In the dynamic thickening model, a flame sensor $\mathcal{S}(\mathbf{x}, t)$ is employed. The relaxation flame sensor of Jaravel *et al.* [175] used in this work was explained in paragraph 3.3.2.2. Since this sensor is based on the transport of a fictive species ψ , the resolution of Eq. (3.35) was implemented in JAGUAR. It has been checked that both AVBP and JAGUAR show exactly the same solution to this equation on a given case (see paragraph 8.1.2). The different steps to apply the TFLES model with the relaxation flame sensor and Charlette's model, using a constant β value, are summed up in Algorithm 7 of Section B.5. Basically, it needs as inputs: n_{pts}^{ff} , δ_L^0 , S_L^0 , $|\dot{\omega}_F|_{1D}^{max}$, α_{cold} , α_{hot} , Sc_ψ , the β value needed in Charlette's model and the type of Charlette's formulation. At the very first time iteration (not a restart solution), there is no field of ψ in the domain so that \mathcal{F}_{SP} is initialized as follows:

1. $\dot{\omega}_F$ is evaluated at all SP assuming $\mathcal{F}_{SP} = \mathcal{E}_{SP} = 1$.
2. \mathcal{S} at all SP is computed, noted \mathcal{S}_{SP} , based on Eq. (3.34).
3. Then, the fictive species is initialized at SP with \mathcal{S} by setting $[\rho\psi]_{SP} = [\rho\mathcal{S}]_{SP}$.

4. Finally, \mathcal{F}_{SP} is initialized using Eq. (3.38) with $\hat{\mathcal{S}}_{SP} = \mathcal{S}_{SP}$.

For perfect restart, $\rho\psi$ is saved in the solution files so that Eq. (3.37) and Eq. (3.38) can be successively employed to get \mathcal{F}_{SP} . \mathcal{E}_{SP} does not need specific treatment at initialization because either Eq. (3.55) or Eq. (3.57) can be computed.

It should be mentioned that this implementation assumes implicitly that δ_L^0 , S_L^0 and $|\dot{\omega}_F|_{1D}^{max}$ are constant during the computation. However, for flames that are not perfectly premixed, that is if ϕ changes locally, δ_L^0 , S_L^0 and $|\dot{\omega}_F|_{1D}^{max}$ vary and have to be computed on the fly during the computation at each SP. In that case, $\tau_{1,cold}$ and \mathcal{F}_{max} also vary and steps 2 and 3 of Algorithm 7 have to be done at all SP at each time step. Usually, δ_L^0 , S_L^0 and $|\dot{\omega}_F|_{1D}^{max}$ are pre-tabulated as functions of ϕ . This tabulation was not implemented in JAGUAR during this work and is left for future developments.

8.1.2. One-dimensional Methane/Air flame using a two-reactions chemistry and the thickened flame model

The objective of this test case is to validate the implementation in JAGUAR of the TF model described in paragraph 8.1.1, firstly for a two-reactions chemistry. In particular, it will be checked that the expected thickening factor is reached and that the fictive species for the relaxation sensor is correctly transported. The same one-dimensional methane-air premixed flame of Table 7.1 is considered. The computational domain is discretized with $N_e = 20$ elements and a polynomial degree $p = 4$ leading to 100 DOF in the domain. This is four times less resolution than the case presented in paragraph 7.2.1, so that the flame will be thickened by a factor four. For the constant thickening case, this means that $\mathcal{F}_{max} = 4$ and $\mathcal{F}_{SP} = \mathcal{F}_{FP} = \mathcal{F}_{max} = 4$. For the dynamic thickening case, $\mathcal{F}_{max} = 4$ and \mathcal{F}_{SP} and \mathcal{F}_{FP} are computed following the methodology described in paragraph 8.1.1.2 where flame sensor parameters are given in Table 8.1. The flame thickness and flame speed used to compute τ_1 with Eq. (A.35) are the ones of Table 7.1. The value of n_{pts}^{ff} is not needed here since \mathcal{F}_{max} is imposed to 4 at all SP for easier validation. Finally, the efficiency function \mathcal{E} is kept to one everywhere because the case is laminar.

| Sc_ψ [-] | α_{cold} [-] | α_{hot} [-] | $ \dot{\omega}_F _{1D}^{max}$ [W.m ⁻³] |
|---------------|---------------------|--------------------|--|
| 0.2 | 0.1 | 0.01 | 74 |

Table 8.1. – Flame sensor parameters for the dynamic thickening of the one-dimensional methane-air premixed flame using the CH₄/Air-2S-BFER chemical mechanism at $\phi = 0.8$.

Figure 8.1 shows the heat release rate profiles for both AVBP and JAGUAR simulations with the same constant and dynamic thickening. CANTERA (non thickened) solution is also reported for comparison. As expected, in simulations with the TF model the flame region is well thickened and the maximum of heat release rate is four times smaller than the maximum value given by CANTERA without thickening. Constant and dynamic thickening models give the same results since in this perfectly premixed case no diffusion occurs outside the reaction zone. In order to check that the thickening of the flame does not affect the flame structure, a transformation of the space variable can be done following Eq. (8.2) [175]:

$$x^* = \int_0^{L_x} \frac{dx}{\mathcal{F}(x)} \quad (8.2)$$

In the x^* -space, the thickened flames must have the same structure than the non-thickened flames in the x -space (the one of CANTERA). This is indeed the case in Figure 8.2 comparing temperature profiles of all JAGUAR and AVBP solutions in x^* -space to the CANTERA solution in x -space. Finally, Figure 8.3 compares the fictive species mass fraction profiles of AVBP and JAGUAR which appear to be identical meaning that Eq. (3.35) is well-resolved by JAGUAR. To conclude on the implementation,

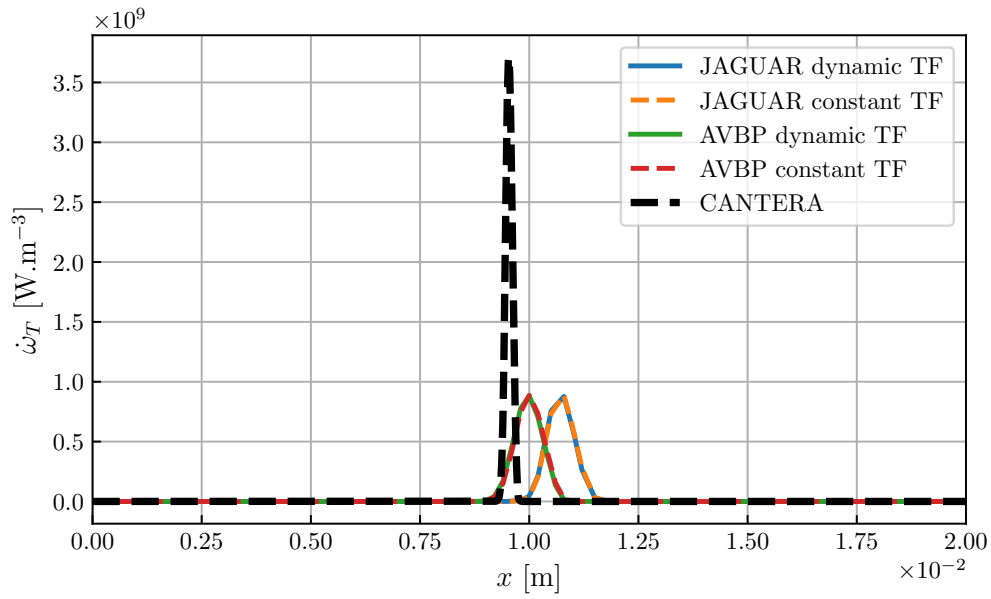


Figure 8.1. – Comparison of heat release rate profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH₄/Air-2S-BFER chemical scheme at $\phi = 0.8$. Non-thickened CANTERA solution is also plotted for comparison.

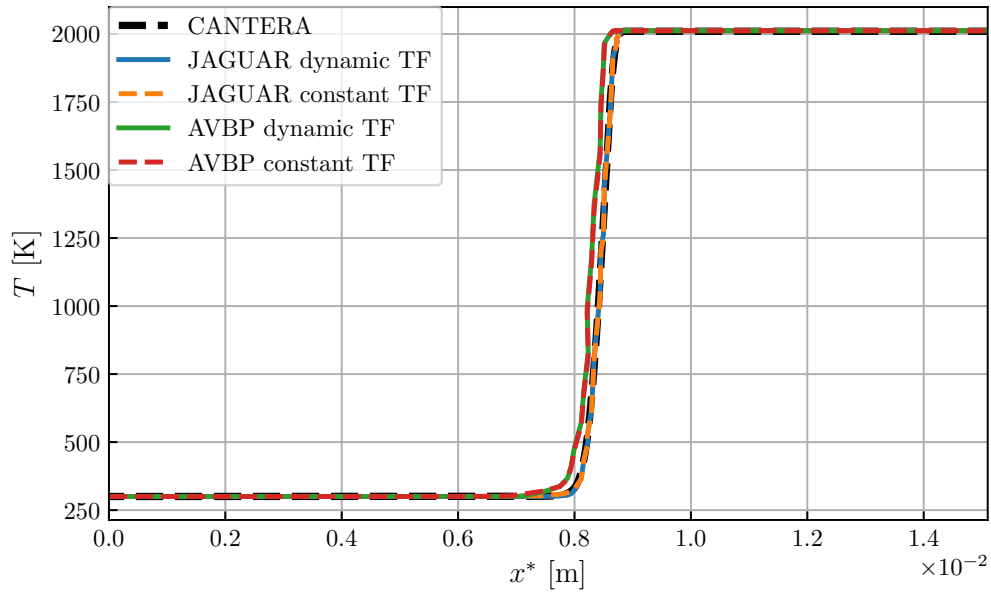


Figure 8.2. – Comparison of temperature profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH₄/Air-2S-BFER chemical scheme at $\phi = 0.8$ in the x^* -space. Non-thickened CANTERA solution in the x -space is plotted for comparison.

flame speeds are reported in Table 8.2. By construction, both the non-thickened and the thickened flame simulations must give the same flame speed. This is exactly verified by JAGUAR, whereas AVBP shows small differences between the two TF approaches and between the thickened and non-thickened approaches.

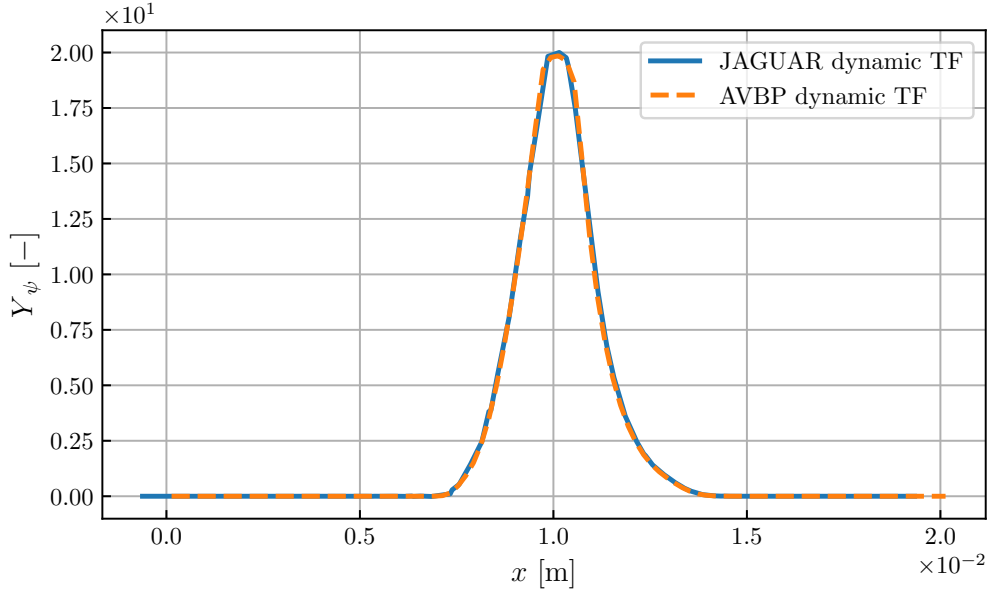


Figure 8.3. – Comparison of fictive species mass fraction profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH₄/Air-2S-BFER chemical mechanism at $\phi = 0.8$.

| Code | JAGUAR dyn TF/cte TF/no TF | AVBP dyn TF/cte TF/no TF |
|-------------------------------|----------------------------|--------------------------|
| S_L^0 [cm.s ⁻¹] | 28.16/28.16/28.16 | 28.33/28.35/28.31 |

Table 8.2. – Comparison of flame speeds between JAGUAR and AVBP for dynamic, constant thickening and without any thickening for a 1D premixed methane-air flame using the CH₄/Air-2S-BFER chemical mechanism at $\phi = 0.8$.

8.1.3. One-dimensional Methane/Air flame using Analytically Reduced Chemistry and the thickened flame model

The objective of this test case is to validate the implementation in JAGUAR of the TF model described in paragraph 8.1.1 for an ARC mechanism. The same one-dimensional methane-air premixed flame of Table 7.4 is considered. To apply the TF model, the computational domain is now discretized with $N_e = 20$ elements and a polynomial degree of $p = 4$. Thus, \mathcal{F}_{max} is again set to 4 and constant and dynamic thickening of the flame front are considered. Flame sensor parameters used here are summed up in Table 8.3 and flame thickness and flame speed used to compute τ_1 are the ones of Table 7.4. Finally, the efficiency function \mathcal{E} is again set to one everywhere since the flame is not turbulent. Temperature profiles in the x^* -space for both AVBP and JAGUAR using constant and dy-

| Sc_ψ [-] | α_{cold} [-] | α_{hot} [-] | $ \dot{\omega}_F _{1D}^{max}$ [W.m ⁻³] |
|---------------|---------------------|--------------------|--|
| 0.2 | 0.1 | 0.01 | 133 |

Table 8.3. – Flame sensor parameters for the dynamic thickening of the one-dimensional methane-air premixed flame using an ARC chemical mechanism at $\phi = 1.0$.

dynamic thickening are shown in Figure 8.4 along with the reference CANTERA non-thickened solution. Contrary to the CH₄/Air-2S-BFER case, the constant and dynamic thickenings now give different results in the burnt gases. This comes from the fact that with ARC mechanisms, the post-flame is still a reacting zone slowly progressing towards equilibrium. Therefore, as constant thickening applies

$\mathcal{F} > 1$ also in this zone, it slows down the chemistry there and this slow-down is not compensated by over-diffusion as temperature and species gradients are close to zero. On the contrary the dynamic model, by keeping unchanged the post-flame zone, allows to recover CANTERA reference results. This however has a minor impact on the flame speed prediction, shown in Table 8.4: flame speeds are very close between constant and dynamic thickenings for both JAGUAR and AVBP.

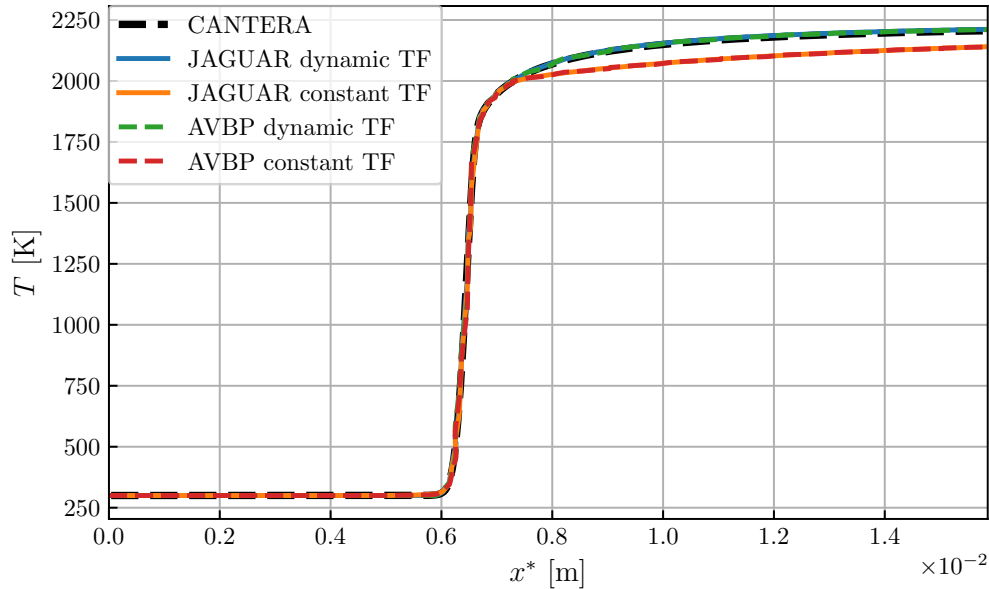


Figure 8.4. – Comparison of temperature profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using an ARC chemical mechanism at $\phi = 1.0$ in the x^* -space. Non-thickened CANTERA solution in the x -space is plotted for comparison.

| Code | JAGUAR dyn TF/cte TF/no TF | AVBP dyn TF/cte TF/no TF |
|-------------------------------|----------------------------|--------------------------|
| S_L^0 [cm.s ⁻¹] | 37.94/37.98/38.06 | 38.01/37.93/38.09 |

Table 8.4. – Comparison of flame speeds between JAGUAR and AVBP for dynamic, constant thickening and without any thickening for a thickened 1D premixed methane-air flame using an ARC chemical mechanism at $\phi = 1.0$.

8.1.4. Two-dimensional Methane/Air burner using a two-reactions chemistry and the thickened flame model

The objective of this test case is to simulate the same 2D burner presented in Figure 7.13 now using the TF model in its dynamic formulation. This allows to check the TF model behavior in the presence of walls and symmetries. All boundary conditions remain unchanged and the CH₄/Air-2S-BFER scheme at $\phi = 0.8$ is still considered. However, as the aim is to thicken the flame front, a coarser discretization is employed with this time only 486 quadrilateral elements with again a polynomial degree of $p = 4$. The total number of DOF for the JAGUAR simulation is 12150 and AVBP simulation is discretized with 12182 DOFs to have a fair comparison. The element characteristic size is around $\Delta_e = 6.8 \times 10^{-4}$ m which is almost three times higher than the Δ_e of paragraph 7.3.1. The maximum thickening was computed using Eq. (3.32) with $n_{pts}^{ff} = 8$ which gives \mathcal{F}_{max} around 2.95 for both AVBP and JAGUAR. Finally, values for Sc_ψ , α_{cold} , α_{hot} and $|\dot{\omega}_F|_{1D}^{max}$ are the ones of Table 8.1 and flame speed and thickness are still those shown in Table 7.1. Figure 8.5 shows the 2D heat release rate fields

obtained with both JAGUAR and AVBP when steady state is reached. The agreement between AVBP and JAGUAR is very good, with a well thickened flame front. Figure 8.6 compares temperature and

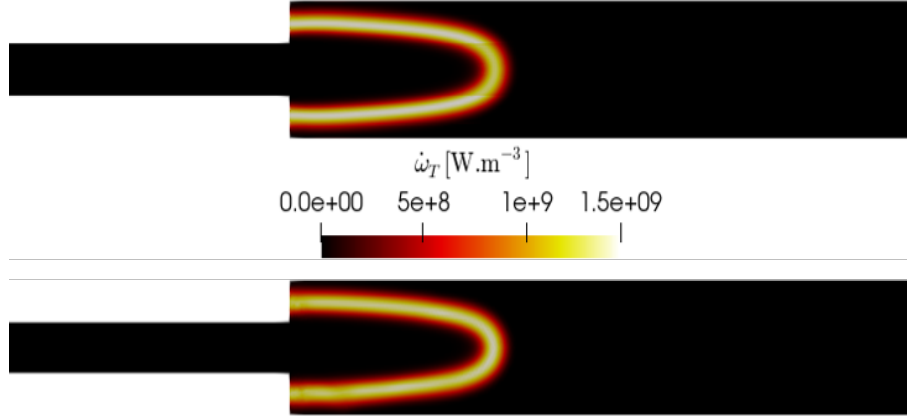


Figure 8.5. – Comparison between JAGUAR (top) and AVBP (bottom) of the 2D heat release rate field for the 2D burner case using the TF model.

heat release rate vertical profiles at $x = 12$ mm between JAGUAR and AVBP: both profiles are in accordance meaning that the TF model is correctly implemented in JAGUAR.

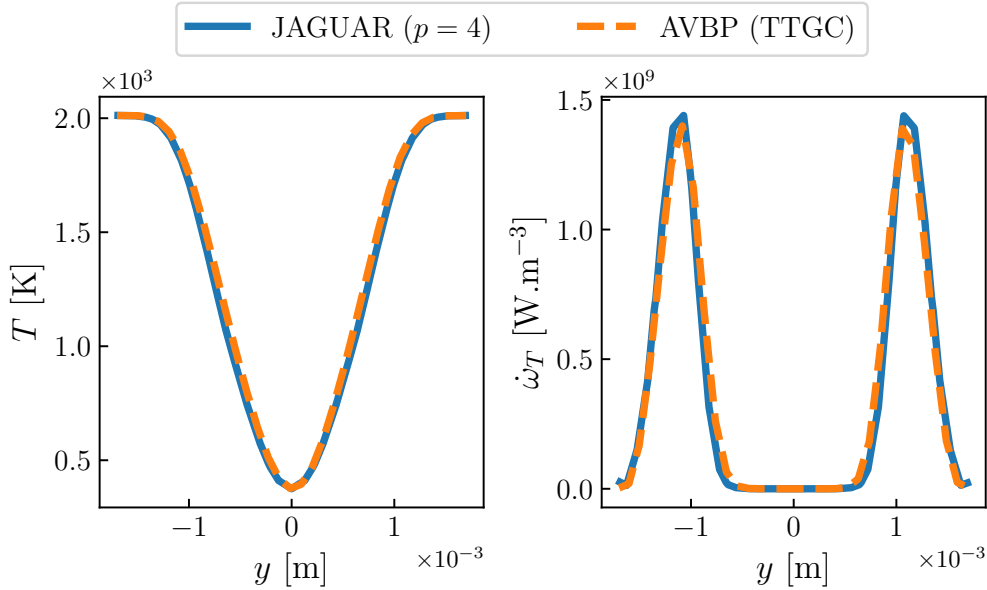


Figure 8.6. – Comparison between JAGUAR and AVBP of temperature and heat release rate profiles at $x = 12$ mm along y -axis for the 2D burner case using the TF model.

8.1.5. Validation of the subgrid-scale velocity computation

8.1.5.1. Position of the problem and presentation of the validation test case

In the case of the saturated Charlette's model (see paragraph 3.3.4.4), \mathcal{E} is directly obtained from \mathcal{F} since $\Delta/\delta_L^0 = \mathcal{F}$. However, for the non-saturated formulation (see paragraph 3.3.4.2), the SGS velocity u'_Δ , evaluated with the operator $\|\nabla^2(\nabla \times \tilde{\mathbf{u}})\|_2$ (see Eq. (3.53)) is needed. In this paragraph, the validation of this operator using the SD discretization is detailed. To that purpose, the velocity

field corresponding to an incompressible and non-viscous vortex is used, given by Eq. (8.3) [294]:

$$\mathbf{u}(r) = \frac{Cr}{R_c^2} \exp\left(\frac{-r^2}{2R_c^2}\right) \mathbf{e}_\theta \equiv u_\theta(r) \mathbf{e}_\theta \quad (8.3)$$

where C determines the vortex strength, R_c is the vortex radius and r the radial coordinate with the origin taken at the vortex center. In Eq. (8.3), \mathbf{e}_θ is the unit vector in the orthoradial direction of the cylindrical basis so that: $\mathbf{e}_\theta = -\sin(\theta) \mathbf{e}_x + \cos(\theta) \mathbf{e}_y$. Taking the derivative of Eq. (8.3) with respect to r gives:

$$\frac{du_\theta(r)}{dr} = \frac{C}{R_c^2} \left(1 - \frac{r^2}{R_c^2}\right) \exp\left(\frac{-r^2}{2R_c^2}\right) \quad (8.4)$$

Then, all maximum values are located at $r = R_c$ for which:

$$u_\theta^{max} \equiv u_\theta(r = R_c) = \frac{C}{R_c} \exp\left(\frac{-1}{2}\right) \quad (8.5)$$

Consequently, this vortex velocity field can be defined with R_c and u_θ^{max} . In this case, Eq. (8.3) is employed with $R_c = 6$ mm and $u_\theta^{max} = 1$ m.s⁻¹. The corresponding velocity field is represented in Figure 8.7a along with a radial profile on the horizontal centerline for $r \geq 0$ in Figure 8.7b. The computational domain is a cube of size $L = 6$ cm discretized with 75 uniform segments along x and y directions and only 1 segment in the z direction, and a polynomial degree $p = 4$.

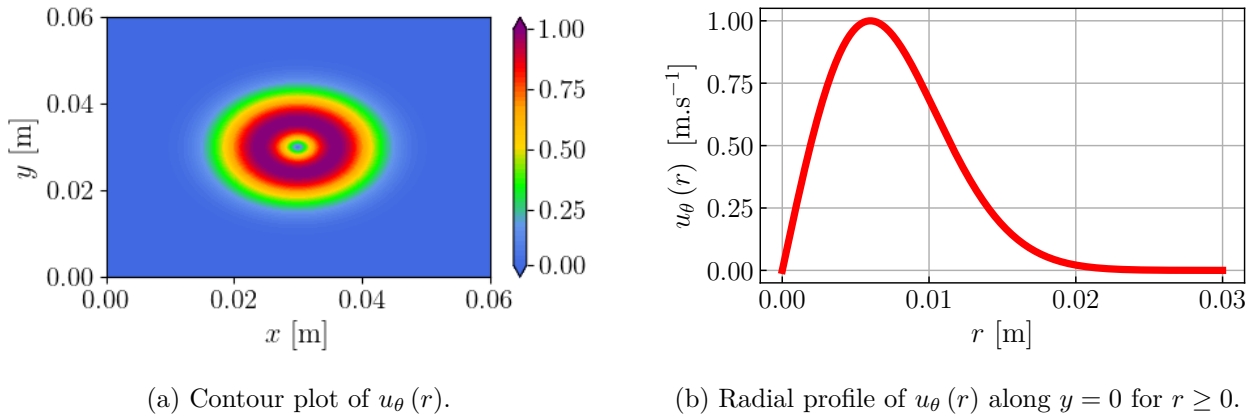


Figure 8.7. – Contour plot (left) and radial profile along $y = 0$ for $r \geq 0$ (right) of u_θ when $R_c = 6$ mm and $u_\theta^{max} = 1$ m.s⁻¹.

8.1.5.2. Analytical solution

It is possible to compute $\|\nabla^2(\nabla \times \mathbf{u})\|_2$ analytically for the velocity field in Eq. (8.3), allowing to validate the numerical operator of JAGUAR. Firstly, $\nabla \times \mathbf{u}$ is computed using formula for the rotational operator in cylindrical coordinates:

$$\nabla \times \mathbf{u} = \frac{1}{r} \frac{\partial}{\partial r} (ru_\theta) \mathbf{e}_z = \frac{C}{R_c^2} \exp\left(\frac{-r^2}{2R_c^2}\right) \left[2 - \frac{r^2}{R_c^2}\right] \mathbf{e}_z \quad (8.6)$$

Then, the gradient in cylindrical coordinates of Eq. (8.6) is employed ending up with a 3×3 matrix A whose components are zero except for A_{31} which is given by:

$$\nabla(\nabla \times \mathbf{u}) = A_{31} = \frac{\partial}{\partial r} (\nabla \times \mathbf{u}) = \frac{-Cr}{R_c^4} \left[4 - \frac{r^2}{R_c^2}\right] \exp\left(\frac{-r^2}{2R_c^2}\right) \quad (8.7)$$

Finally, the divergence in cylindrical coordinates of Eq. (8.7) is taken ending up with a vector along \mathbf{e}_z for $\nabla^2(\nabla \times \mathbf{u}) = \nabla \cdot [\nabla(\nabla \times \mathbf{u})]$:

$$\nabla^2(\nabla \times \mathbf{u}) = \left(\frac{\partial A_{31}}{\partial r} + \frac{A_{31}}{r} \right) \mathbf{e}_z = \frac{8C}{R_c^4} \left(-1 + \frac{r^2}{R_c^2} - \frac{r^4}{8R_c^4} \right) \mathbf{e}_z \quad (8.8)$$

The L^2 -norm of Eq. (8.8) is the analytical solution expected to be obtained after one time iteration with the SD method.

8.1.5.3. Subgrid-scale velocity calculation with the SD method and validation

The computation of $\|\nabla^2(\nabla \times \mathbf{u})\|_2$ at SP is done as follows:

1. The gradient of the velocity vector with respect to physical coordinates, noted $\nabla_{\mathbf{x}}\mathbf{u}$, is known at all FP along ξ , η and ζ directions either from the original or the SDLIFT formulation.
2. From $\nabla_{\mathbf{x}}\mathbf{u}$, the rotational of \mathbf{u} with respect to physical coordinates is obtained at all these FP through Eq. (8.9):

$$\nabla_{\mathbf{x}} \times \mathbf{u} = \begin{pmatrix} \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z} \\ \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \end{pmatrix} \equiv \boldsymbol{\omega} \quad (8.9)$$

where $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^T$ is the notation used here for the rotational vector of \mathbf{u} .

3. As for the conservative or primitive variables, the gradient with respect to physical coordinates of $\boldsymbol{\omega}$ is computed at all FP along ξ , η and ζ directions either using the original or the SDLIFT formulation. It ends up with $\nabla_{\mathbf{x}}\boldsymbol{\omega}$ which is a 3×3 matrix stored at each FP along ξ , η and ζ directions.
4. Then, Eq. (4.71) is employed to compute $\nabla_{\mathbf{x}} \cdot (\nabla_{\mathbf{x}}\boldsymbol{\omega})$ at SP since:

$$\nabla_{\mathbf{x}} \cdot (\nabla_{\mathbf{x}}\boldsymbol{\omega}) = \frac{1}{|J|} \nabla_{\xi} \cdot (|J| J^{-1} \nabla_{\mathbf{x}}\boldsymbol{\omega}) \quad (8.10)$$

5. Finally, the L^2 -norm of Eq. (8.10) is computed to get $\|\nabla^2(\nabla \times \mathbf{u})\|_2$ at SP.

Because in Eq. (3.53) it is actually $\Delta_x^3 \|\nabla^2(\nabla \times \mathbf{u})\|_2$ that is needed, the division by the Jacobian at SP in Eq. (8.10) is not done in practice since $\Delta_x^3 = |J|$. It was only the case for computing $\|\nabla^2(\nabla \times \mathbf{u})\|_2$ here with JAGUAR following the five steps detailed above.

Results after one time iteration, for the discretization described in paragraph 8.1.5.1, are shown in Figure 8.8. A perfect agreement between JAGUAR solution and the L^2 -norm of Eq. (8.8) can be seen. Therefore, the SD discretization computes very well the operator $\|\nabla^2(\nabla \times \mathbf{u})\|_2$ and thus also u'_Δ .

8.2. Positivity-preserving limiter

In order to ensure the positivity of mass fractions, pressure and temperature, a limiting procedure, firstly introduced in the DG context for compressible Euler equations by Zhang and Shu [91] and extended in the SD framework for shock-capturing [90] and multi-species Euler equations [89], is employed. Zhang and Shu [91] demonstrated that this limiting procedure is conservative if the reconstructed variables after the limitation procedure are the conservatives variables. Therefore the same method is used here. The limiter was not necessary for the laminar cases presented in Chapter 7 but for the turbulent cases considered in this chapter, its use greatly improves the numerical stability.

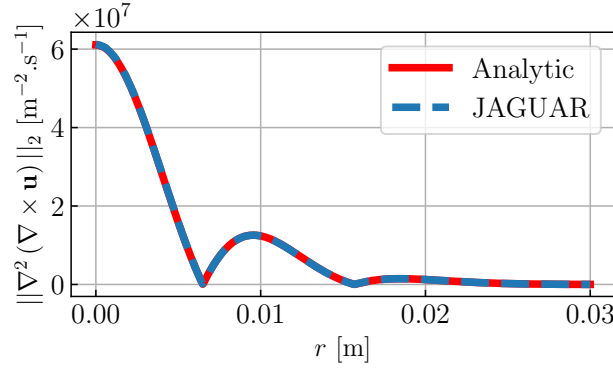


Figure 8.8. – Comparison between $\nabla_{\mathbf{x}}\mathbf{u}$ obtained with the methodology of paragraph 8.1.5.3 and the L^2 -norm of Eq. (8.8).

Its implementation slightly differs from the current literature: this point is discussed at the end of this section. Without loss of generality, the method is described for a given element Ω_e which was flagged for limiting process. Firstly, the spatial average of conservative variables over SP in Ω_e noted $\bar{\mathbf{U}}^e$ is computed and $\rho_k^e(\mathbf{x})$ is replaced by:

$$\hat{\rho}_k^e(\mathbf{x}) = \theta_k (\rho_k^e(\mathbf{x}) - \bar{\rho}_k^e) + \bar{\rho}_k^e \text{ for } k = 1, N_s \quad (8.11)$$

with:

$$\theta_k = \min \left(\frac{\bar{\rho}_k^e}{\bar{\rho}_k^e - \rho_{k,min}^e}, 1 \right) \text{ for } k = 1, N_s \quad (8.12)$$

where $\rho_{k,min}^e = \min_{\alpha} (\rho_k)$ for $k = 1, N_s$ and α is the set of SP and FP in Ω_e . This operation guarantees the positivity of mass fractions. Density at each SP is then recomputed using Eq. (2.92) to get $\hat{\rho}^e(\mathbf{x})$ ensuring $Y_k \leq 1$ for $k = 1, N_s$. The following vector in Ω_e is now defined:

$$\hat{\mathbf{U}}^e(\mathbf{x}) \equiv \begin{pmatrix} \hat{U}_1^e(\mathbf{x}) \\ \hat{U}_2^e(\mathbf{x}) \\ \hat{U}_3^e(\mathbf{x}) \\ \hat{U}_4^e(\mathbf{x}) \\ \hat{U}_5^e(\mathbf{x}) \\ \hat{U}_{5+k}^e(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \hat{\rho}^e(\mathbf{x}) \\ (\rho u)^e(\mathbf{x}) \\ (\rho v)^e(\mathbf{x}) \\ (\rho w)^e(\mathbf{x}) \\ (\rho E)^e(\mathbf{x}) \\ \hat{\rho}_k^e(\mathbf{x}) \end{pmatrix} \quad (8.13)$$

In order to always keep the pressure above a user-defined minimum value ϵ_P , at each α where $P(\hat{\mathbf{U}}^{e,\alpha}) < \epsilon_P$, the following equation for a variable $t_\epsilon^\alpha \in [0, 1]$ is solved [91]:

$$P \left[(1 - t_\epsilon^\alpha) \bar{\mathbf{U}}^e + t_\epsilon^\alpha \hat{\mathbf{U}}^{e,\alpha} \right] = \epsilon_P \iff P \left[\bar{\mathbf{U}}^e + t_\epsilon^\alpha (\hat{\mathbf{U}}^{e,\alpha} - \bar{\mathbf{U}}^e) \right] = \epsilon_P \quad (8.14)$$

otherwise $t_\epsilon^\alpha = 1$ if $P(\hat{\mathbf{U}}^{e,\alpha}) \geq \epsilon_P$. This actually avoids the pressure at any point α to go below ϵ_P . Eq. (8.14) is a second-order algebraic equation in t_ϵ^α under the approximation that sensible enthalpies are computed at a fixed temperature $T(\hat{\mathbf{U}}^{e,\alpha})$ as it is shown in paragraph A.7.1. It can be solved either analytically or using an iterative procedure such as Ridders' iterative algorithm [295] as suggested by Lodato [90]. The use of this algorithm avoids the approximation made on sensible enthalpies but there is also a risk that the iterative procedure does not converge well. In this work

the solving was done analytically. Once each t_ϵ^α is found, the corrected conservative variables at SP in the flagged element can be computed:

$$\tilde{\mathbf{U}}^e(\mathbf{x}) = \theta_2 \left(\hat{\mathbf{U}}^e(\mathbf{x}) - \overline{\mathbf{U}}^e \right) + \overline{\mathbf{U}}^e \quad (8.15)$$

with $\theta_2 = \min_\alpha(t_\epsilon^\alpha)$. In some cases, it was also found that limiting the temperature field obtained with $\tilde{\mathbf{U}}^e(\mathbf{x})$ also offers a gain in stability. The equation is very similar to the pressure equation since $P = \rho R_{gas} T$:

$$\rho R_{gas} T \left[\overline{\mathbf{U}}^e + t_\epsilon^\alpha \left(\tilde{\mathbf{U}}^{e,\alpha} - \overline{\mathbf{U}}^e \right) \right] = \rho R_{gas} \epsilon_T \quad (8.16)$$

and is solved at each α where $T(\tilde{\mathbf{U}}^{e,\alpha}) < \epsilon_T$ avoiding temperature to go below ϵ_T . Eq. (8.16) is also a second-order algebraic equation in t_ϵ^α without approximation this time since the temperature at which the h_{sk} , used in $\hat{h}_s^{e,\alpha}$ (defined in paragraph A.7.1), are computed is fixed to ϵ_T . This equation is derived in paragraph A.7.2. Then, new corrected conservative variables at SP in the flagged element are obtained:

$$\tilde{\tilde{\mathbf{U}}}^e(\mathbf{x}) = \theta_3 \left(\tilde{\mathbf{U}}^e(\mathbf{x}) - \overline{\mathbf{U}}^e \right) + \overline{\mathbf{U}}^e \quad (8.17)$$

where $\theta_3 = \min_\alpha(t_\epsilon^\alpha)$. This methodology is applied at each RK stage and has the following differences from what is mentioned in the literature [89, 90] in the SD context:

- There is no limiting procedure applied at FP to modify conservative variables at these points. The values at FP are only used to spot minimum values of ρ_k , P and T in an element and check if they are respectively above 0, ϵ_P and ϵ_T in order to compute θ_k , θ_2 and θ_3 . Only conservative variables at SP are modified using these values of θ_k , θ_2 and θ_3 .
- The numerator in Eq. (8.12) is not set to $\bar{\rho}_k^e - \epsilon_Y$, with $\epsilon_Y \approx 10^{-13}$, since it is possible to have $Y_k = 0$ for some species and then $\bar{\rho}_k^e$ is really zero.
- A limitation procedure for temperature was needed additionally to the classical one applied for pressure. It might come from the non-linear character of Eq. (8.14) because of the sensible enthalpies needed in the multi-species case. It is not something that was taken into account in Zhang and Shu paper [91] who worked with calorically perfect gases and no mention of this issue was reported in Tofaili *et al.* [89] even if they were using multi-species thermally perfect gases.

8.3. Simulation of the Cambridge flame burner

8.3.1. Presentation of the case

The Cambridge burner, firstly studied by Sweeney *et al.* [120], is an academic configuration which is composed of two co-axial tubes surrounding a central bluff-body as shown on Figure 8.9 where burner dimensions and injected streams are illustrated. Inner (indexed i) and outer (indexed o) inlet streams contain CH_4/Air mixtures and are controlled independently in terms of bulk velocity $U_{b,i/o}$ and equivalence ratio $\phi_{i/o}$. The last stream is a co-flow (indexed cf) of air ($\phi_{cf} = 0$) at $U_{b,cf} = 0.4 \text{ m.s}^{-1}$ used to isolate the flame from ambient perturbations. Finally, all injected streams are at $T_0 = 298 \text{ K}$. This burner has been widely studied experimentally with different levels of CH_4/Air stratification (variations of ϕ_i and ϕ_o) and different values of swirl providing an abundant experimental database of fifteen different cases denoted from SwB1 to SwB15 [119, 120, 296–298]. Consequently, it has been also much investigated numerically with LES [182, 290, 299, 300] and DNS [301, 302]. The numerical simulations in this work focus on two non-swirled cases for which operating conditions are indicated in Table 8.5: a non-reacting configuration SwBc and a premixed ($\phi_i = \phi_o$) reactive configuration SwB1. The bulk Reynolds numbers of the inner and outer streams are $Re_{b,i} \approx 5960$ and $Re_{b,o} \approx 11500$ respectively [298], i.e., lead to a turbulent flame.

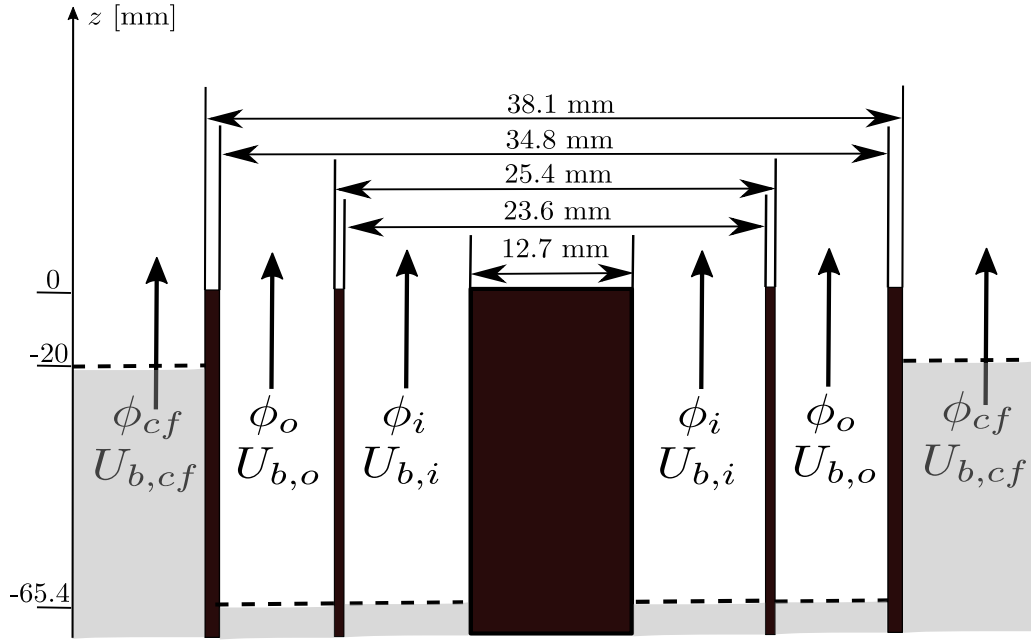


Figure 8.9. – Sketch of the Cambridge burner with associated dimensions and injected streams. The white area corresponds to the computational domain.

| Case | ϕ_i [-] | ϕ_o [-] | ϕ_{cf} [-] | $U_{b,i}$ [m.s ⁻¹] | $U_{b,o}$ [m.s ⁻¹] | $U_{b,cf}$ [m.s ⁻¹] |
|------|--------------|--------------|-----------------|--------------------------------|--------------------------------|---------------------------------|
| SwBc | 0 | 0 | 0 | 8.31 | 18.7 | 0.4 |
| SwB1 | 0.75 | 0.75 | 0 | 8.31 | 18.7 | 0.4 |

Table 8.5. – Operating conditions considered in this work for the simulations of the Cambridge burner.

8.3.2. Computational domain

The computational domain has the following characteristics:

- The injector streams start at $z = -65.4$ mm as illustrated on Figure 8.9 and are of length $L_{inj} = 6D_{bb}$, where $D_{bb} = 12.7$ mm is the bluff-body diameter, ending up at $z = 0$. This length was shown sufficient to have an established turbulent flow inside the injectors.
- The co-flow stream starts at $z = -20$ mm.
- The domain is extended 250 mm downstream of the injector exits in the z -direction so that the length of the chamber is $L_{ch} = 250$ mm. The radius of the chamber is set to $R_{ch} = 140$ mm.

8.3.3. Boundary conditions

Inlet and outlet boundary conditions are imposed through the NSCBC methodology described in paragraph 5.1.3. For the inlet of the injection tubes at $z = -65.4$ mm, an analytic power-law profile is imposed for the mean axial velocity of the form [286]:

$$\bar{w}_{i/o}(r_{i/o}) = U_{cl,i/o} \left[1 - \left(\frac{2r_{i/o} - (r_{1,i/o} + r_{2,i/o})}{r_{2,i/o} - r_{1,i/o}} \right)^4 \right], \quad r \in [r_{1,i/o}, r_{2,i/o}] \quad (8.18)$$

where $r_{1,i/o}$ and $r_{2,i/o}$ are the radius of respectively the inner and outer walls of the injected stream and $U_{cl,i/o}$ is the centerline velocity. Values of $U_{cl,i/o}$ are determined to recover the expected bulk

velocity in each injector, following Eq. (A.57) of Section A.8:

$$U_{cl,i} = 10.38 \text{ m.s}^{-1} \text{ and } U_{cl,o} = 23.37 \text{ m.s}^{-1} \quad (8.19)$$

Homogeneous isotropic turbulence is superimposed to these mean velocity profiles using the synthetic random Fourier method detailed in paragraph 6.4.2 with a VKP energy spectrum using $N = 300$ logarithmically distributed turbulent modes generated with $u_{rms,i/o} = 0.1U_{b,i/o}$ and an integral length scale $L_e = 1 \text{ mm}$ for both injector streams. These parameters result from a sensitivity analysis performed on a reduced domain to find the values that best fit measurements directly downstream of injector streams at $z = 2 \text{ mm}$. Mass fractions are set in the injection streams so as to obtain the equivalence ratios defined in Table 8.5, i.e., for the SwBc case:

$$Y_{O_2,i/o} = 0.2226, Y_{N_2,i/o} = 0.7774 \text{ and } Y_{k,i/o} = 0 \text{ for the remaining species} \quad (8.20)$$

and for the SwB1 case:

$$Y_{CH_4,i/o} = 0.042, Y_{O_2,i/o} = 0.2232, Y_{N_2,i/o} = 0.7348 \text{ and } Y_{k,i/o} = 0 \text{ for the remaining species} \quad (8.21)$$

For the inlet co-flow at $z = -20 \text{ mm}$, a uniform axial flow at $U_{b,cf}$ is imposed with species mass fractions set to $Y_{O_2,cf} = 0.2226$ and $Y_{N_2,cf} = 0.7774$. Outlet boundary conditions imposing $P = 101325 \text{ Pa}$ are prescribed on the sides of the chamber and at outlet. Finally, all walls are treated with the adiabatic no-slip boundary condition imposed following the approach explained in paragraph 5.2.3.2.

8.3.4. Mesh discretization

In order to study the impact of the scheme order on the results, two different discretizations were considered with both a characteristic distance between two DOF of $\Delta_x = 0.3 \text{ mm}$ in the flame region downstream of the injected streams. The first mesh is built to have $p = 2$ (order 3) in all elements and is composed of $N_e = 378424$ second-order hexahedral elements ending up with approximately 10.2 millions of DOF. The second mesh is built to have $p = 3$ (order 4) in all elements and is composed of $N_e = 205070$ second-order hexahedral elements ending up with approximately 13.1 millions of DOF.

8.3.5. Chemistry and combustion model

The chemical scheme that is employed is the two-reactions CH₄/Air-2S-BFER [130]. Previous numerical studies have employed more detailed mechanisms such as the one of Lindstedt [303] used in Mercier *et al.* [182, 304] or the GRI-3.0 scheme [305] considered in Proch and Kempf [300] and Proch *et al.* [301, 302]. In these cases, detailed mechanisms were affordable thanks to a tabulated chemistry approach where only the mixture fraction Z and the progress variable C are computed in the LES or DNS. However, it has been demonstrated [130] that the CH₄/Air-2S-BFER mechanism is able to reproduce the correct evolution of velocity, temperature, laminar flame speed and major species which are the measured quantities used for validation. The TFLES combustion model introduced in Section 8.1 with the relaxation flame sensor and the saturated Charlette's model at a constant β is considered. The maximum thickening is determined so as to have at least seven points in the flame front, and is found around 4.4 since $\Delta_x = 0.3 \text{ mm}$ and $\delta_L^0 = 0.48 \text{ mm}$ for the CH₄/Air-2S-BFER scheme at $\phi = \phi_i = \phi_o = 0.75$.

8.3.6. Other numerical parameters

For the non-reacting case SwBc, the average approach of Sun *et al.* [47] for the diffusion scheme is employed since it is faster than the SDLIFT formulation and gives enough satisfying results. However,

concerning the premixed reacting case SwB1, the SDLIFT formulation is needed to stabilize the numerical solution in shear layers compared to the average formulation which showed high numerical instabilities as depicted in Section 7.5. The SDLIFT formulation was employed with the IC scheme coupled with g_{DG} correction function in this case. Moreover, the use of the positivity-preserving limiter introduced in Section 8.2 also helped to avoid undesired numerical behavior for the reacting case. Finally, for both non-reacting and reacting simulations, time integration was carried out by the SSP-ERK(5,4) scheme of Spiteri and Ruuth [226], convective fluxes at interfaces were handled by the HLLC scheme [201] and no turbulent subgrid-scale model was employed, as explained in paragraph 3.2.2.4.

8.3.7. Post-treatment

All simulations were run first for 10 convective times to evacuate transient phenomena. Based on L_{inj} and $U_{b,i}$ this corresponds to 79 ms. Then, simulations were restarted for another 10 convective times to collect time-average statistics. Because the configuration is axisymmetric, azimuthal averaging was finally applied to further enrich the statistics in 2D (r, z) maps.

8.3.8. Results analysis on case SwBc

Simulations of the SwBc case have been performed on the two discretizations mentioned in paragraph 8.3.4. Radial profiles of the statistics of axial and radial velocities at different downstream locations from the burner exit are illustrated in Figure 8.10 for the cases $p = 2$ and $p = 3$. They are compared with Laser Doppler Anemometry (LDA) data of Zhou *et al.* [298] and numerical results from the LES of Mercier [286] on their finer mesh with $\Delta_x = 0.25$ mm in the flame region, very close to the value of 0.3 mm employed here. This LES was carried out using the YALES2 low-Mach number and unstructured finite volume flow solver [306]. A general good agreement is observed between JAGUAR results and experimental data and also with other numerical results. The differences between simulations and measurements are almost the same for both JAGUAR and YALES2, only the radial velocity RMS is better predicted by JAGUAR. Comparing the two JAGUAR discretizations, it appears that mean velocities are not much impacted by p , whereas RMS velocities especially the radial one are better predicted with $p = 3$. This demonstrates that for a given mesh, a higher polynomial degree improves the quality of the results. Nevertheless, both discretizations tested with JAGUAR are sufficient to capture the statistics of the aerodynamic fields. Finally, as explained in paragraph 3.2.2.4, it is possible to check if discretizations employed were sufficient to capture approximately 80% of the TKE without using a subgrid-scale model. Figure 8.11 shows the evolution of Pope's criterion (see Eq. (3.26)), computed with the Vreman model [157] using $C_{SGS} = 0.10$, on the $p = 2$ discretization in the burner mid-section. It should be mentioned that this criterion was computed on the fly at SP, only for post-processing purposes, without applying the Vreman model at FP for flux computation. According to Figure 8.11, since Pope's criterion requires $\mathcal{M} < 0.2$, the $p = 2$ discretization is sufficient for a good LES computation without the necessity of adding a SGS turbulent model.

8.3.9. Results analysis on case SwB1

The premixed reacting case SwB1 is now considered using the two discretizations described in paragraph 8.3.4. Simulations are initialized from a first run with only cold gases which are ignited by a small sphere of burnt gases and of diameter D_{bb} introduced at the bluff-body level. Preliminary tests have shown that $\beta = 0.4$ for the Charlette's efficiency function gives the best results for JAGUAR simulations. Contour plots in the burner mid-section of temperature, heat release rate, equivalence ratio and axial velocity are shown in Figure 8.12. The flame is anchored at the burner bluff-body because of the adiabatic conditions as already observed in Mercier [286] while experiments predict a lifted flame

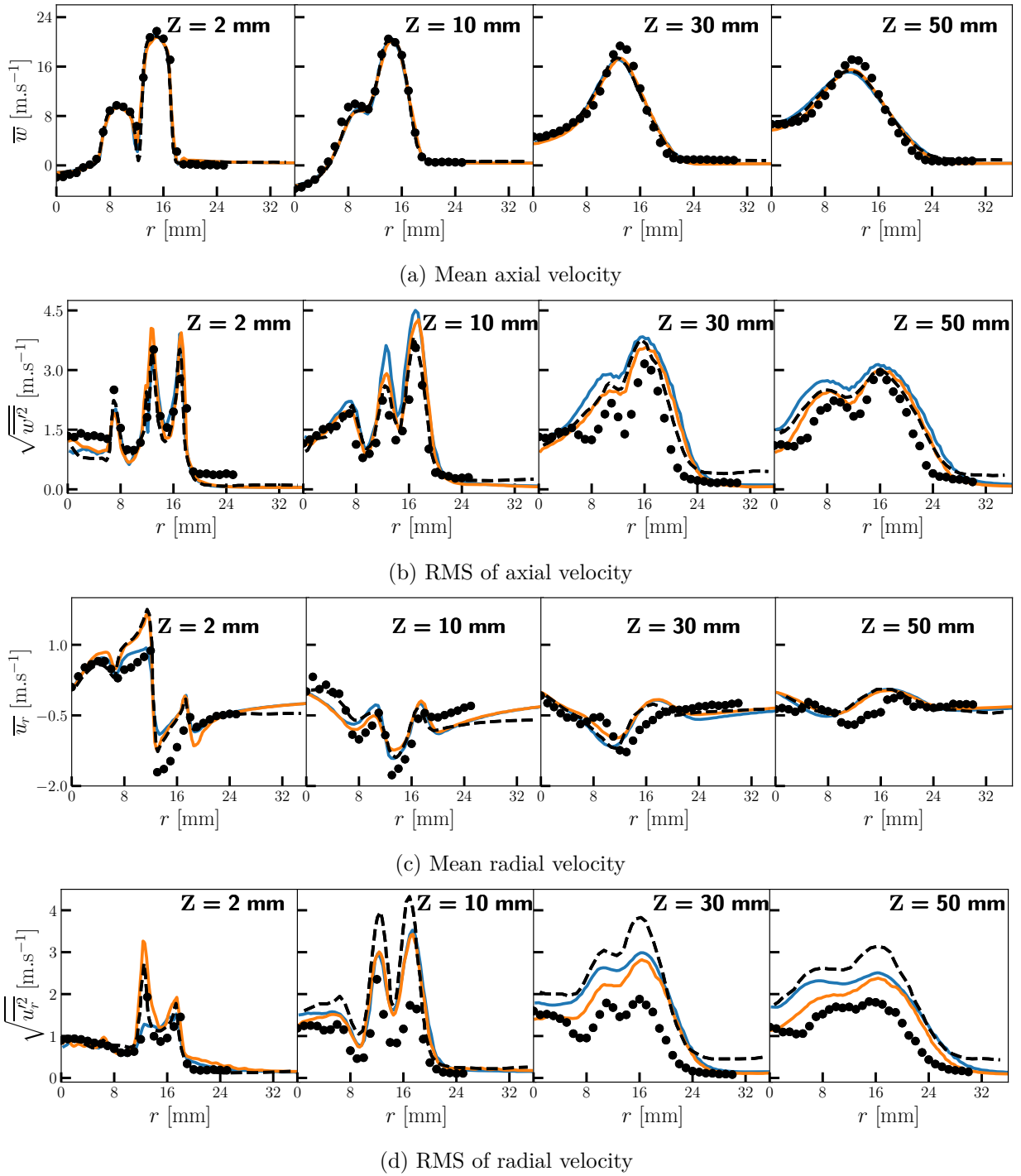


Figure 8.10. – Radial profiles of the statistics of axial and radial velocities at different distances from the burner exit for the non-reacting case SwBc. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Mercier [286] (---) and experimental data [298] (•).

due to the non-adiabaticity of the bluff-body surface. It can be seen that the velocity downstream the bluff-body is very stable without turbulent fluctuations, attributed to re-laminarisation effects of combustion, which was already highlighted in previous works on this configuration [290, 298, 301, 304].

Radial profiles of the statistics of axial and radial velocities at different downstream locations from

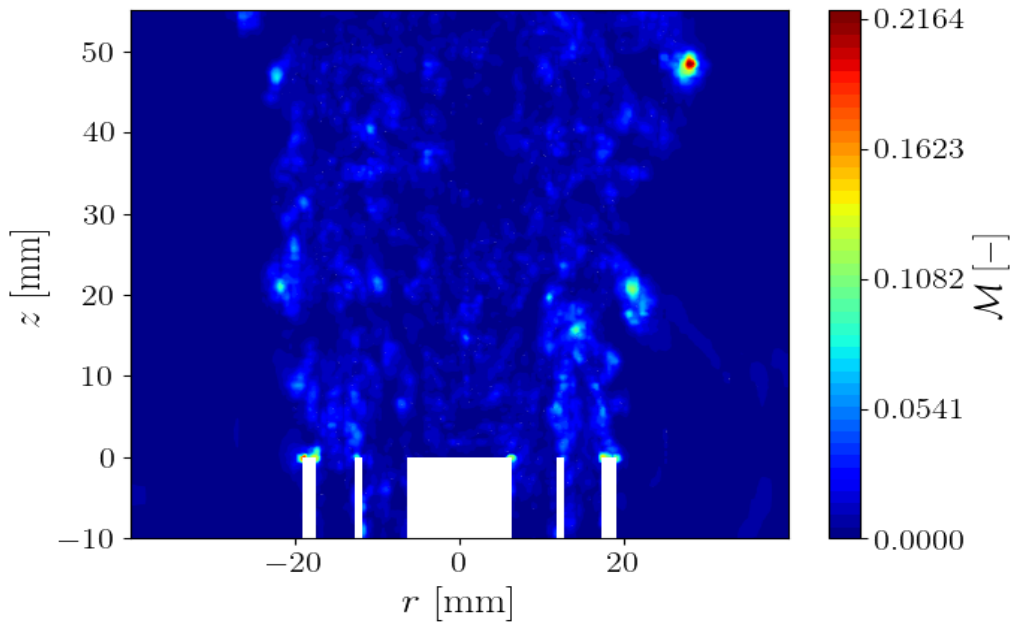


Figure 8.11. – Evolution of Pope's criterion, computed with the Vreman model [157] using $C_{SGS} = 0.10$, on the $p = 2$ discretization in the burner mid-section.

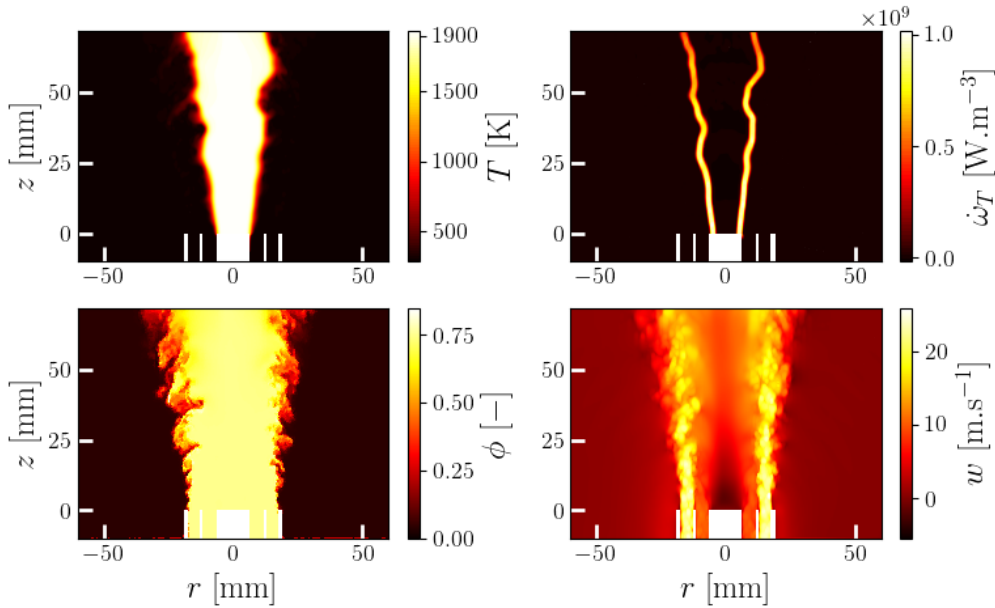


Figure 8.12. – Contour plots in the burner mid-section of instantaneous temperature T (top-left), heat release rate $\dot{\omega}_T$ (top-right), equivalence ratio ϕ (bottom-left) and axial velocity (bottom-right) obtained with JAGUAR for the $p = 3$ case.

the burner exit are illustrated in Figure 8.13 for the cases $p = 2$ and $p = 3$. They are compared with Laser Doppler Anemometry (LDA) data of Zhou *et al.* [298] and numerical results from the LES of Proch and Kempf [300] on their finer mesh with $\Delta_x = 0.25$ mm in the flame region very close to the value of 0.3 mm employed here. The LES of Proch and Kempf was conducted using their in-house FV code PsiPhi [307, 308]. As in the non-reacting case, JAGUAR results are in good agreement with both experimental and numerical data. Looking at the mean axial velocity at $z = 10$

mm, it seems that JAGUAR slightly overpredicts the recirculation zone behind the bluff-body. This overprediction was also present in Proch and Kempf results, but is more pronounced with JAGUAR. It was said in their paper [300] to originate from the numerically attached vs experimentally lifted flame due to the different wall heat flux at the bluff body and was also observed in Nambully *et al.* [290]. Finally, radial profiles of temperature and of mean equivalence ratio statistics, at the same downstream distances from the burner exit, are represented in Figure 8.14 again for $p = 2$ and $p = 3$ cases simulated with JAGUAR, along with Proch and Kempf [300] results. Experimental data come from Rayleigh-measurements provided by Sweeney *et al.* [296]. Both mean temperature and equivalence ratio obtained with JAGUAR are in good agreement with both experimental and numerical results. However, temperature fluctuations are significantly smaller than the measurements. This is attributed to the TFLES combustion model which by construction deletes part of the resolved fluctuations and recovers them with the efficiency function. This effect is less important when using the TFLES model with a premixed flamelet tabulation method as in the LES of Proch's [309]. This tendency is less visible on the $p = 3$ simulation especially in $z = 10$ mm and $z = 30$ mm. It shows that, for a given mesh size in the flame region, increasing the spatial order of the scheme improves the prediction of the wrinkling even if the thickening factor is almost identical. Consequently, the limitations of the TFLES model can be compensated by using higher order schemes advocating for the use of high-order methods in combustion.

This test case has shown that the SD method developed in this work for reacting flows, is able to simulate three dimensional turbulent flames with satisfying results close to the ones previously obtained with classical numerical methods doing LES of combustion. It also highlights the fact that coarser mesh discretizations can be employed using high-order methods for the same results: Δ_x was set to 0.3 mm in the flame region for JAGUAR with a maximum of 13.1×10^6 DOF using $p = 3$ whereas Proch and Kempf [300] had a Δ_x of 0.25 mm in this region but with more than 103.2×10^6 DOF. This high difference is due to the use of equidistant orthogonal cartesian grids in the LES of Proch and Kempf whereas in this work the mesh is refined only in the flame region. Other LES of the Cambridge burner, made with unstructured FV meshes refined only in the flame region, were also considered [182, 310] and had a number of DOF between 6×10^6 to 17×10^6 for a minimum Δ_x value of 0.25 mm. Therefore, the discretizations employed in this work are also in this range of DOF values.

8.3.10. Computational cost

The Cambridge burner is a good test case to analyze the computational cost of JAGUAR between non-reacting and reacting flow simulations on a realistic 3D turbulent flame. The discussion will be oriented on the iteration cost κ , introduced in paragraph 7.4.4, since time steps were very similar for all simulations. Table 8.6 sums up the iteration costs for SwBc and SwB1 cases for $p = 2$ and $p = 3$ discretizations. The SwBc case presented in paragraph 8.3.8 was simulated with only the

| Polynomial degree p | 2 | 3 |
|--|------|------|
| SwBc with $N_s = 0$, Average approach, no limiter | 10.1 | 8.9 |
| SwBc with $N_s = 3$, Average approach, no limiter | 17.3 | 15.6 |
| SwBc with $N_s = 3$, Average approach, with limiter | 18.1 | 15.8 |
| SwBc with $N_s = 3$, SDLIFT approach, no limiter | 29.5 | 25.9 |
| SwB1 with $N_s = 7$, SDLIFT approach, with limiter | 48.3 | 43.5 |

Table 8.6. – Iteration cost κ [$\mu\text{s.ite}^{-1}.\text{DOF}^{-1}$] for SwBc and SwB1 cases for $p = 2$ and $p = 3$ discretizations with different numerical treatments. The SwBc case with $N_s = 0$ was added to estimate the cost of adding one transport equation. All computations were done on 760 processors provided by the same cluster.

reactants CH_4 , O_2 and N_2 so that $N_s = 3$ in this case whereas the first line in Table 8.6 corre-

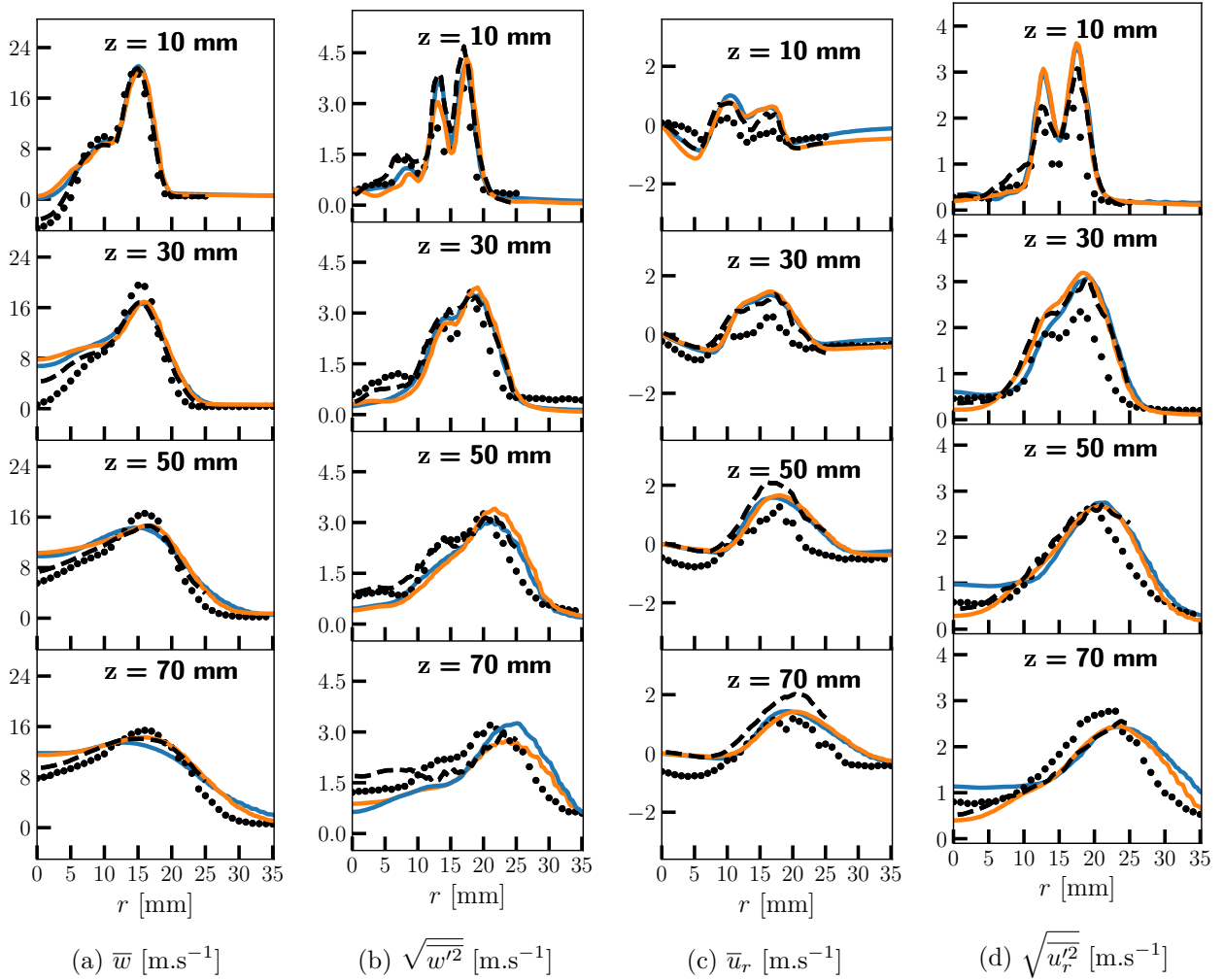


Figure 8.13. – Radial profiles of the statistics of axial and radial velocities at different distances from the burner exit. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Proch and Kempf [300] (---) and experimental data [298] (•).

sponds to the exact same case with only pure air that is $N_s = 0$. Comparing these two test cases allows to find an estimation of the iteration cost of adding one transport equation in JAGUAR which seems to be around $2.4 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for $p = 2$ and $2.2 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for $p = 3$. The SwB1 case is well more costly firstly because four more species have to be transported: the three products of combustion CO_2 , CO and H_2O and the fictive species needed for the relaxation sensor ψ . Thus, from the SwBc case at $N_s = 3$, κ is increased by a value of $9.6 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ (respectively $8.8 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$) due to the transport of additional species for $p = 2$ (respectively $p = 3$). It brings the iteration cost of SwB1 around $26.9 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ (respectively $24.4 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$) for $p = 2$ (respectively $p = 3$) case. However, in Table 8.6, the iteration cost for SwB1 case at $p = 2$ (respectively $p = 3$) is $48.3 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ (respectively $43.5 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$). Consequently, there is still $21.3 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ (respectively $19.1 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$) for $p = 2$ (respectively $p = 3$) which does not come from the transport of additional species. Actually, simulations of the SwB1 case differ from the SwBc in three main ways:

- The use of the SDLIFT formulation for the SwB1 case, instead of the average approach for the SwBc case.
- The use of the positivity-preserving limiter for the SwB1 case which is not activated for the

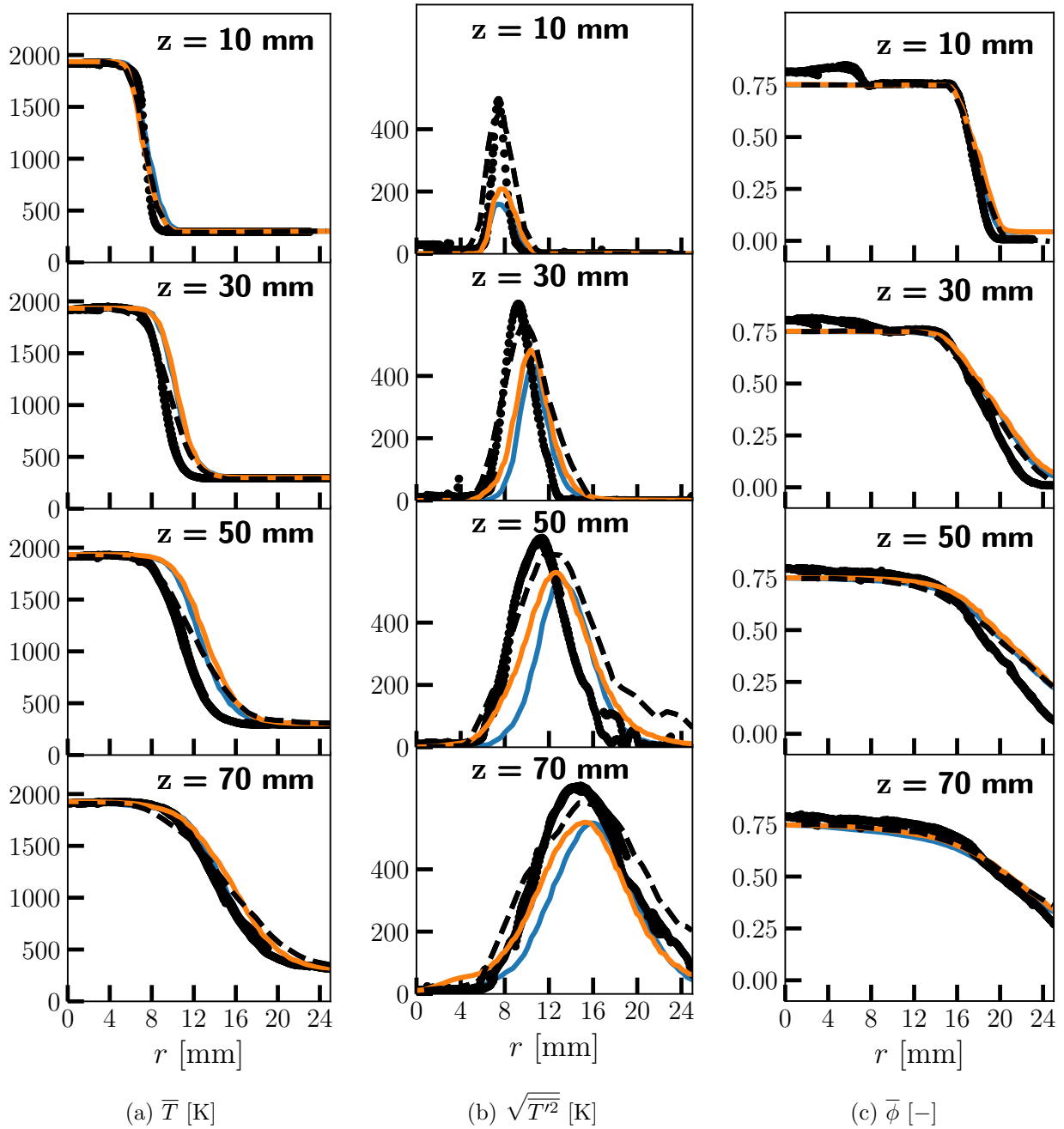


Figure 8.14. – Radial profiles of the statistics of temperature and of mean equivalence ratio at different distances from the burner exit. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Proch and Kempf [300] (—) and experimental data [296](•).

SwBc case.

- The calculation of combustion source terms only considered in the SwB1 case.

Thus, simulations of the SwBc case with $N_s = 3$ have been carried out either with the SDLIFT formulation (IC diffusion scheme with g_{DG} correction function here) or the limiter activated. Corresponding iteration costs have been reported in the fourth and fifth lines of Table 8.6 for $p = 2$ and $p = 3$ cases. According to these results it can be concluded that:

- The positivity-preserving limiter increases κ by 4.6% for $p = 2$ and by 1.3% for $p = 3$. This difference between $p = 2$ and $p = 3$ probably comes from the fact that $p = 3$ has 46% less elements

than $p = 2$ in the domain so probably less troubled elements. Nevertheless, the increase of κ due to the limiter seems not to be the main responsible for the large difference of κ between SwBc and SwB1 cases. However, these values are probably higher in the reacting case since there are surely more troubled elements where the limiting process has to be activated.

- The SDLIFT formulation increases κ by 71% for $p = 2$ and by 66% for $p = 3$ which is a lot. Again, the small difference between the two orders is probably due to the lower number of interfaces in the $p = 3$ case since it has less elements.

Therefore, the SDLIFT formulation is the main responsible parameter for the increase in iteration cost between SwBc and SwB1 cases simulated with JAGUAR. For instance in the $p = 2$ case, adding four transported species brings the iteration cost for SwB1 to $26.9 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ as stated above. Then, using only the SDLIFT formulation, this iteration cost increases by 70% ending up with $45.7 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ quite close to $48.3 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ written in Table 8.6. The remaining differences probably originate from the limiter and the calculation of combustion source terms.

Notice finally that, in all cases because DOF is kept constant, κ at $p = 3$ is approximately 10% smaller than at $p = 2$ as already highlighted in paragraph 7.4.4.

Working on optimizing this very new formulation is unavoidable for the future of JAGUAR in combustion. Indeed, the implementation of combustion equations in JAGUAR does not follow the usual ways that allow to accelerate it by Cassagne *et al.* [115] that is:

1. The use of matrix/matrix products with optimized memory placement.
2. The use of cache blocking by packing contiguous elements with same polynomial degree p with a cache value equal to the cash value of the considered machine to avoid useless copies.

With such features, JAGUAR has gained more than 30% of performance that is why the expected gain for the SDLIFT formulation is probably huge. This optimization work was not done during this thesis since it was needed to see first if the SDLIFT formulation was able to improve the results and also because the optimization process requires to review some parts of the code which will impact all JAGUAR users. Therefore, it has to be made carefully and not enough time was available during the thesis for such work.

8.3.11. Perspectives for future works on case SwB1 using local polynomial adaptation

The results on the SwB1 case have shown that increasing the polynomial degree within each element, at the same number of points in the flame front, better predicts the flame wrinkling when TFLES is employed. Indeed, reducing the thickening of the flame, and so the value of n_{pts}^{ff} , produces more wrinkling of the flame as highlighted in Figure 8.15 where temperature fluctuations are represented in the $p = 2$ discretization case for both $n_{pts}^{ff} = 7$ (original computation) and $n_{pts}^{ff} = 10$. Thus, simulations of the exact same premixed reacting case can be performed on the mesh built for $p = 2$ but using local polynomial adaptation in order to have $p = 4$ in the flame region, instead of $p = 2$, to reduce the thickening of the flame and therefore the impact of the TFLES model. The key point is how to set the polynomial distribution within each element and especially on which criteria. In paragraph 7.4.5, the use of the time-averaged of the norm of the density gradient, as a sensor, was employed on a 2D premixed laminar burner showing a good potential to locate where to refine the flame region. However, other sensors based on the HRR (similar to the one used in TFLES) or temperature gradient can be also considered. These three sensors have been computed on simulations of the Cambridge burner with $p = 2$ inside each element following the methodology described in paragraph 4.5.1. Their corresponding polynomial distribution for $p_{min} = 2$ and $p_{max} = 4$ are illustrated in Figure 8.16 along with the total number of DOF they have in the computational domain.

All three sensors capture well the flame region. The sensors based on the HRR and $\|\nabla\rho\|_2$ having less DOF compared to the $\|\nabla T\|_2$ sensor may be preferred. Recalling that the original discretization with $p = 2$ everywhere has 10.2M DOF, polynomial distributions based on HRR and $\|\nabla\rho\|_2$ add

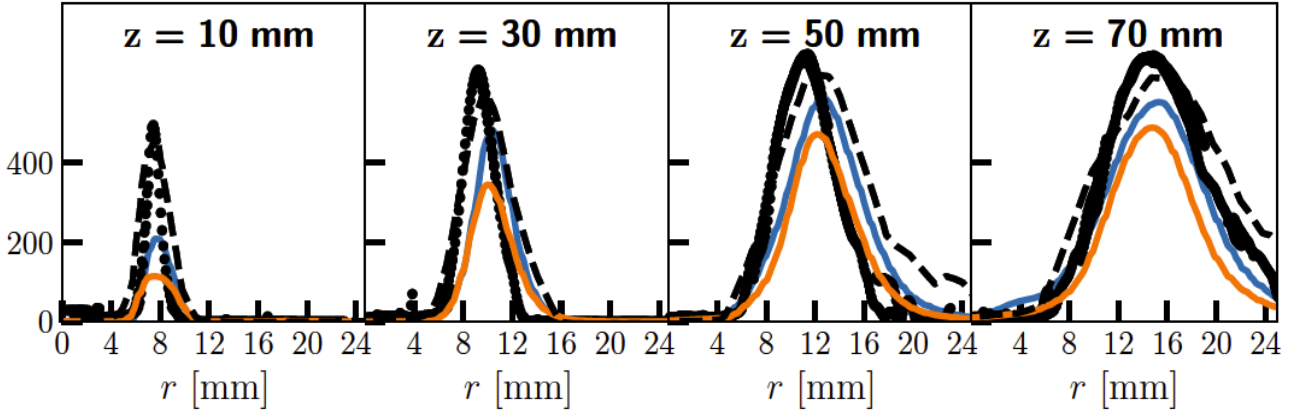


Figure 8.15. – Comparison of temperature fluctuations $\sqrt{T'^2}$ at different distances from the burner exit in the $p = 2$ discretization case for both $n_{pts}^{ff} = 7$ (original computation) and $n_{pts}^{ff} = 10$. Legends: JAGUAR $n_{pts}^{ff} = 7$ (—), JAGUAR $n_{pts}^{ff} = 10$ (—), numerical results of Proch and Kempf [300] (---) and experimental data [296] (•).

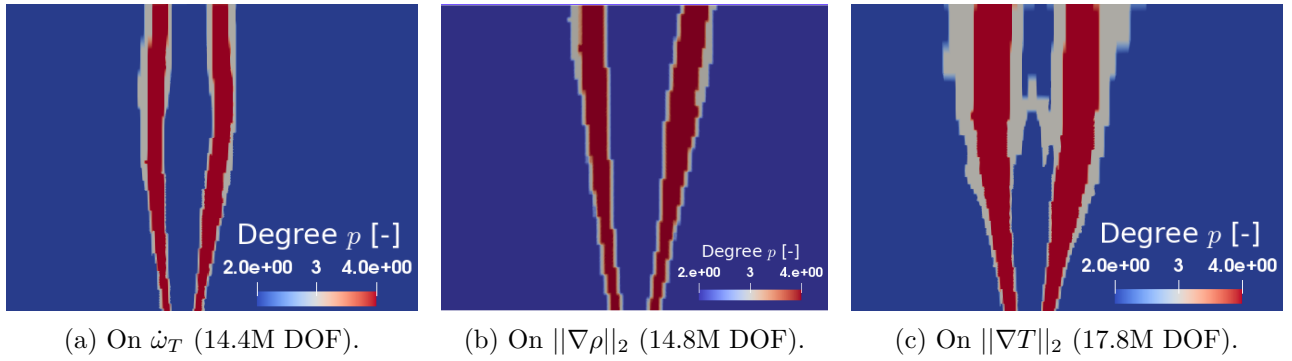


Figure 8.16. – Comparison of polynomial distributions obtained with different sensors on the Cambridge burner when $p_{min} = 2$ and $p_{max} = 4$. The computation of sensor values is done using a first run at constant $p = 2$.

around 50% DOF but Δ_x in the flame region is reduced by a factor of 3/5 ending up to a value of 0.18 mm. The new maximum thickening therefore goes from 6 to 3.6 which leads to a more wrinkled thickened flame, better catching experiment results in Figure 8.14b. This is probably the best way for simulating turbulent combustion cases with the SD method at a correct computational cost as it was already concluded at the end of Section 7.4:

1. Start with a mesh built for a low polynomial degree p .
2. Run a first simulation with this low p value in all elements for computing sensor values for each element.
3. Start a new simulation with a varying polynomial degree per element set thanks to the already computed sensor on the first simulation.

In the original guidelines of this work, it was not expected to do local polynomial adaptation on reacting cases since a lot of work was first needed for doing combustion cases at constant p values. However, it appeared that the use of local polynomial adaptation was essential for keeping the method competitive in terms of computational cost. Consequently, some combustion cases were simulated using p -adaptation in this work to show the potential of the method but there was not enough time for simulating the three discretizations presented on Figure 8.16.

8.4. Simulation of the VOLVO configuration

These simulations were made in collaboration with a sixth-months intern, Arthur Berthelot.

8.4.1. Presentation of the case

The VOLVO burner is a configuration firstly introduced experimentally by Sjunnesson *et al.* [121, 122]. It is composed of a rectangular chamber of constant cross section $0.12 \text{ m} \times 0.24 \text{ m}$ in the yz -plane with a bluff-body equilateral triangle flame holder of edge length $D_{bb} = 0.04 \text{ m}$ located at $x = x_{bb} = 0.82 \text{ m}$ downstream of the inlet as shown in Figure 8.17. The total length of the configuration is $L_x =$

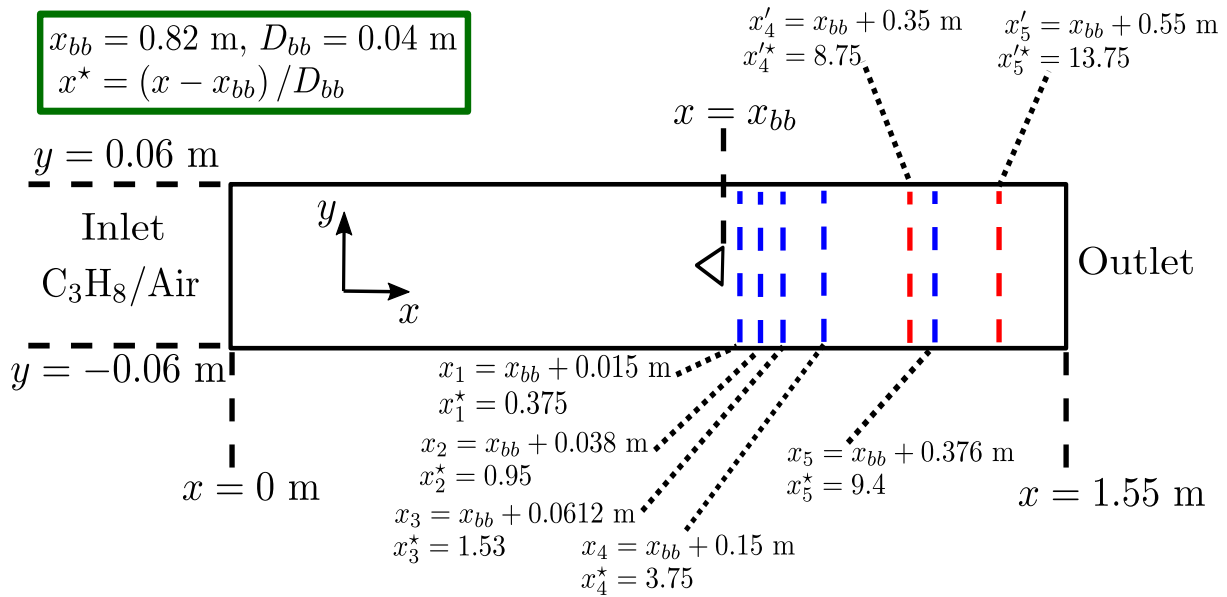


Figure 8.17. – Sketch of the VOLVO burner in the xy -plane with associated dimensions and locations of experimental data in both x and x^* spaces.

1.55 m. Three experimental elements are commonly not considered in numerical simulations [311, 312]: the fuel feeding line, the seeding and the honeycomb used to generate a turbulence level of 3% of the inlet bulk velocity [121]. In numerical simulations, premixed air and propane (C_3H_8) are injected at $T_{in} = 288 \text{ K}$ either at $U_b = 16.6 \text{ m.s}^{-1}$ or $U_b = 17.3 \text{ m.s}^{-1}$ respectively for the non-reacting and reacting cases. The amount of C_3H_8 and air injected at the inlet is determined to have an inlet equivalence ratio set to $\phi_{in} = 0.65$ in this work corresponding to a stable operating point without thermoacoustic instabilities [311]. This configuration is too large for DNS and multiple LES were carried out using several numerical methods such as FV [311–314], DG [26] and very recently LBM [111]. Although the non-reacting flow is similarly predicted by all methods [315], the results obtained in the reacting case differ significantly between the solvers. The reasons for that are quite complex but it appears that employing more accurate chemical schemes with fully dynamic turbulent combustion models and HO numerical schemes improve the results [312]. Consequently, this configuration is an excellent validation test case for the SD algorithm. In this work, both the non-reacting and the reacting cases, in the stable operating point $\phi_{in} = 0.65$, are simulated with SD and compared to:

- experimental data [121, 122]: mean axial velocity \bar{u} , y -profiles of several variables at $x_1, x_2, x_3, x_4, x_5, x'_4$, and x'_5 (see Figure 8.17) and along the chamber centerline ($y = 0$) from x_{bb} .
- LES results of several numerical methods [26, 111, 311] for the reacting case only, at the same locations as experimental data, which have the closer modeling in terms of chemical scheme and turbulent combustion models to have a fair comparison.

8.4.2. Mesh discretization

For the non-reacting simulations, discretizations are built to have a characteristic distance between two DOF in the xy -plane of $\Delta_x = \Delta_y = 0.5$ mm and $\Delta_z = 2$ mm in the spanwise direction. Three different discretizations respectively at $p = 2$, $p = 3$ and $p = 4$ are used. They end up with a number of DOF between 15 and 16 millions within the computational domain as shown in Table 8.7 where values of N_e , p and DOF are given for each case. Concerning the reacting case, a mesh where the

| p | 2 | 3 | 4 |
|-----------------------|--------|--------|--------|
| N_e | 597800 | 237540 | 120624 |
| DOF [$\times 10^6$] | 16.1 | 15.2 | 15.1 |

Table 8.7. – Values of N_e , p and DOF for the three different discretizations used for the non-reacting case of the VOLVO burner at $\phi_{in} = 0.65$.

polynomial distribution was set by hand, represented in Figure 8.27 in the bluff-body region, was employed. It is composed of three different zones:

- One zone with $x \in [0.77 \text{ m}, 1.0 \text{ m}]$ where $p = 4$ in order to have $\Delta_x = \Delta_y = 0.6$ mm in this region. It is also refined in terms of mesh elements as depicted in Figure 8.26.
- One zone with $x \in [1.0 \text{ m}, 1.1 \text{ m}]$ where $p = 3$ and a mesh element size equal to twice the one in the bluff-body refined zone.
- The remaining of the mesh, that is $x < 0.77 \text{ m}$ and $x > 1.1 \text{ m}$, is at $p = 2$ and is also made of bigger elements.

Finally, 80 1D uniform segments are used to discretize the spanwise direction giving a characteristic distance between two DOF in this direction for the bluff-body region of $\Delta_z = 0.6$ mm. Thus, the discretization is isotropic in the bluff-body region for the reacting case and the total number of DOF in the domain is around 24.5 millions. More details about this discretization for the reacting case are given in paragraph 8.4.8.1

8.4.3. Boundary conditions

Inlet and outlet boundary conditions are imposed through the NSCBC methodology described in paragraph 5.1.3. For the inlet, the axial velocity profile injected is given by Eq. (6.38) with $n = 7$, $h = 0.06$ m and $U_{cl} = 18.97 \text{ m.s}^{-1}$ (respectively $U_{cl} = 19.77 \text{ m.s}^{-1}$) for the non-reacting (respectively reacting) case obtained from Eq. (6.39) to have the expected bulk velocity. As opposed to what was done in previous studies [111, 311, 312], no turbulence injection appeared to be needed in this work. Species mass fractions at the inlet are specified to have $\phi_{in} = 0.65$:

$$Y_{C_3H_8,in} = 0.04, Y_{N_2,in} = 0.7363, Y_{O_2,in} = 0.2237 \quad (8.22)$$

At the outlet, static pressure is imposed at 101325 Pa and finally the bluff-body and all chamber sides are modeled as adiabatic no-slip walls using the approach detailed in paragraph 5.2.3.2.

8.4.4. Chemistry and combustion model

The chemical scheme considered for all reactive simulations is the two-reactions C_3H_8 /Air mechanism developed in [311] especially for the VOLVO configuration. The TFLES combustion model introduced in Section 8.1 with the relaxation flame sensor and the saturated Charlette's model with a constant β set to 0.5 is employed. The maximum thickening is computed to have at least 7 points in the flame front which is around 7 in the bluff-body region since $\delta_L^0 = 0.61$ mm for the C_3H_8 /Air mechanism at $\phi_{in} = 0.65$.

8.4.5. Other numerical parameters

As for the Cambridge burner, the non-reacting case is simulated using the average approach of Sun *et al.* [47] for the diffusion scheme whereas the reacting case employs the SDLIFT formulation using IC scheme coupled with g_{DG} correction function for the diffusion scheme. The positivity-preserving limiter introduced in Section 8.2 is also activated in the reacting case to stabilize the numerical solution. Finally, for both non-reacting and reacting simulations, time integration was carried out by the SSP-ERK(5,4) scheme of Spiteri and Ruuth [226], convective fluxes at interfaces were handled by the HLLC scheme [201] and no turbulent subgrid scale model was employed.

8.4.6. Post-treatment

For non-reacting simulations, all calculations were run first for 1 convective time, based on L_x and $U_b = 16.6 \text{ m.s}^{-1}$ which corresponds to 93 ms, to evacuate the transient flow. Then, these simulations were restarted for another 3 convective times to collect time-average statistics which were finally spatially averaged in the z -direction. Concerning the reacting case, simulations were run for 0.5 convective times based on L_x and $U_b = 17.3 \text{ m.s}^{-1}$ and then statistics were collected during one convective time only due to the cost of this simulation. However, Rochette *et al.* [312] reported convergence for the reacting case after around 52 ms of physical time, which is half of the present averaging time.

8.4.7. Results analysis on the non-reacting case

Figure 8.18 presents the vortex shedding structures obtained in the simulations, visualized with an iso-contour of Q -criterion colored by v . A von Karman-like flow, already observed in previous studies [111, 311, 315], can be seen. In Figures 8.19-8.24, JAGUAR results for $p = 2$, $p = 3$ and $p = 4$ are compared against experimental data for the mean axial velocity along the centerline $y = 0$ in x^* -space defined on Figure 8.17, and y -profiles of mean and RMS velocities at five different axial locations x_1 - x_5 also represented on Figure 8.17. The recirculation zone behind the bluff-body is very well captured for any value of p as shown on Figure 8.19. Moreover, y -profiles of the different velocities are also in very good agreement with experimental data. Finally, as for the Cambridge flame burner, Pope's criterion (see Eq. (3.26)) can be plotted to see if VOLVO discretizations are sufficient to have most of the TKE spectrum. It is illustrated on Figure 8.25 for the $p = 3$ case, near the bluff-body region, where the Vreman model with $C_{SGS} = 0.10$ was used to estimate the SGS TKE at SP. The criterion $\mathcal{M} < 0.2$ is respected almost everywhere in the flow region and especially in the region of interest downstream of the bluff-body. Thus, similarly to the Cambridge flame burner, the discretization considered is sufficient for a LES computation without necessarily employing a SGS turbulent model. It ends the validation of the non-reacting VOLVO configuration with JAGUAR.

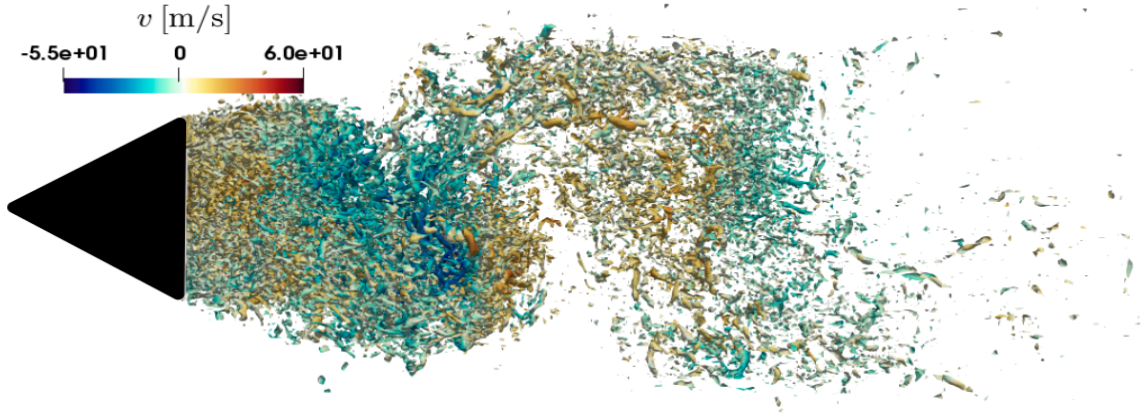


Figure 8.18. – Vortex shedding structure for the non-reacting case visualized using an iso-contour of Q -criterion at $7 \times 10^7 \text{ s}^{-1}$ colored by instantaneous transverse velocity.

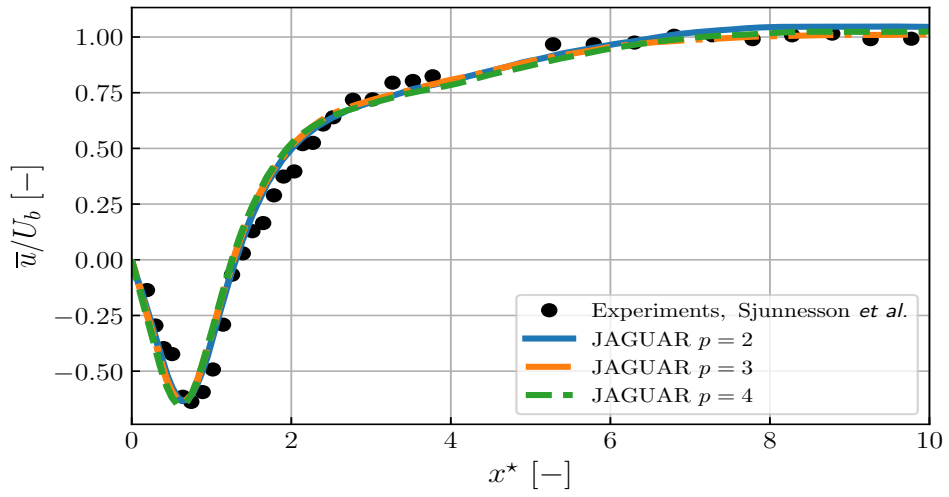


Figure 8.19. – \bar{u}/U_b along $y = 0$ in the non-reacting case.

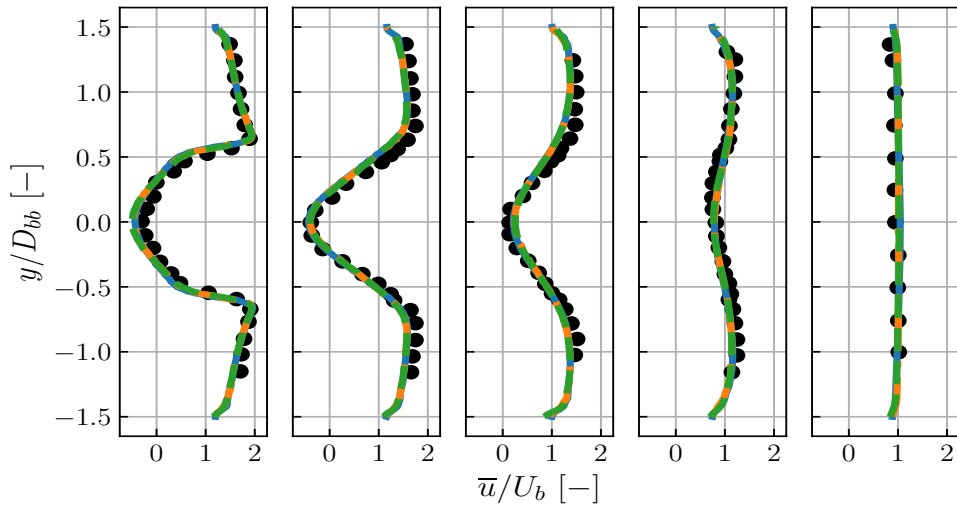


Figure 8.20. – y -profiles of \bar{u}/U_b in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends.

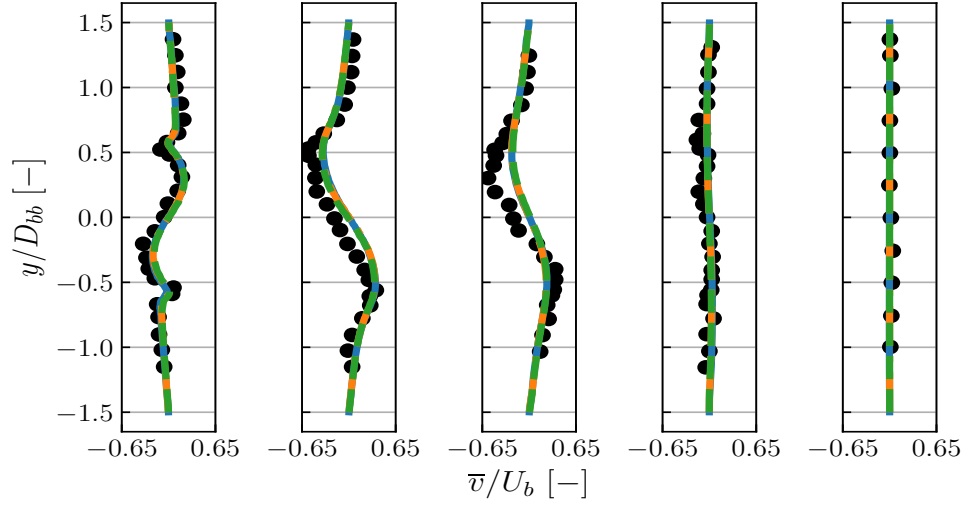


Figure 8.21. y -profiles of \bar{v}/U_b in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends.

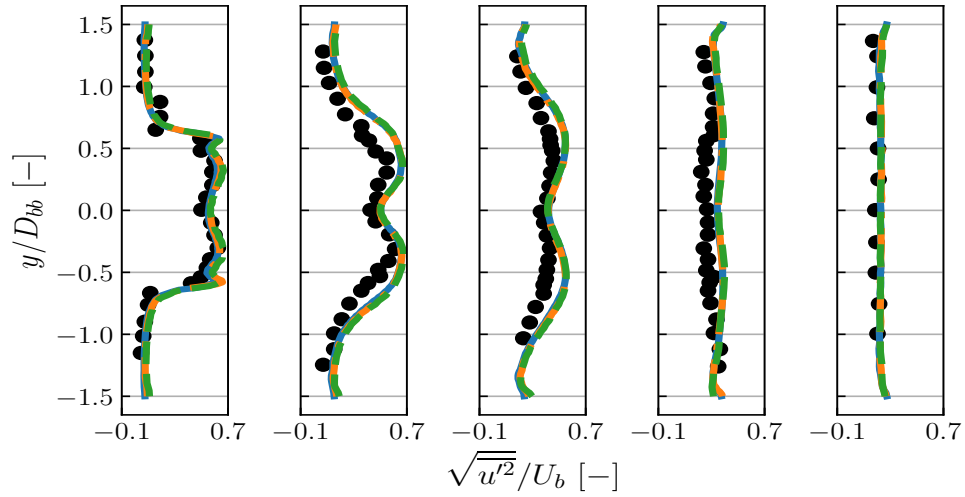


Figure 8.22. y -profiles of $\sqrt{u'^2}/U_b$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends.

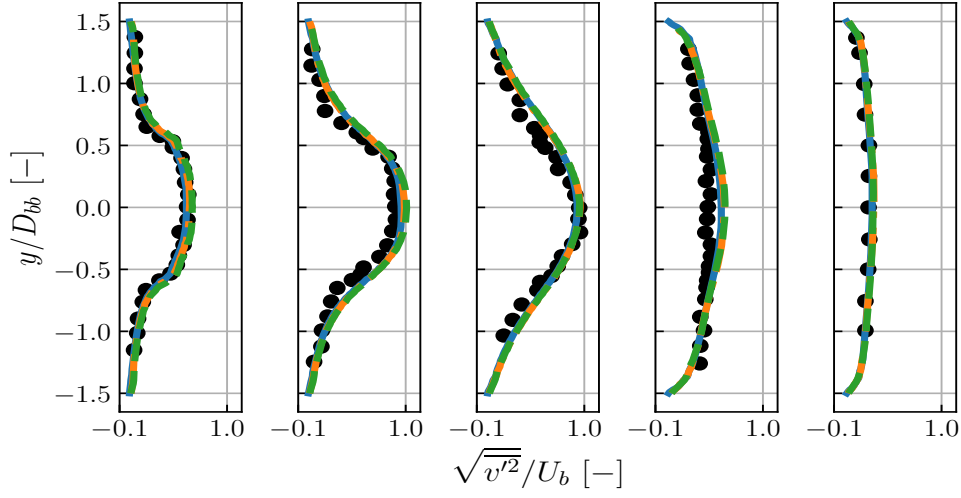


Figure 8.23. – y -profiles of $\sqrt{v'^2}/U_b$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends.

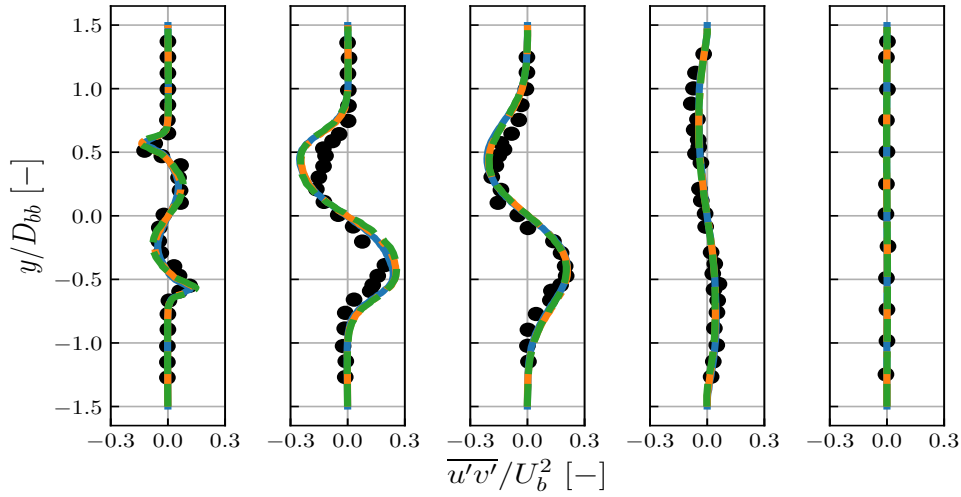


Figure 8.24. – y -profiles of $\overline{u'v'}/U_b^2$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends.

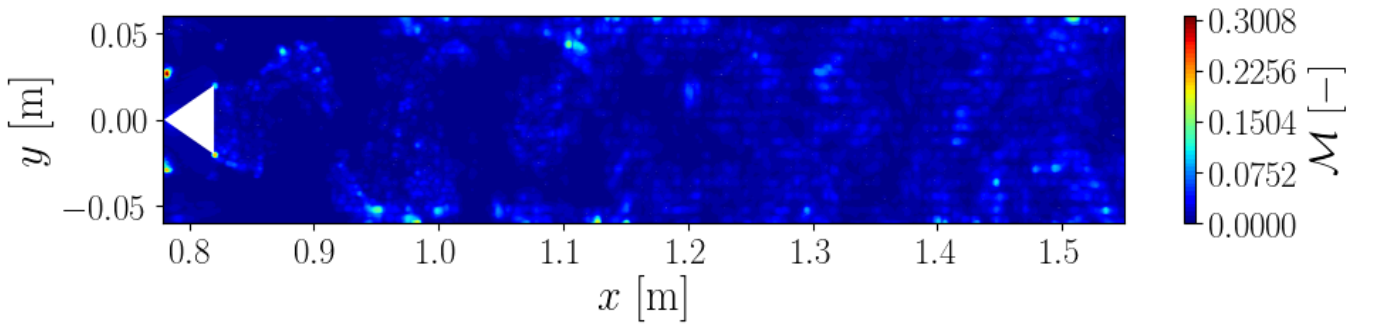


Figure 8.25. – Evolution of Pope's criterion, computed with the Vreman model [157] using $C_{SGS} = 0.10$, on the $p = 3$ discretization near the bluff-body region.

8.4.8. Results analysis on the reacting case

8.4.8.1. Comments on the discretization

As explained in paragraph 8.4.2, the use of local polynomial adaptation was necessary to keep a reasonable number of DOF within the computational domain. The mesh elements and the zones with different polynomial degrees, set by hand in this case, are represented respectively in Figures 8.26 and 8.27.

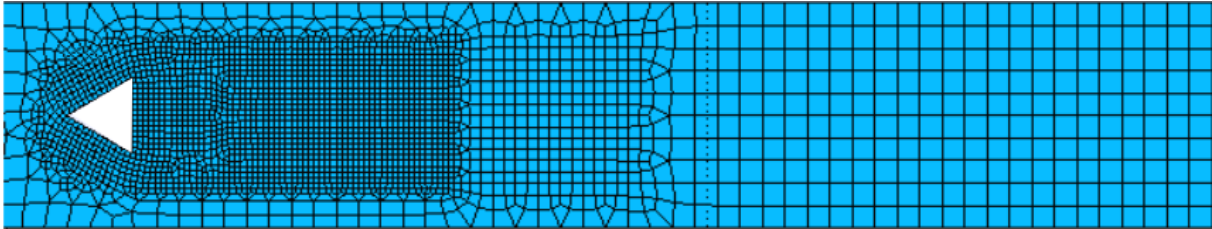


Figure 8.26. – View of the mesh elements near the bluff-body region for the reacting case in the xy -plane.



Figure 8.27. – Polynomial degree distribution near the bluff-body region for the reacting case in the xy -plane.

The need for p -adaptation for the reacting flow is attributed to the following reasons:

- Firstly, the TFLES model requires to not thicken the flame too much near the bluff-body otherwise it will not stay attached. Keeping the non-reacting discretizations Δ_z of 2 mm induced a too strong thickening in this direction, leading to flame blow-off. Thus, a refinement in the spanwise direction was needed so as to not exceed a maximum thickening of the flame of 7 with 7 points inside the flame front.
- However, due to the use of the Gmsh mesh generator [316] with only hexahedral elements, this refinement in the spanwise direction applies all over the computational domain. This leads to a very high number of DOF that is not affordable unless local polynomial degree distribution is employed.

The three different refinement zones illustrated on Figure 8.27 have been chosen to have most of the experimental probes in refined zones. Measurements from x_1 to x_4 are indeed in the very refined zone. However x_5 , x'_4 and x'_5 probes where the average temperature is measured are in the coarsened zone.

Finally, the discretization was slightly coarsened in the xy -plane in the very refined zone close to the bluff-body to also reduce the number of DOF ending up with $\Delta_x = \Delta_y = \Delta_z = 0.6$ mm in this region.

8.4.8.2. Results

Reacting simulations are initialized with a non-reacting solution which is ignited by either surrounding the bluff-body with burnt gases or by filling all the right side of the chamber, from $x = x_{bb}$ to the outlet, with burnt gases.

Figure 8.28 shows an iso-surface of progress variable, based on temperature, at $c = 0.5$, and colored by the instantaneous wrinkling factor. The temperature increase due to the combustion process has entailed the creation of high viscosity and large density ratio regions eliminating the vortex shedding pattern observed in Figure 8.18 as already noticed in previous studies [111, 312]. Figure 8.29 shows

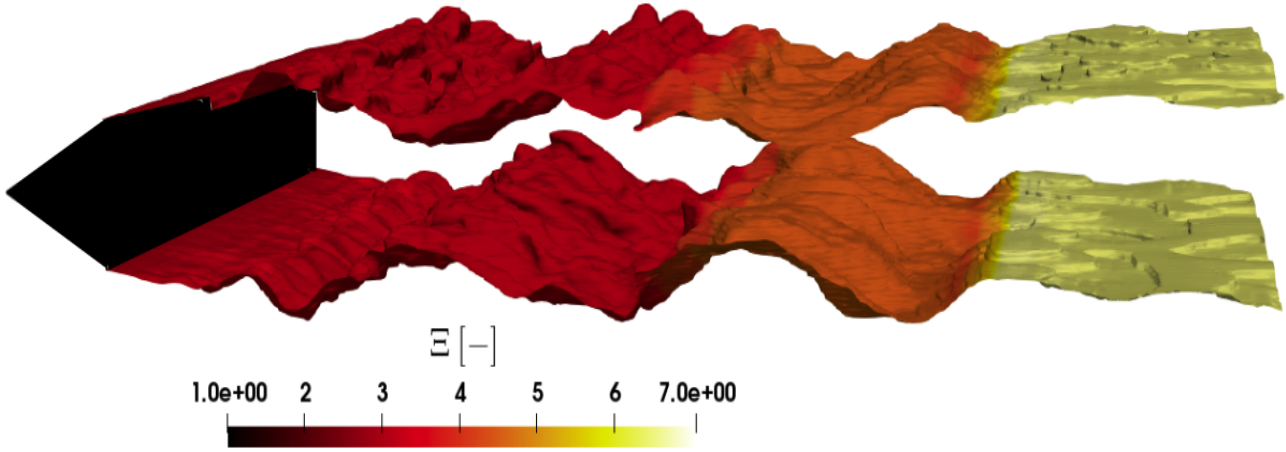


Figure 8.28. – Iso-surface of progress variable $c = 0.5$ colored by instantaneous wrinkling factor.

the evolution of the mean axial velocity along the centerline $y = 0$ in the x^* -space for the present JAGUAR simulation, experiment [121, 122] and other LES performed using the same two-reactions chemical scheme for propane:

- Ghani *et al.* [311] used the AVBP solver [179, 283] with the TTGC scheme for convective fluxes. Their mesh was composed of 37M of DOF with $\Delta_x = 2$ mm in the bluff-body region. The TFLES model was used with the original flame sensor of Legier *et al.* [174] and the Charlette's model [178, 181] for the efficiency function but without specifying if it is the saturated version and if β is constant or dynamically computed. They used the Sigma model [158] as SGS model.
- Tayyab *et al.* [111] have employed a LBM solver that was recently extended to multi-species reacting flows [109, 110]. They had 17.5M of DOF with $\Delta_x = 1$ mm in the bluff-body region. Again, the TFLES model was used but with the generic flame sensor [176] and Charlette's model [178, 181] for the efficiency function also without specifying the Charlette's version and the way β is obtained. They used the Vreman model [157] as SGS model.
- Lv *et al.* [26] used a DG algorithm at constant $p = 3$ in the whole domain. Their discretization was composed of only 2.5M of DOF with $\Delta_x = 3$ mm in the bluff-body region. However, their domain was half smaller in axial direction and of one third only in the spanwise direction of the usual width represented in Figure 8.17. Moreover, there is no mention of any turbulent combustion model or SGS model. This is questionable especially for the combustion model since their Δ_x is far from being sufficient to resolve the flame front.

All these parameters are summed up in Table 8.8.

The evolution of the mean axial velocity along $y = 0$ is globally well-captured by JAGUAR:

- The recirculation zone is slightly shifted compared to experiments but other numerical methods also predict such shift.
- The flow acceleration, due to gas expansion in the reacting case, is correctly captured with a small underestimation for $x^* > 9 \iff x > 1.18$ m. It might be due to the JAGUAR discretization that is coarsened from $x = 1.10$ m to the outlet.

In Figure 8.30-8.34, y -profiles of mean and RMS velocities at the five axial locations $x_1 - x_5$ obtained with JAGUAR are illustrated and compared with experimental data. A general good agreement is

| Numerical Method | DOF [$\times 10^6$] | Δ_x [mm] | Mod combu | Mod SGS |
|--|-----------------------|-----------------|----------------------------------|---------|
| AVBP (TTGC), Ghani <i>et al.</i> [311] | 37 | 2 | TF-Class-Char-?-? | Sigma |
| LBM, Tayyab <i>et al.</i> [111] | 17.5 | 1 | TF-Gener-Char-?-? | Vreman |
| DG $p = 3$, Lv <i>et al.</i> [26] | 2.5 | 3 | ? | ? |
| JAGUAR | 24.5 | 0.6 | TF-Relax-Char-Sat- $\beta = 0.5$ | None |

Table 8.8. – Summary numerical methods used in previous works and this work for comparison on the reacting VOLVO burner. Question tags indicate that no information is available.

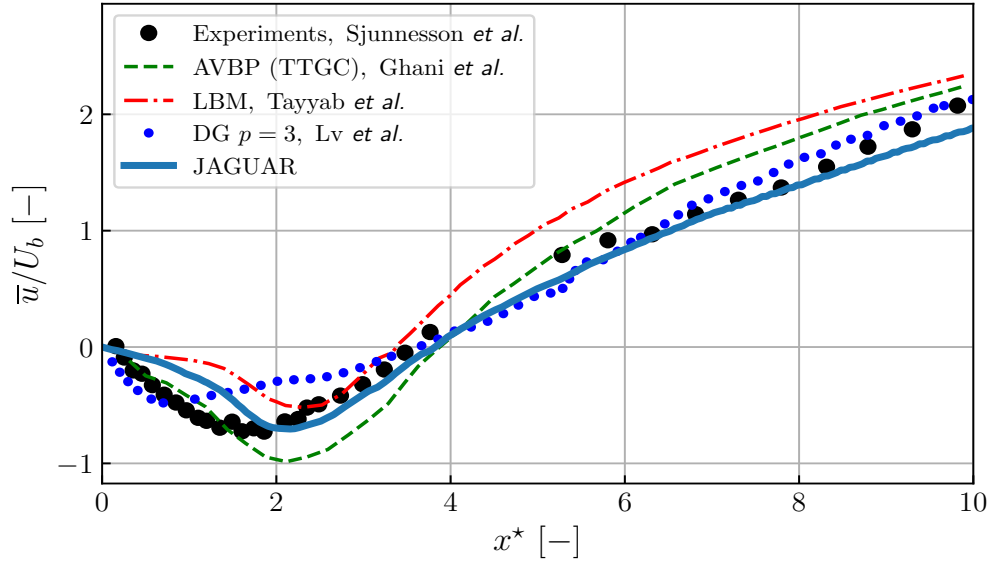


Figure 8.29. – \bar{u}/U_b along $y = 0$ in the reacting case.

observed even at $x = x_5$ where the discretization is coarse. Finally, Figure 8.35 presents y -profiles of the mean temperature at the three axial locations x_4 , x'_4 and x'_5 . Again, the mean temperature profile is globally well captured at each of these three locations. The differences with experimental data at x'_4 and x'_5 locations have already been observed in previous studies [312, 314]. Employing a more detailed chemical scheme and a dynamic β formulation for the saturated Charlette model can also improve the results as evidenced by Rochette *et al.* [312]. Finally, the use of a better polynomial distribution in the future, based on an appropriate sensor (see the discussion in paragraph 8.3.11), will probably help to better discretize x'_4 and x'_5 regions without increasing the total number of DOF in the domain.

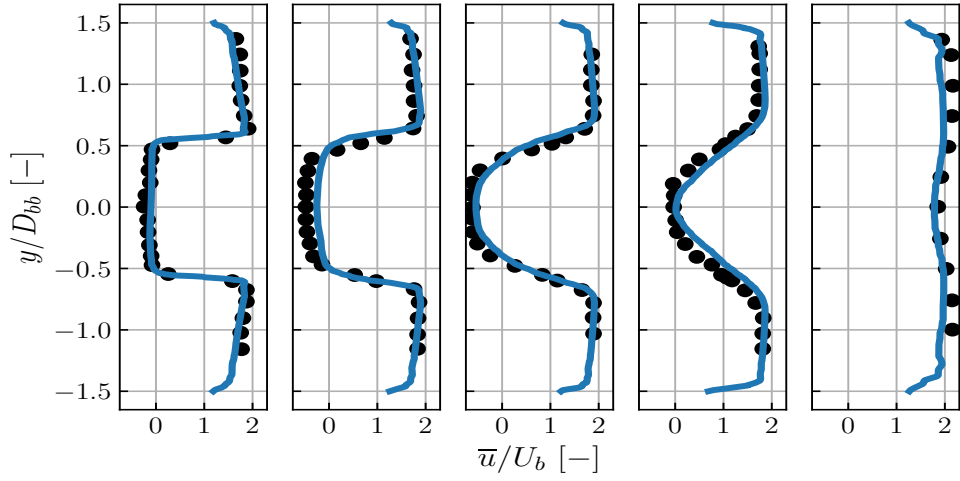


Figure 8.30. – y -profiles of \bar{u}/U_b in the reacting case at $x_1 - x_5$ obtained with JAGUAR.

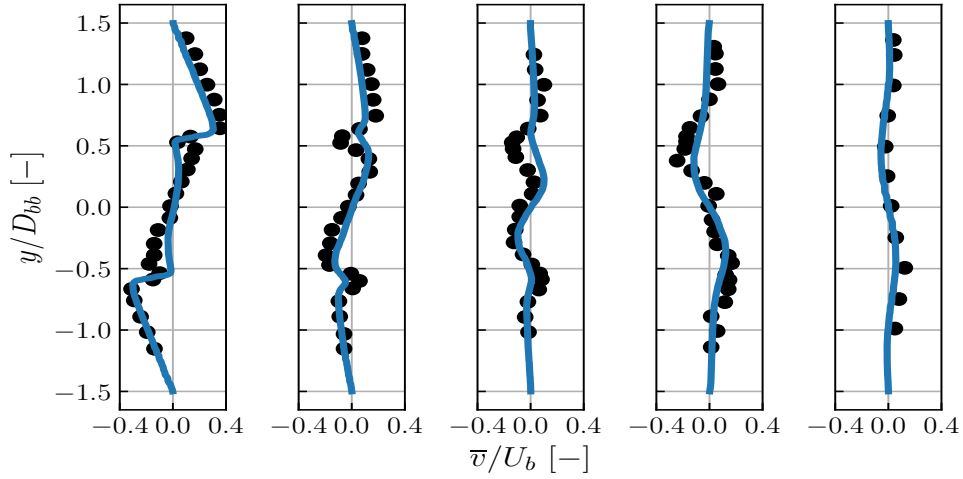


Figure 8.31. – y -profiles of \bar{v}/U_b in the reacting case at $x_1 - x_5$ obtained with JAGUAR.

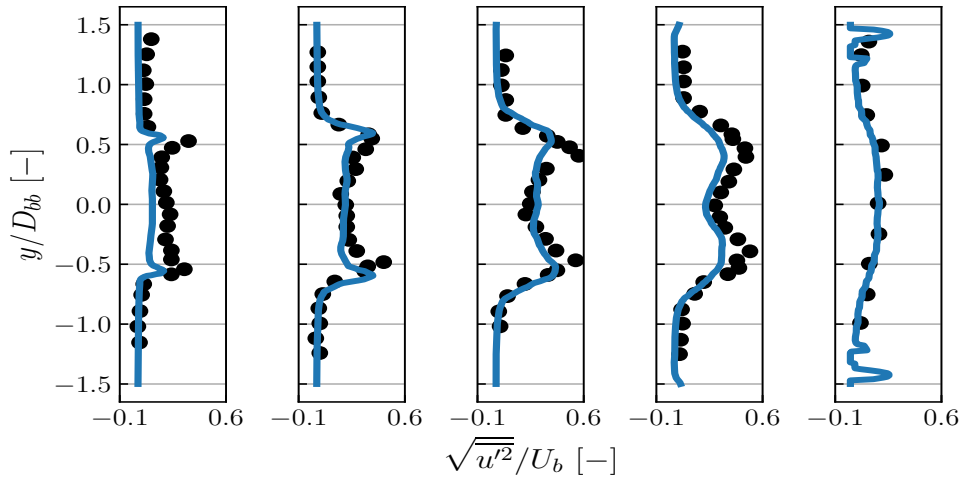


Figure 8.32. – y -profiles of $\sqrt{u'^2}/U_b$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR.

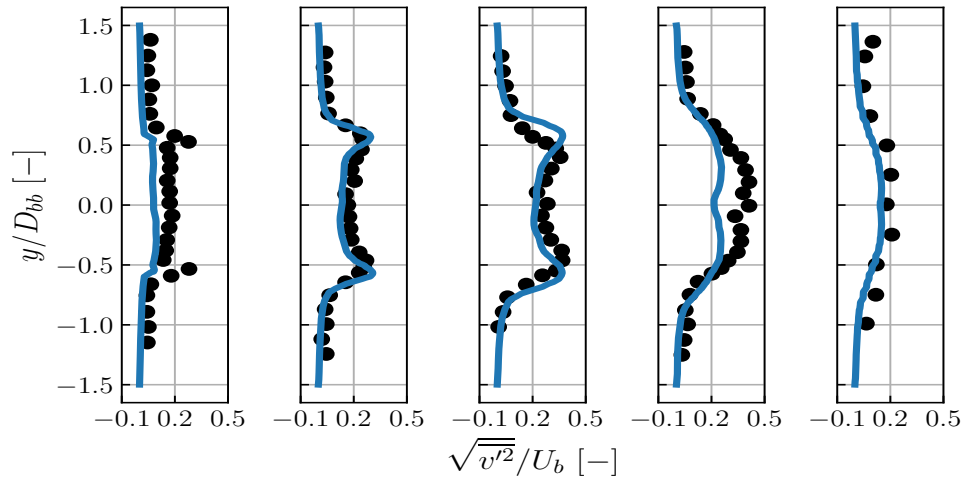


Figure 8.33. $-y$ -profiles of $\sqrt{v'^2}/U_b$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR.

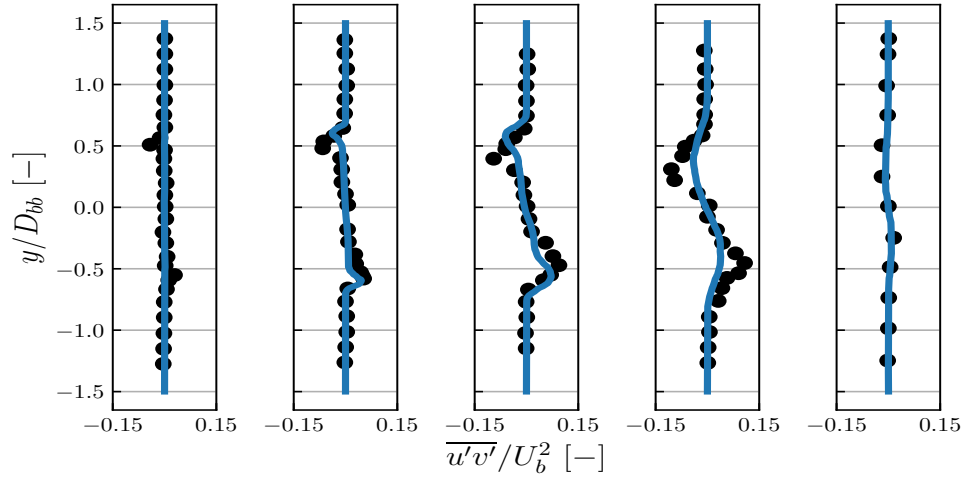


Figure 8.34. $-y$ -profiles of $\overline{u'v'}/U_b^2$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR.

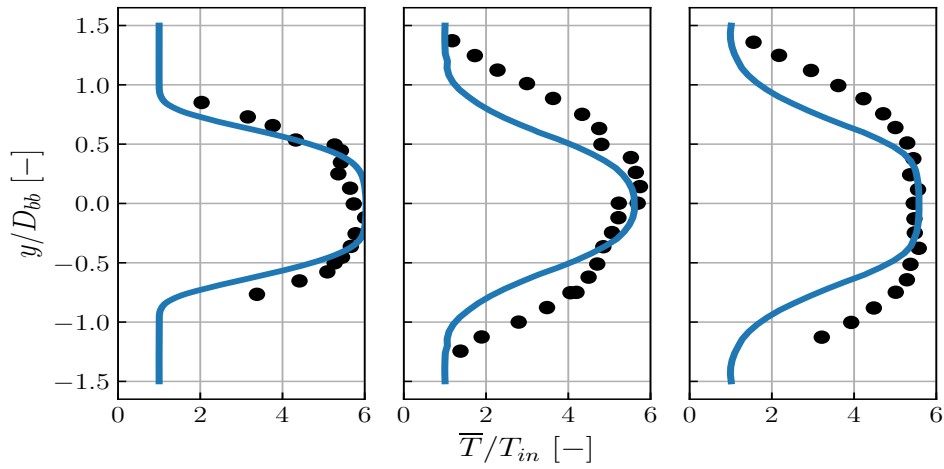


Figure 8.35. $-y$ -profiles of \overline{T}/T_{in} in the reacting case at x_4, x'_4 and x'_5 obtained with JAGUAR.

8.4.8.3. Discussion on computational cost

Since JAGUAR and AVBP are two codes developed at CERFACS which use same species transport approach and same combustion source terms computation, it is possible to compare their iteration cost in the non-reacting and reacting cases for the VOLVO configuration. In particular, in paragraph 8.3.10 it was shown that JAGUAR iteration cost highly increases for reacting simulations and that this increase is not only due to the additional transport of combustion products and the calculation of combustion source terms. Table 8.9 sums up the iteration costs of both non-reacting and reacting flow simulations of the VOLVO configuration using JAGUAR and AVBP with either the LW or TTGC convection scheme. It should be mentioned that non-reacting JAGUAR simulations are the ones of $p = 3$ case. As in the Cambridge burner case, reacting simulations with JAGUAR are well more costly than non-reacting ones although the cost of one transport equation seems still to be around $2 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$. AVBP also shows a large increase of the iteration cost between non-reacting and reacting simulations, for both convection schemes, but this increase is less important. The cost of one transport equation is around $0.78 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ and $1.23 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ respectively for the LW and the TTGC scheme which in both cases is smaller than for JAGUAR. This analysis shows that there is still a high potential for improvement for reducing the cost of transport equations in JAGUAR.

| Code | JAGUAR ($p = 3$) | AVBP (LW) | AVBP (TTGC) |
|------------------------------|--------------------|-----------|-------------|
| VOLVO non-reacting $N_s = 0$ | 11.2 | 9.46 | 18.5 |
| VOLVO non-reacting $N_s = 3$ | 17.6 | 11.8 | 22.2 |
| VOLVO reacting $N_s = 7$ | 42.0 | 15.2 | 30.1 |

Table 8.9. – Iteration costs $\kappa [\mu\text{s.ite}^{-1}.\text{DOF}^{-1}]$ for non-reacting and reacting VOLVO simulations with JAGUAR for the $p = 3$ discretization and AVBP. All computations were done on 760 processors provided by the same cluster.

8.5. Summary of this chapter

In this chapter, the use of the TFLES model with the relaxation flame sensor and saturated Charlette's efficiency function at a constant β value is detailed in the context of the SD discretization. Since two sets of points are employed in the SD method, thickening factors and efficiency functions are calculated at both SP and FP. Simulations of 1D and 2D laminar premixed flames on coarse meshes with flame thickening, have shown that the relaxation flame sensor and the thickening factor are very well computed by JAGUAR and in perfect agreement with AVBP results. Validation on a canonical test case for the operator used in the expression of u'_Δ was also performed with a perfect match between numerical and analytical solutions.

For the turbulent combustion cases, the use of positivity-preserving limiter was needed to stabilize the solution. The implementation of such limiter was therefore detailed and some differences with the usual algorithms presented in the literature are discussed. Basically, the main difference in its use during this work was the necessity of having, besides limitations on species mass fractions and pressure, a limitation on the temperature. It is attributed to the non-linear dependence on temperature of the equations considered in the limitations process due to the temperature dependence of enthalpies in the multi-species case. Moreover, the original versions of this limiter were almost never applied in a multi-species turbulent reacting flow context where this additional constraint on temperature might be crucial.

Then, simulations of the Cambridge flame burner have been presented using two different discretizations, respectively at constant $p = 2$ and $p = 3$ in the whole domain, with both a characteristic distance

between two DOF of 0.3 mm in the flame region. Firstly, non-reacting simulations were carried out to see if both discretizations correctly capture the aerodynamic fields. This was indeed the case since results were in very good agreement with experiments and other LES available in the literature on the same case. The premixed Cambridge flame was simulated afterwards and again results were found in good accordance with experimental data and other LES except for the RMS of temperature which were underestimated by JAGUAR. This effect was said to be originated from the use of the TFLES model entailing a loss of the wrinkling due to the thickening of the flame. However, it was observed that working at a higher p value diminishes this effect, although the flame front is similarly discretized. This highlights the fact that the resolved wrinkling of the flame seems to vary with p . Further investigations, out of the scope of this work, should be considered to see if there is a need to take into account the value of p when computing the efficiency function \mathcal{E} . It is inline with the work of Chapelier and Lodato [159] who developed a SGS turbulent model which adapts the sub-grid dissipation locally to an element based on the polynomial energy spectrum within each element. A comparison of the iteration cost between non-reacting and reacting simulations performed with JAGUAR show that:

- The iteration cost for adding an additional transported species in the flow is $2.4 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for $p = 2$ and $2.2 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for $p = 3$.
- The SDLIFT formulation is extremely expensive for now with respectively 71% and 66% of increase for $p = 2$ and $p = 3$ cases compared with calculations using the average approach of Sun *et al.* [47]. It is the main reason why reacting simulations of the Cambridge burner were much more costly than non-reacting ones with JAGUAR since the average approach which is not stable for reacting cases could not be used. The implementation of the SDLIFT formulation needs to be optimized using both matrix/matrix products with appropriate memory placement and cache blocking.

Finally, some guidelines for using local polynomial adaptation in the future on the Cambridge burner were given in order to put high polynomial degrees only in the flame region to reduce the thickening and better capture the flame wrinkling.

Another 3D turbulent premixed flame, the VOLVO configuration, was computed with JAGUAR firstly in the non-reacting case with constant $p = 2$, $p = 3$ and $p = 4$ discretizations and in a second time in the reacting case using a discretization with $2 \leq p \leq 4$. The non-reacting case is very well reproduced for any value of p with JAGUAR. Results on the reacting simulation are in a good agreement with both experimental data and other LES results on the same configuration. Finally, a comparison between JAGUAR and AVBP in terms of iteration cost on this configuration has revealed that:

- Like JAGUAR, AVBP reacting simulations are well more costly than non-reacting ones but the increase is less important with AVBP.
- AVBP iteration cost for one transport equation is around $0.78 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for the LW scheme and $1.23 \mu\text{s.ite}^{-1}.\text{DOF}^{-1}$ for the TTGC scheme. These values are respectively three and two times smaller than JAGUAR ones although both codes use the same transport modeling. Therefore, it is concluded that there is a large potential to reduce JAGUAR iteration cost in combustion by working on optimizing the resolution of transport equations in JAGUAR.

Conclusion and perspectives

9.1. Discussion

The global objective of this thesis was the extension of the SD method to combustion applications and the study of its interest and the new possibilities it offers compared to already existing numerical methods. To achieve this, it was necessary to develop a stable SD formulation when applied to the reacting multi-species NSE. Then, a thorough numerical validation of this algorithm was performed from 1D laminar premixed flames to 3D turbulent premixed flames showing very satisfying results compared to already existing methods.

The development of a SD algorithm able to simulate multi-species reacting flows started by showing that the classical SD discretization creates strong pressure oscillations when dealing with a thermally perfect gas in the context of a contact discontinuity. This behavior originates from the reconstruction of pressure at FP from the interpolated conservative variables on these points due to the non-linearity in the equation linking ρE and P for a thermally perfect gas. This issue was corrected by firstly evaluating pressure at SP and then interpolating this same pressure at FP ending up with the TUPY SD formulation. With this new formulation associated to extended characteristic and wall boundary conditions to multi-species gas, 1D and 2D laminar premixed flames have been simulated with both simplified two-reactions or semi-detailed ARC chemical schemes. Results were similar to those obtained with well-established combustion solvers CANTERA and AVBP.

Performances of the SD method in combustion were then investigated. A first study conducted on a 2D laminar burner has shown that, for the same error level, it is better in terms of accuracy and computational cost to employ large elements with high p values and not the contrary. This study also demonstrated the usefulness of local p -adaptation which allows to significantly reduce the computational cost of the SD method, making it very competitive with already existing ones, without losing accuracy. A second study made on several 1D and 2D laminar premixed flames with very coarse discretizations highlighted the unstable property of the average approach for the diffusion scheme at interface FP, leading to the development of the SDLIFT formulation. Using this new formulation, an improvement of the stability was observed making the SD method less sensitive to the number of points inside the flame front than the AVBP solver. Such result is very interesting in the context of turbulent combustion where usually it is way too costly to have a large number of points inside the flame front.

Simulations of turbulent combustion cases required first simulations of non-reacting turbulent flows with the SD code JAGUAR employed in this thesis. In particular, the implementation of a turbulent inlet boundary condition was needed. A synthetic random Fourier method was chosen to generate turbulent perturbations at the inlet and was validated on several test cases with different geometries. It is a good compromise in terms of quality of the generated turbulence and computational cost compared to other existing methods for injecting turbulence at an inlet of a domain. Moreover, its formalism allows the use of any kind of turbulent spectrum making it very flexible. However, this methodology is not optimum for reproducing near-wall turbulence but it can be easily coupled to an artificial tripping approach or extended to inject non-homogeneous and anisotropic turbulence which, in both cases, has exhibited to improve near-wall turbulence description.

In order to move to reacting turbulent flows, the implementation of a turbulent combustion model in JAGUAR was needed. The TFLES model was then adapted to the SD formalism and incorporated

in JAGUAR. After checking that this model performed as expected on simple 1D and 2D laminar flames, simulations of two academic 3D turbulent burners were carried out namely the Cambridge flame burner and the VOLVO configuration. Concerning the Cambridge burner, results were found in very good agreement with both experiments and other existing LES showing the capability of the SD method developed in this work to simulate complex turbulent combustion. One significant observation is that, at constant number of points inside the flame front, a higher value of p better predicts the wrinkling of the flame that is lost due to thickening. Thus, it clearly advocates for the use of local p -adaptation in the future in order to reduce the thickening of the flame front at a lower computational cost. Nevertheless, this test case has also highlighted that the SDLIFT formulation remains extremely costly in the current JAGUAR implementation making reacting computations very expensive compared to non-reacting ones which can be run without this formulation. This is due to its very young implementation in JAGUAR that does not benefit from matrix/matrix operations and cache blocking. Finally, the VOLVO configuration was correctly simulated with JAGUAR, in both non-reacting and reacting cases, where most of the results show very good agreement with experimental data and other LES previously published. For the reacting case, local p -adaptation was employed to make the simulation affordable and with sufficient accuracy in the zones of interest.

9.2. Future work

This thesis has demonstrated that the SD method can handle simulations of multiple premixed combustion cases and give new alternatives to better resolve the flame front, at a smaller cost compared to the constant p case, using local p -adaptation. Consequently, it is very important to keep working on local p -adaptation and especially on finding the appropriate sensors for the flame resolution. Comparisons with the current state of the art in combustion, which is using a FV method coupled with the AMR technique, is a necessary step that needs to be done in the next years. Moreover, since the polynomial degree seems to have an influence on the resolved flame wrinkling, it could be interesting to work on the adaptation of the efficiency function in the TFLES model to account for this effect.

The work conducted here also brought to light the limitations in terms of iteration cost of JAGUAR especially in the reacting turbulent cases. It is still a very young CFD code which needs to be optimized to remain competitive for simulating complex reacting turbulent flows. The main sources of this increase of iteration cost were identified and needs now to be treated with surely an important improvement of JAGUAR performances.

The development of the SDLIFT formulation, as a new way to evaluate interface states and gradients for viscous fluxes, demonstrated the impact of the diffusion scheme on the results in the multi-species reacting case. Thus, there is a need to understand precisely why the stability limit of the usual average approach of Sun *et al.* [47] seems lower than for the SDLIFT approach. Consequently, a stability analysis between these two formulations would be of great interest.

A last point also requires to be tackled for the future of the SD method in combustion which is to deal with simplex elements (triangles and tetrahedra) rather than only tensor-product elements (quadrilaterals and hexahedra) like in this thesis. Indeed, real combustion chambers have very complicated geometries which are easily meshed using tetrahedra cells. This will require firstly to find stable FP locations on tetrahedra for $p \geq 3$ since the current state of the art works for $p = 2$ at most. Secondly, the adaptation of characteristic boundary conditions is necessary because, for now, the treatment uses the direction per direction property of tensor-product elements to impose flux derivatives. Finally, since local p -adaptation seems to be essential for the future of the SD method in combustion, simplex elements must support local p -adaptation which to the author's knowledge, has never been published in the SD literature.

Appendices

Useful calculations

A.1. Quadrature rule for Gauss-Chebyshev points of the first kind in SD

The Gauss-Chebyshev quadrature rule, using N Gauss-Chebyshev quadrature points of the first kind, has the objective to compute:

$$\int_0^1 \frac{h(\xi)}{\sqrt{1-\xi^2}} d\xi = \sum_{j=1}^N \omega_j h(\xi_j) \quad (\text{A.1})$$

where ξ_j is given by Eq. (4.14) and the weights ω_j on $[0, 1]$ are set as:

$$\omega_j = \frac{\pi}{2N} \quad (\text{A.2})$$

However, in the SD framework, it is only $h(\xi)$ that needs to be integrated on $N = p + 1$ SP defined by the ξ_j 's. Thus, the weights ω_j presented in Eq. (A.2) cannot be used since they were derived when $\sqrt{1-\xi^2}$ was at the denominator of the integrated function. Consequently, new values of ω_j , such that a polynomial of degree p at most must be integrated exactly, have to be found. There are $p + 1$ unknowns for the ω_j then $p + 1$ equations are needed to find the corresponding weights associated to each ξ_j . These $p + 1$ equations can be obtained by considering the integration of $\xi^p, \dots, \xi^0 = 1$:

$$\begin{aligned} \int_0^1 \xi^p d\xi &= \sum_{j=1}^{p+1} \omega_j \xi_j^p = \frac{1}{p+1} \\ &\vdots \\ \int_0^1 \xi^0 d\xi &= \sum_{j=1}^{p+1} \omega_j \xi_j^0 = \frac{1}{1} \end{aligned} \quad (\text{A.3})$$

This system of equations can be put into matrix form:

$$\begin{bmatrix} 1 & \dots & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_1 & \dots & \xi_j & \dots & \xi_{p+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_1^p & \dots & \xi_j^p & \dots & \xi_{p+1}^p \end{bmatrix} \begin{bmatrix} \omega_1 \\ \vdots \\ \omega_j \\ \vdots \\ \omega_{p+1} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1/j \\ \vdots \\ \frac{1}{p+1} \end{bmatrix} \quad (\text{A.4})$$

where the matrix in the left-hand-side is a Vandermonde matrix. Solving Eq. (A.4) for each value of p will give the values of the ω_j 's to use for the integration of any polynomial of degree p at most using Gauss-Chebyshev quadrature points of the first kind. Here is an example for $p = 2$ where there are 3 SP which have the following locations given by Eq. (4.14):

$$\xi_1 = \frac{1}{2} \left(1 - \frac{\sqrt{3}}{2} \right), \xi_2 = \frac{1}{2}, \xi_3 = \frac{1}{2} \left(1 + \frac{\sqrt{3}}{2} \right) \quad (\text{A.5})$$

The 3 weights values will be given by solving:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ \xi_1 & \xi_2 & \xi_3 \\ \xi_1^2 & \xi_2^2 & \xi_3^2 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \end{bmatrix} \implies \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} 0.22222222 \\ 0.55555556 \\ 0.22222222 \end{bmatrix} \quad (\text{A.6})$$

A.2. Compute normal vector at a given FP

A.2.1. Link between the isoparametric and physical normal vectors

The objective here is to find a relation between the normal vector at an interface FP in the isoparametric domain $\hat{\mathbf{n}} = (n_\xi, n_\eta, n_\zeta)^T$ and the normal vector at this same interface FP in the physical domain $\mathbf{n} = (n_x, n_y, n_z)^T$. Starting from the definition of $\hat{\mathbf{n}}$ saying that for any tangent vector $\hat{\mathbf{t}} = (t_\xi, t_\eta, t_\zeta)^T$ to this FP:

$$\hat{\mathbf{n}} \bullet \hat{\mathbf{t}} = 0 \iff \hat{\mathbf{n}}^T \cdot \hat{\mathbf{t}} = 0 \quad (\text{A.7})$$

where \bullet means the dot product between 2 vectors and \cdot means the product between 2 tensors. Using the components of each vector it can be viewed as:

$$n_\xi t_\xi + n_\eta t_\eta + n_\zeta t_\zeta = 0 \iff (n_\xi, n_\eta, n_\zeta) \cdot \begin{pmatrix} t_\xi \\ t_\eta \\ t_\zeta \end{pmatrix} = 0 \quad (\text{A.8})$$

This is more a general case that is written here compared to the SD case where $\hat{\mathbf{n}}$ can take only three possible vector values: $(1, 0, 0)^T$ for ξ -FP, $(0, 1, 0)^T$ for η -FP or $(0, 0, 1)^T$ for ζ -FP. Now, J and J^{-1} are introduced in the right side of Eq. (A.7) following:

$$\begin{aligned} \hat{\mathbf{n}}^T \cdot J^{-1} J \cdot \hat{\mathbf{t}} &= 0 \\ \iff \hat{\mathbf{n}}^T \cdot \left[[J^{-1}]^T \right]^T \cdot J \cdot \hat{\mathbf{t}} &= 0 \\ \iff \left[[J^{-1}]^T \cdot \hat{\mathbf{n}} \right]^T \cdot J \cdot \hat{\mathbf{t}} &= 0 \\ \iff [J^{-1}]^T \cdot \hat{\mathbf{n}} \bullet J \cdot \hat{\mathbf{t}} &= 0 \end{aligned} \quad (\text{A.9})$$

so that by identification:

$$\mathbf{n} = [J^{-1}]^T \cdot \hat{\mathbf{n}} \quad (\text{A.10})$$

$$\mathbf{t} = J \cdot \hat{\mathbf{t}} \quad (\text{A.11})$$

with $\mathbf{t} = (t_x, t_y, t_z)^T$ an arbitrary tangent vector to the interface FP in the physical domain.

A.2.2. Physical normal vector at interface FP along ξ -direction

In this case, $\hat{\mathbf{n}}_{FP} = (1, 0, 0)^T$ so that Eq. (A.10) gives:

$$\mathbf{n}_{FP} = \begin{pmatrix} \xi_x \\ \xi_y \\ \xi_z \end{pmatrix} \quad (\text{A.12})$$

Usually, unit normal vector are preferred and \mathbf{n}_{FP} is replaced by:

$$\mathbf{n}_{FP}^u = \frac{\mathbf{n}_{FP}}{\|\mathbf{n}_{FP}\|_2} = \frac{1}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}} \begin{pmatrix} \xi_x \\ \xi_y \\ \xi_z \end{pmatrix} \quad (\text{A.13})$$

In Eq. (A.13), $\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} = A_{FP}$ is the area at the interface FP.

A.2.3. Physical normal vector at interface FP along η -direction

In this case, $\hat{\mathbf{n}}_{FP} = (0, 1, 0)^T$ so that the unit normal vector and the area at the interface FP are:

$$\mathbf{n}_{FP}^u = \frac{1}{\sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}} \begin{pmatrix} \eta_x \\ \eta_y \\ \eta_z \end{pmatrix} \text{ and } A_{FP} = \sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2} \quad (\text{A.14})$$

A.2.4. Physical normal vector at interface FP along ζ -direction

In this case, $\hat{\mathbf{n}}_{FP} = (0, 0, 1)^T$ so that the unit normal vector and the area at the interface FP are:

$$\mathbf{n}_{FP}^u = \frac{1}{\sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2}} \begin{pmatrix} \zeta_x \\ \zeta_y \\ \zeta_z \end{pmatrix} \text{ and } A_{FP} = \sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2} \quad (\text{A.15})$$

A.3. Tangential vectors for a given normal vector

Given an unit normal vector $\mathbf{n}^u = (n_x^u, n_y^u, n_z^u)^T$ at a boundary FP, two unit vectors can be defined to create a local orthonormal basis at the boundary FP:

- If $|n_z^u| < 0.7$:

$$\mathbf{t}_1^u = \frac{1}{\sqrt{(n_x^u)^2 + (n_y^u)^2}} \begin{pmatrix} n_y^u \\ -n_x^u \\ 0 \end{pmatrix} \text{ and } \mathbf{t}_2^u = \frac{1}{\sqrt{(n_x^u)^2 + (n_y^u)^2}} \begin{pmatrix} -n_x^u n_z^u \\ -n_y^u n_z^u \\ (n_x^u)^2 + (n_y^u)^2 \end{pmatrix} \quad (\text{A.16})$$

- Else:

$$\mathbf{t}_1^u = \frac{1}{\sqrt{(n_y^u)^2 + (n_z^u)^2}} \begin{pmatrix} 0 \\ -n_z^u \\ n_y^u \end{pmatrix} \text{ and } \mathbf{t}_2^u = \frac{1}{\sqrt{(n_y^u)^2 + (n_z^u)^2}} \begin{pmatrix} (n_y^u)^2 + (n_z^u)^2 \\ -n_x^u n_y^u \\ -n_x^u n_z^u \end{pmatrix} \quad (\text{A.17})$$

A.4. Expression of total energy for a thermally perfect gas similar to the one of a calorically perfect gas

For a thermally perfect gas, total energy is expressed as:

$$\rho E = \rho h_s - P + \rho \frac{\|\mathbf{u}\|_2^2}{2} \quad (\text{A.18})$$

In the case of a calorically perfect gas with constant $\gamma = C_p/C_v$, Eq. (A.18) becomes:

$$\rho E = \frac{P}{\gamma - 1} + \rho \frac{\|\mathbf{u}\|_2^2}{2} \quad (\text{A.19})$$

because $h_s = C_p T$. However, for a thermally perfect gas, h_s is given by Eq. (2.14) recalled here for clarity:

$$h_s = \sum_{k=1}^{N_s} Y_k \int_{T_0}^T C_{pk}(T') dT' \quad (\text{A.20})$$

By introducing the effective heat capacity at constant pressure:

$$\tilde{C}_{pk}(T) = \frac{1}{T - T_0} \int_{T_0}^T C_{pk}(T') dT' \quad (\text{A.21})$$

Eq. (A.18) is now given by:

$$\rho E = \rho(T - T_0) \sum_{k=1}^{N_s} Y_k \tilde{C}_{pk} - \rho \bar{R} T \sum_{k=1}^{N_s} \frac{Y_k}{W_k} + \rho \frac{\|\mathbf{u}\|_2^2}{2} \quad (\text{A.22})$$

To simplify Eq. (A.22), the reference temperature is defined to be $T_0 = 0$ K so that:

$$\rho E = \rho T \sum_{k=1}^{N_s} Y_k \left(\tilde{C}_{pk} - \frac{\bar{R}}{W_k} \right) + \rho \frac{\|\mathbf{u}\|_2^2}{2} = \frac{PW}{\bar{R}} \sum_{k=1}^{N_s} Y_k \left(\tilde{C}_{pk} - \frac{\bar{R}}{W_k} \right) + \rho \frac{\|\mathbf{u}\|_2^2}{2} \quad (\text{A.23})$$

By identification, Eq. (A.23) is similar to Eq. (A.19) if an effective heat capacity ratio $\tilde{\gamma}$ (not to be confused with the one of Eq. (4.111) for the Roe's average of the heat capacity ratio) is introduced such that:

$$\tilde{\gamma} - 1 = \frac{\bar{R}}{W \sum_{k=1}^{N_s} Y_k \left(\tilde{C}_{pk} - \frac{\bar{R}}{W_k} \right)} \iff \tilde{\gamma} = \frac{\sum_{k=1}^{N_s} Y_k \tilde{C}_{pk}}{\sum_{k=1}^{N_s} Y_k \left(\tilde{C}_{pk} - \frac{\bar{R}}{W_k} \right)} \quad (\text{A.24})$$

A.5. System of equations for a NSCBC inlet imposing velocities, temperature and species mass fractions

The objective is to obtain time-derivatives expressions of the primitive variables $\mathbf{Q} = (\rho, u, v, w, P, Y_k)^T$ as a function of \mathcal{N} and \mathcal{S} . To do so, Eq. (5.24) is multiplied by $P_{\mathbf{Q}}^{-1} = R_{A_\xi}$ defined in Eq. (F.42) ending up with:

$$|J| \frac{\partial \rho}{\partial t} = -n_x^u (\mathcal{N}_1^* + \mathcal{S}_1) - n_y^u (\mathcal{N}_2^* + \mathcal{S}_2) - n_z^u (\mathcal{N}_3^* + \mathcal{S}_3) - \frac{\rho}{\sqrt{2}c} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \quad (\text{A.25})$$

$$|J| \frac{\partial u}{\partial t} = \frac{n_z^u c}{\rho} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{n_y^u c}{\rho} (\mathcal{N}_3^* + \mathcal{S}_3) - \frac{n_x^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (\text{A.26})$$

$$|J| \frac{\partial v}{\partial t} = -\frac{n_z^u c}{\rho} (\mathcal{N}_1^* + \mathcal{S}_1) + \frac{n_x^u c}{\rho} (\mathcal{N}_3^* + \mathcal{S}_3) - \frac{n_y^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (\text{A.27})$$

$$|J| \frac{\partial w}{\partial t} = \frac{n_y^u c}{\rho} (\mathcal{N}_1^* + \mathcal{S}_1) - \frac{n_x^u c}{\rho} (\mathcal{N}_2^* + \mathcal{S}_2) - \frac{n_z^u}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ - \mathcal{N}_- - \mathcal{S}_-) \quad (\text{A.28})$$

$$|J| \frac{\partial P}{\partial t} = -\frac{\rho c}{\sqrt{2}} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \quad (\text{A.29})$$

$$|J| \frac{\partial Y_k}{\partial t} = -(\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}) \quad \text{for } k = 1, N_s \quad (\text{A.30})$$

Then, time-derivatives of u , v and w are given in Eq. (A.26)-Eq. (A.28) and those of Y_k in Eq. (A.30). It remains to find an expression for $|J|(\partial T/\partial t)$. Thanks to Eq. (2.16) the differential of T is given by:

$$\begin{aligned}
\partial T &= \frac{T}{P} \partial P - \frac{T}{\rho} \partial \rho - TW \partial \left(\frac{1}{W} \right) = \frac{T}{P} \partial P - \frac{T}{\rho} \partial \rho - TW \sum_{k=1}^{N_s} \frac{\partial Y_k}{W_k} \\
\Rightarrow |J| \frac{\partial T}{\partial t} &= \frac{T}{P} |J| \frac{\partial P}{\partial t} - \frac{T}{\rho} |J| \frac{\partial \rho}{\partial t} - TW \sum_{k=1}^{N_s} \frac{1}{W_k} |J| \frac{\partial Y_k}{\partial t} \\
\Rightarrow |J| \frac{\partial T}{\partial t} &= -\frac{T\rho c}{\sqrt{2}P} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \\
&\quad + \frac{T}{\rho} \left[n_x^u (\mathcal{N}_1^* + \mathcal{S}_1) + n_y^u (\mathcal{N}_2^* + \mathcal{S}_2) + n_z^u (\mathcal{N}_3^* + \mathcal{S}_3) + \frac{\rho}{\sqrt{2}c} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \right] \\
&\quad + TW \sum_{k=1}^{N_s} \frac{\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}}{W_k} \\
\Rightarrow |J| \frac{\partial T}{\partial t} &= \frac{T}{\rho} \left[n_x^u (\mathcal{N}_1^* + \mathcal{S}_1) + n_y^u (\mathcal{N}_2^* + \mathcal{S}_2) + n_z^u (\mathcal{N}_3^* + \mathcal{S}_3) \right] - \frac{T}{\sqrt{2}} \underbrace{\left(\frac{\rho c}{P} - \frac{1}{c} \right)}_{\frac{\gamma-1}{c}} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \\
&\quad + TW \sum_{k=1}^{N_s} \frac{\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}}{W_k} \\
\Rightarrow |J| \frac{\partial T}{\partial t} &= \frac{T}{\rho} \left[n_x^u (\mathcal{N}_1^* + \mathcal{S}_1) + n_y^u (\mathcal{N}_2^* + \mathcal{S}_2) + n_z^u (\mathcal{N}_3^* + \mathcal{S}_3) \right] - \frac{T(\gamma-1)}{\sqrt{2}c} (\mathcal{N}_+^* + \mathcal{S}_+ + \mathcal{N}_- + \mathcal{S}_-) \\
&\quad + TW \sum_{k=1}^{N_s} \frac{\mathcal{N}_{5+k}^* + \mathcal{S}_{5+k}}{W_k}
\end{aligned} \tag{A.31}$$

A.6. How to choose parameters of the relaxation flame sensor

There is a need to explain how to choose the different values of ψ_0 , τ_0 , τ_1 and D_ψ used in the flame sensor. Starting from the one-dimensional analysis it can be shown that the filter width varies as [175]:

$$\Delta_{filt} \sim \sqrt{D_\psi \tau_1} \log(\psi_0) \sim \sqrt{D_\psi \tau_1} \tag{A.32}$$

by neglecting the logarithmic dependence on ψ_0 . Eq. (A.32) shows that τ_0 and ψ_0 have very small influence on Δ_{filt} so that they are commonly set at the following values:

$$\tau_0 = 20\Delta t \tag{A.33}$$

$$\psi_0 = 20 \tag{A.34}$$

The time τ_1 can be taken proportional to the chemical timescale:

$$\tau_1 = \alpha \tau_{chem} = \alpha \frac{\delta_L^0}{S_L^0} \tag{A.35}$$

The diffusion coefficient D_ψ can be taken proportional to the thermal diffusivity D_T through Eq. (A.36) or by considering a species Schmidt number Sc_ψ with Eq. (A.37):

$$D_\psi = \beta D_T \tag{A.36}$$

$$Sc_\psi = \frac{Pr}{\beta} \tag{A.37}$$

Since flame parameters δ_L^0 , S_L^0 and thermal diffusion coefficient are almost constant for given flame conditions, the filter width can be controlled roughly by α and β because:

$$\Delta_{filt} \sim \sqrt{D_\psi \tau_1} \sim \sqrt{\beta D_T \alpha \tau_{chem}} \sim \sqrt{\alpha \beta} \delta_L^0 \quad (\text{A.38})$$

using $\delta_L^0 \sim \sqrt{D_T \tau_{chem}}$. Eq. (A.38) shows that the filter width and the thermal flame thickness are proportional to $\sqrt{\alpha \beta}$ which makes it easier to parametrize.

A.6.1. Choosing α

This parameter will set the value of τ_1 . Actually, to have an easier control of Δ_{filt} in both fresh and burnt gases, there are a value for α in the fresh gases noted α_{cold} and a value for α in the burnt gases noted α_{hot} . The recommendations proposed in [175] are:

$$\alpha_{cold} \approx 0.1 \text{ and } \alpha_{hot} \approx \frac{\alpha_{cold}}{10} \approx 0.01 \quad (\text{A.39})$$

but these values might be slightly different depending on the chemical scheme considered. Numerically, the switch between fresh and burnt gases is done at a temperature $T_{switch} = 1600$ K as advised in [175].

A.6.2. Choosing β

This parameter will be set through Sc_ψ and is recommended to be taken as:

$$Sc_\psi = \min_k (Sc_k) \quad (\text{A.40})$$

which is usually around $Sc_H \approx 0.12$.

A.7. Algebraic equations for the positivity-preserving limiter

A.7.1. For the limitation of pressure

For a multi-species perfect gas, pressure is expressed following Eq. (A.41):

$$P = \sum_{k=1}^{N_s} \rho Y_k h_{sk} - \rho E + \frac{u^2 + v^2 + w^2}{2\rho} = \sum_{k=1}^{N_s} U_{5+k} h_{sk} - U_5 + \frac{U_2^2 + U_3^2 + U_4^2}{2U_1} \quad (\text{A.41})$$

where the U_i are the components of \mathbf{U} . Eq. (8.14) using the pressure definition in Eq. (A.41) can be written as:

$$\begin{aligned} & \underbrace{2 \left(\bar{U}_1^e + t_\epsilon^\alpha \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \right) \sum_{k=1}^{N_s} \left(\bar{U}_{5+k}^e + t_\epsilon^\alpha \left(\hat{U}_{5+k}^{e,\alpha} - \bar{U}_{5+k}^e \right) \right) h_{sk}}_A \\ & - \underbrace{2 \left(\bar{U}_1^e + t_\epsilon^\alpha \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \right) \left(\bar{U}_5^e + t_\epsilon^\alpha \left(\hat{U}_5^{e,\alpha} - \bar{U}_5^e \right) \right)}_B \\ & + \underbrace{\left(\bar{U}_2^e + t_\epsilon^\alpha \left(\hat{U}_2^{e,\alpha} - \bar{U}_2^e \right) \right)^2 + \left(\bar{U}_3^e + t_\epsilon^\alpha \left(\hat{U}_3^{e,\alpha} - \bar{U}_3^e \right) \right)^2 + \left(\bar{U}_4^e + t_\epsilon^\alpha \left(\hat{U}_4^{e,\alpha} - \bar{U}_4^e \right) \right)^2}_C \\ & = \underbrace{2\epsilon_P \left(\bar{U}_1^e + t_\epsilon^\alpha \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \right)}_D \end{aligned} \quad (\text{A.42})$$

where by introducing $\bar{h}_s^e = \sum_{k=1}^{N_s} \bar{U}_{5+k}^e h_{sk}$ and $\hat{h}_s^{e,\alpha} = \sum_{k=1}^{N_s} \hat{U}_{5+k}^{e,\alpha} h_{sk}$:

$$A = 2 \left[\bar{U}_1^e \bar{h}_s^e + t_\epsilon^\alpha \left(\left(\hat{U}_1^{e,\alpha} - 2\bar{U}_1^e \right) \bar{h}_s^e + \bar{U}_1^e \hat{h}_s^{e,\alpha} \right) + (t_\epsilon^\alpha)^2 \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \left(\hat{h}_s^{e,\alpha} - \bar{h}_s^e \right) \right] \quad (\text{A.43})$$

$$B = 2 \left[\bar{U}_1^e \bar{U}_5^e + t_\epsilon^\alpha \left(U_5^{e,\alpha} \bar{U}_1^e - 2\bar{U}_1^e \bar{U}_5^e + \hat{U}_1^{e,\alpha} \bar{U}_5^e \right) + (t_\epsilon^\alpha)^2 \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \left(U_5^{e,\alpha} - \bar{U}_5^e \right) \right] \quad (\text{A.44})$$

$$C = \left(\bar{U}_2^e \right)^2 + \left(\bar{U}_3^e \right)^2 + \left(\bar{U}_4^e \right)^2 + 2t_\epsilon^\alpha \left[\bar{U}_2^e \left(U_2^{e,\alpha} - \bar{U}_2^e \right) + \bar{U}_3^e \left(U_3^{e,\alpha} - \bar{U}_3^e \right) + \bar{U}_4^e \left(U_4^{e,\alpha} - \bar{U}_4^e \right) \right] + (t_\epsilon^\alpha)^2 \left[\left(U_2^{e,\alpha} - \bar{U}_2^e \right)^2 + \left(U_3^{e,\alpha} - \bar{U}_3^e \right)^2 + \left(U_4^{e,\alpha} - \bar{U}_4^e \right)^2 \right] \quad (\text{A.45})$$

$$D = 2\epsilon_P \left(\bar{U}_1^e + t_\epsilon^\alpha \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \right) \quad (\text{A.46})$$

such that Eq. (A.42) is:

$$A - B + C = D \quad (\text{A.47})$$

By identification, to have an equation of the form:

$$a (t_\epsilon^\alpha)^2 + b t_\epsilon^\alpha + c = 0 \quad (\text{A.48})$$

a , b and c have the following values:

$$a = \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \left(\hat{h}_s^{e,\alpha} - \bar{h}_s^e - U_5^{e,\alpha} + \bar{U}_5^e \right) + \frac{\left(U_2^{e,\alpha} - \bar{U}_2^e \right)^2 + \left(U_3^{e,\alpha} - \bar{U}_3^e \right)^2 + \left(U_4^{e,\alpha} - \bar{U}_4^e \right)^2}{2} \quad (\text{A.49})$$

$$b = \left(\hat{U}_1^{e,\alpha} - 2\bar{U}_1^e \right) \bar{h}_s^e + \bar{U}_1^e \left(\hat{h}_s^{e,\alpha} - U_5^{e,\alpha} + 2\bar{U}_5^e + \epsilon_P \right) - \hat{U}_1^{e,\alpha} \left(\bar{U}_5^e + \epsilon_P \right) + \bar{U}_2^e \left(U_2^{e,\alpha} - \bar{U}_2^e \right) + \bar{U}_3^e \left(U_3^{e,\alpha} - \bar{U}_3^e \right) + \bar{U}_4^e \left(U_4^{e,\alpha} - \bar{U}_4^e \right) \quad (\text{A.50})$$

$$c = \bar{U}_1^e \left(\bar{h}_s^e - \bar{U}_5^e - \epsilon_P \right) + \frac{\left(\bar{U}_2^e \right)^2 + \left(\bar{U}_3^e \right)^2 + \left(\bar{U}_4^e \right)^2}{2} \quad (\text{A.51})$$

It should be mentioned that an approximation is made in Eq. (A.42) due to h_{sk} in $\hat{h}_s^{e,\alpha}$ that is evaluated at temperature $T \left(\hat{U}^{e,\alpha} \right)$ to avoid solving a non-linear equation.

A.7.2. For the limitation of temperature

In this paragraph, $\hat{U}^{e,\alpha}$ is used for $\tilde{U}^{e,\alpha}$ without loss of generality but for clarity purposes. Using Eq. (A.41), Eq. (8.16) can be written as:

$$2\epsilon_T \left(\bar{U}_1^e + t_\epsilon^\alpha \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \right) \bar{R} \sum_{k=1}^{N_s} \frac{\bar{U}_{5+k}^e + t_\epsilon^\alpha \left(\hat{U}_{5+k}^{e,\alpha} - \bar{U}_{5+k}^e \right)}{W_k} = A - B + C \quad (\text{A.52})$$

where A , B and C have been defined in Eq. (A.43)-Eq. (A.45) so that only D is different from pressure equation. In this case, by introducing $\overline{(\rho/W)}^e = \sum_{k=1}^{N_s} \bar{U}_{5+k}^e / W_k$ and $\widehat{(\rho/W)}^{e,\alpha} = \sum_{k=1}^{N_s} \hat{U}_{5+k}^{e,\alpha} / W_k$, D is given by:

$$D = 2\epsilon_T \bar{R} \left[\bar{U}_1^e \overline{\left(\frac{\rho}{W} \right)}^e + t_\epsilon^\alpha \left(\bar{U}_1^e \left(\widehat{\left(\frac{\rho}{W} \right)}^{e,\alpha} - 2\overline{\left(\frac{\rho}{W} \right)}^e \right) + \hat{U}_1^{e,\alpha} \overline{\left(\frac{\rho}{W} \right)}^e \right) \right] + 2\epsilon_T \bar{R} (t_\epsilon^\alpha)^2 \left(\hat{U}_1^{e,\alpha} - \bar{U}_1^e \right) \left(\widehat{\left(\frac{\rho}{W} \right)}^{e,\alpha} - \overline{\left(\frac{\rho}{W} \right)}^e \right) \quad (\text{A.53})$$

Therefore, coefficients a , b and c for an equation as in Eq. (A.48) are:

$$a = \left(\widehat{U}_1^{e,\alpha} - \overline{U}_1^e \right) \left(\widehat{h}_s^{e,\alpha} - \overline{h}_s^e - U_5^{e,\alpha} + \overline{U}_5^e - \epsilon_T \overline{R} \left(\left(\frac{\rho}{W} \right)^{e,\alpha} - \overline{\left(\frac{\rho}{W} \right)^e} \right) \right) + \frac{\left(U_2^{e,\alpha} - \overline{U}_2^e \right)^2 + \left(U_3^{e,\alpha} - \overline{U}_3^e \right)^2 + \left(U_4^{e,\alpha} - \overline{U}_4^e \right)^2}{2} \quad (\text{A.54})$$

$$b = \left(\widehat{U}_1^{e,\alpha} - 2\overline{U}_1^e \right) \overline{h}_s^e + \overline{U}_1^e \left(\widehat{h}_s^{e,\alpha} - U_5^{e,\alpha} + 2\overline{U}_5^e + \epsilon_T \overline{R} \left(\left(\frac{\rho}{W} \right)^{e,\alpha} - 2\overline{\left(\frac{\rho}{W} \right)^e} \right) \right) - \widehat{U}_1^{e,\alpha} \left(\overline{U}_5^e + \epsilon_T \overline{R} \overline{\left(\frac{\rho}{W} \right)^e} \right) + \overline{U}_2^e \left(U_2^{e,\alpha} - \overline{U}_2^e \right) + \overline{U}_3^e \left(U_3^{e,\alpha} - \overline{U}_3^e \right) + \overline{U}_4^e \left(U_4^{e,\alpha} - \overline{U}_4^e \right) \quad (\text{A.55})$$

$$c = \overline{U}_1^e \left(\overline{h}_s^e - \overline{U}_5^e - \epsilon_T \overline{R} \overline{\left(\frac{\rho}{W} \right)^e} \right) + \frac{\left(\overline{U}_2^e \right)^2 + \left(\overline{U}_3^e \right)^2 + \left(\overline{U}_4^e \right)^2}{2} \quad (\text{A.56})$$

In that case, no approximation is done in Eq. (A.52) because the temperature at which the h_{sk} , used in $\widehat{h}_s^{e,\alpha}$, are computed is really fixed to ϵ_T .

A.8. Link between centerline and bulk velocities for the axial velocity field set in Cambridge burner simulations

The mean velocity profiles imposed at the inlet of both inner and outer injectors of the Cambridge burner follows an analytical power-law profile given by Eq. (8.18) introduced in paragraph 8.3.3. In this Section, $r_{1,i/o}$, $r_{2,i/o}$, $U_{cl,i/o}$ and $U_{b,i/o}$ will be denoted respectively by r_1 and r_2 , U_{cl} and U_b for clarity purposes. The value of U_{cl} is computed to satisfy:

$$\begin{aligned} U_b &= \frac{1}{\pi (r_2^2 - r_1^2)} \int_{r_1}^{r_2} U_{cl} \left[1 - \left(\frac{2r - (r_1 + r_2)}{r_2 - r_1} \right)^4 \right] 2\pi r dr \\ \Rightarrow \frac{U_b}{U_{cl}} &= 1 - \frac{2 \int_{r_1}^{r_2} [16r^5 - 32r^4 (r_1 + r_2) + 24r^3 (r_1 + r_2)^2 - 8r^2 (r_1 + r_2)^3 + r (r_1 + r_2)^4] dr}{(r_2 + r_1) (r_2 - r_1)^5} \\ \Rightarrow \frac{U_b}{U_{cl}} &= 1 - \frac{\frac{16}{3} (r_2^6 - r_1^6) - \frac{64}{5} (r_1 + r_2) (r_2^5 - r_1^5) + 12 (r_1 + r_2)^2 (r_2^4 - r_1^4) - \frac{16}{3} (r_1 + r_2)^3 (r_2^3 - r_1^3) + (r_1 + r_2)^4 (r_2^2 - r_1^2)}{(r_2 + r_1) (r_2 - r_1)^5} \end{aligned} \quad (\text{A.57})$$

Algorithms

B.1. Algorithms to compute species and heat fluxes in the multi-species case under Hirschfelder and Curtiss approximation

This section gives the algorithms to compute species and energy fluxes introduced respectively in paragraphs 2.4.3.3 and 2.4.4.

B.1.1. Algorithm to compute species flux

Under Hirschfelder and Curtiss approximation, $\mathbf{M} = [\mathbf{M}_k]_{1 \leq k \leq N_s}$ is usually computed at each point following Algorithm 1.

Algorithm 1 Algorithm to compute \mathbf{M} at a given point

- 1: Get values of $\mathbf{Y} = [Y_k]_{1 \leq k \leq N_s}$ and $\nabla \mathbf{Y} = [\nabla Y_k]_{1 \leq k \leq N_s}$.
 - 2: Compute $\nabla (1/W)$ using Eq. (2.70).
 - 3: Compute \mathbf{V}^c using Eq. (2.71) and save $D_k (\nabla Y_k - Y_k W \nabla (1/W))$ into \mathbf{M}_k for $k = 1, N_s$.
 - 4: Get final \mathbf{M}_k by doing $\mathbf{M}_k = \rho (\mathbf{M}_k - Y_k \mathbf{V}^c)$ according to Eq. (2.72) for $k = 1, N_s$.
-

B.1.2. Algorithm to compute the energy flux

The main steps to compute the energy flux when species flux is under Hirschfelder and Curtiss approximation are summed up in Algorithm 2.

Algorithm 2 Algorithm to compute \mathbf{q} at a given point

- 1: Get values of $\mathbf{Y} = [Y_k]_{1 \leq k \leq N_s}$ and of $\nabla \mathbf{Y} = [\nabla Y_k]_{1 \leq k \leq N_s}$.
 - 2: Compute $\nabla (1/W)$, \mathbf{V}^c and \mathbf{M} following Algorithm 1.
 - 3: Get T , P , ρ , ∇P , $\nabla \rho$ and $h_{sk}(T)$.
 - 4: Compute W using Eq. (2.9).
 - 5: Compute \mathbf{q} using Eq. (2.77).
-

B.2. Algorithm for gradient computation with SDLIFT formulation

The main steps to compute the gradients in Cartesian coordinates of conservative variables at ξ -FP in 3D using the SDLIFT formulation (see paragraph 4.4.2) are summed up in Algorithm 3.

Algorithm 3 Computation of gradients in Cartesian coordinates at ξ -FP with SDLIFT formulation

- 1: \mathbf{U} at SP in the isoparametric domain, $\widehat{\mathbf{U}}_{(j_1, j_2, j_3)}^e$, and discontinuous \mathbf{U} at FP in the physical domain, $\mathbf{U}_{(k_1, j_2, j_3)}$, are known.
 - 2: Compute $(\partial \mathbf{U}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ by using Eq. (4.116) evaluated at ξ -FP.
 - 3: **if** BR1 or BR2 is used **then**
 - 4: From interface values of $\mathbf{U}_{(k_1, j_2, j_3)}$, compute common states using Eq. (4.27).
 - 5: From common states, compute $\widetilde{\mathbf{U}}_{h(k_1, j_2, j_3)}^e$ using Eq. (4.122) and $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ using Eq. (4.117).
 - 6: **if** BR1 is used **then**
 - 7: From $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ values, compute common normal derivatives using Eq. (4.30).
 - 8: **else if** BR2 is used **then**
 - 9: From interface values of $(\partial \mathbf{U}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ and common states, compute $(\partial \widetilde{\mathbf{U}}_h^{e, L} / \partial \xi)$ and $(\partial \widetilde{\mathbf{U}}_h^{e, R} / \partial \xi)$ at each interface FP using the last equal signs in Eq. (4.44) and Eq. (4.45).
 - 10: Compute common normal derivatives using Eq. (4.43).
 - 11: **end if**
 - 12: **else if** IC is used **then**
 - 13: From interface values of $\mathbf{U}_{(k_1, j_2, j_3)}$ and $(\partial \mathbf{U}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ compute common states using Eq. (4.49).
 - 14: From common states, compute $\widetilde{\mathbf{U}}_{h(k_1, j_2, j_3)}^e$ using Eq. (4.122) and $(\partial \widetilde{\mathbf{U}}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ using Eq. (4.117).
 - 15: From interface values of $(\partial \mathbf{U}_h^e / \partial \xi)_{(k_1, j_2, j_3)}$ and common states, compute also common normal derivatives based on Eq. (4.46).
 - 16: **end if**
 - 17: Compute $(\partial \mathbf{U} / \partial \xi)_{(k_1, j_2, j_3)}^C$ using Eq. (4.119) evaluated at ξ -FP.
 - 18: Compute tangential derivatives $(\partial \mathbf{U} / \partial \eta)$ and $(\partial \mathbf{U} / \partial \zeta)$ at ξ -FP using either classical or Huynh's approach.
 - 19: Compute gradients in Cartesian coordinates using Eq. (4.113).
 - 20: At interface FP, average the gradients in Cartesian coordinates between the left and right states.
-

B.3. Algorithms for Navier-Stokes Characteristic Boundary Conditions in generalized coordinates

This section gives the algorithms to employ characteristic boundary conditions using either Kim and Lee or Fievet et *al.* formulation described respectively in paragraphs 5.1.2 and 5.1.3.

B.3.1. Kim and Lee's formulation

The main steps to compute characteristic boundary conditions in generalized coordinates using Kim and Lee [234, 235] formulation are summed up in Algorithm 4.

Algorithm 4 Kim and Lee algorithm to apply characteristic boundary conditions in generalized coordinates

- 1: Use Eq. (5.17) to compute an initial value of \mathcal{L} .
 - 2: Some values of \mathcal{L} are modified to account for the NSCBC treatment. This gives new ξ -convective characteristics \mathcal{L}^* .
 - 3: A new value of $(\partial \hat{\mathbf{E}}_c / \partial \xi)$, noted $(\partial \hat{\mathbf{E}}_c / \partial \xi)^*$, is obtained by using Eq. (5.17) with \mathcal{L}^* .
 - 4: Finally, $(\partial \hat{\mathbf{E}}_c / \partial \xi)^*$ is put back into Eq. (5.6) to march $\hat{\mathbf{U}}$ in time.
-

B.3.2. Fievet et *al.* formulation

The main steps to compute characteristic boundary conditions in generalized coordinates using the recent Fievet et *al.* [117] formulation are summed up in Algorithm 5.

Algorithm 5 Fievet et *al.* algorithm to apply characteristic boundary conditions in generalized coordinates

- 1: Evaluate initial guesses for \mathcal{N} and \mathcal{S} at NSCBC FP in $\xi = 1$ using Eq. (5.25) and Eq. (5.26).
 - 2: Modify some values of \mathcal{N} to account for the NSCBC treatment. This gives \mathcal{N}^* .
 - 3: Compute new values of $(\partial \hat{\mathbf{E}} / \partial \xi)(1)$ by inverting Eq. (5.25) with \mathcal{N}^* . This gives $(\partial \hat{\mathbf{E}} / \partial \xi)^*(1)$.
 - 4: Compute the new flux value $\hat{\mathbf{E}}_{p+2}$ using Eq. (5.28). This gives $\hat{\mathbf{E}}_{p+2}^*$.
 - 5: Compute the new value of $(\partial \hat{\mathbf{E}} / \partial \xi)^*(SP)$ which will be put back into Eq. (4.84) to march $\hat{\mathbf{U}}$ in time.
-

B.4. Global algorithm to generate synthetic Fourier turbulence

The turbulent velocity vector given by Eq. (6.69) in paragraph 6.4.2 is generated following Algorithm 6 where L_{ref} is a characteristic size of the domain where turbulence is generated and L_{cut} is the smallest size of vortices that the mesh can resolved.

Algorithm 6 Algorithm for computing the turbulent velocity vector using a synthetic random Fourier method

- 1: The user gives u_{rms} , L_e , a reference length L_{ref} , a cutting length L_{cut} , a number of Fourier modes N , the type of energy spectrum $i_{spe} = 0$ for PP or $i_{spe} = 1$ for VKP and the kind of repartition for modes $i_{rep} = 0$ for linear or $i_{rep} = 1$ for logarithmic. The kinematic viscosity of the fluid ν is also assumed to be known.
 - 2: Compute κ_e and ϵ depending on the type of spectrum chosen:
 - 3: **if** $i_{spe} = 0$ **then**
 - 4: $\kappa_e = 2\pi/L_e$ and ϵ is obtained with Eq. (6.60)
 - 5: **else if** $i_{spe} = 1$ **then**
 - 6: κ_e and ϵ are obtained with respectively Eq. (6.64) and Eq. (6.66)
 - 7: **end if**
 - 8: Compute $\kappa_1 = 2\pi/L_{ref}$, $\kappa_N = \min(\kappa_{Kol}, 2\pi/L_{cut})$ where κ_{Kol} is given by Eq. (6.65).
 - 9: **if** $i_{rep} = 0$ **then**
 - 10: Compute $\Delta\kappa_{lin}$ with Eq. (6.71)
 - 11: **else if** $i_{rep} = 1$ **then**
 - 12: Compute $\Delta\kappa_{log}$ with Eq. (6.72)
 - 13: **end if**
 - 14: **for** $n = 1, \dots, N$ **do**
 - 15: **if** $i_{rep} = 0$ **then**
 - 16: Compute $\kappa_n = \kappa_1 + (n - 1) \Delta\kappa_{lin}$
 - 17: **else if** $i_{rep} = 1$ **then**
 - 18: Compute $\kappa_n = \exp(\ln(\kappa_1) + (n - 1) \Delta\kappa_{log})$
 - 19: **end if**
 - 20: Compute $\Delta\kappa_n$ and then u_{tn} using Eq. (6.70).
 - 21: Compute ϕ_n and ψ_n using the probabilities defined in Eq. (6.75).
 - 22: Compute θ_n using Eq. (6.77).
 - 23: Compute Cartesian components of κ_n using Eq. (6.78).
 - 24: Construct σ_n following Eq. (6.79).
 - 25: **end for**
 - 26: $\mathbf{u}'_{in}(\mathbf{x})$ can be computed using Eq. (6.69).
-

B.5. Algorithm for applying TFLES model with relaxation flame sensor using a SD discretization

The dynamic TFLES model using the relaxation flame sensor introduced in Section 3.3, and applied in a SD context in paragraph 8.1.1.2, is summed up in Algorithm 7. As highlighted in paragraph 8.1.1.2, this implementation assumes implicitly that δ_L^0 , S_L^0 and $|\dot{\omega}_F|_{1D}^{max}$ are constant during the computation which will become untrue if flames are not perfectly premixed.

Algorithm 7 TFLES model using relaxation flame sensor for a SD discretization

- 1: The user gives n_{pts}^{ff} , δ_L^0 , S_L^0 , $|\dot{\omega}_F|_{1D}^{max}$, α_{cold} , α_{hot} , $S_{c\psi}$, β and the type of Charlette's formulation used $i_{ch} = 0$ for the non-saturated or $i_{ch} = 1$ for the saturated one.
 - 2: Compute $\tau_{1,cold}$ and $\tau_{1,hot}$ using Eq. (A.35) with respectively α_{cold} and α_{hot} .
 - 3: Compute \mathcal{F}_{max} using Eq. (3.32), where Δ_x is given by Eq. (4.138), and set $\psi_0 = 20$.
 - 4: **if** not_restart **then**
 - 5: Set $\mathcal{S}_{SP}(\mathbf{x}, t = 0) = \mathcal{E}_{SP}(\mathbf{x}, t = 0) = 1$ and compute $\dot{\omega}_{F,SP}$ at all SP.
 - 6: Compute $\mathcal{S}_{SP}(\mathbf{x}, t = 0)$ using Eq. (3.34) at all SP.
 - 7: Initialize $[\rho\psi]_{SP}$ at all SP by doing: $[\rho\psi(\mathbf{x}, t = 0)]_{SP} = [\rho\mathcal{S}(\mathbf{x}, t = 0)]_{SP}$.
 - 8: Compute $\mathcal{F}_{SP}(\mathbf{x}, t = 0)$ at all SP with Eq. (3.38) where $\hat{\mathcal{S}}_{SP}(\mathbf{x}, t = 0) = \mathcal{S}_{SP}(\mathbf{x}, t = 0)$.
 - 9: **else**
 - 10: Compute $\mathcal{S}_{SP}(\mathbf{x}, t)$ at all SP using Eq. (3.34).
 - 11: Compute $\hat{\mathcal{S}}_{SP}(\mathbf{x}, t)$ at all SP using Eq. (3.37).
 - 12: Compute $\mathcal{F}_{SP}(\mathbf{x}, t)$ at all SP using Eq. (3.38).
 - 13: **if** $i_{ch} = 0$ **then**
 - 14: Compute $u'_{\Delta,SP}$ at all SP using Eq. (3.53).
 - 15: Compute $\Delta_{SP} = \mathcal{F}_{SP}\delta_L^0$ at all SP.
 - 16: Compute $Re_{\Delta,SP}$ at all SP using Eq. (3.52).
 - 17: Compute \mathcal{E}_{SP} at all SP using Eq. (3.55).
 - 18: **else if** $i_{ch} = 1$ **then**
 - 19: Compute \mathcal{E}_{SP} at all SP using Eq. (3.57) with $\Delta_{SP}/\delta_L^0 = \mathcal{F}_{SP}$.
 - 20: **end if**
 - 21: Interpolate \mathcal{F}_{SP} and \mathcal{E}_{SP} at all FP to get \mathcal{F}_{FP} and \mathcal{E}_{FP} .
 - 22: Average \mathcal{F}_{FP} and \mathcal{E}_{FP} at interface FP.
 - 23: Multiply all diffusion coefficients at all FP by $\mathcal{F}_{FP}\mathcal{E}_{FP}$.
 - 24: Compute diffusive fluxes at all FP for reacting NSE equations and for Eq. (3.35).
 - 25: Compute $\dot{\omega}_{k,SP}$ at all SP and multiply them by $\mathcal{E}_{SP}/\mathcal{F}_{SP}$.
 - 26: Compute $\dot{\omega}_{\psi,SP}$ at all SP using Eq. (3.36) and multiply it by $\mathcal{E}_{SP}/\mathcal{F}_{SP}$ only if $\mathcal{S}_{SP} < 0.05$.
 - 27: Update conservative variables in time.
 - 28: **end if**
-

Definitions and results on some families of orthogonal polynomials

C.1. Space of polynomial

On interval $I = [-1, 1]$, let's define the L^2 -inner product of two polynomials f and g defined on I by:

$$\langle f, g \rangle = \int_{-1}^1 f(\xi) g(\xi) d\xi \quad (\text{C.1})$$

For an integer $m \geq 0$, let's also denote by $\mathbf{P}^m(I)$ the space of polynomials of degree m or less on I . $\mathbf{P}^m(I)$ is a vector space of dimension $m+1$ on which a simple basis is $(\xi^0, \xi^1, \dots, \xi^m)^T$. A polynomial f defined on interval $I = [-1, 1]$ is orthogonal to $\mathbf{P}^m(I)$ if it is orthogonal to a basis of $\mathbf{P}^m(I)$:

$$\langle f, \xi^k \rangle = \int_{-1}^1 f(\xi) \xi^k d\xi = 0 \text{ for } 0 \leq k \leq m \quad (\text{C.2})$$

Eq. (C.2) shows that being orthogonal to $\mathbf{P}^m(I)$ provides $m+1$ conditions for building f .

C.2. Legendre polynomials

C.2.1. Definition

Still on $I = [-1, 1]$ and for an integer $k \geq 0$, the Legendre polynomial P_k is defined as the unique polynomial of degree k that is orthogonal to $\mathbf{P}^{k-1}(I)$ and verifies $P_k(1) = 1$. Consequently, for $k > m$, P_k is orthogonal to \mathbf{P}^m . Additionally, since P_k is of degree k , it has exactly k zeros in I which are refer to as Gauss-Legendre points.

C.2.2. Quadrature rule

The k Gauss-Legendre points, noted $[\xi_i]_{1 \leq i \leq k}^T$, are used in the Gauss-Legendre quadrature rule for integrating any smooth function h on I with [189]:

$$\int_{-1}^1 h(\xi) d\xi \approx \sum_{i=1}^k \omega_i h(\xi_i) \quad (\text{C.3})$$

where ω_i is the weight associated to the i -th quadrature point ξ_i . For Gauss-Legendre points, it can be obtained analytically through:

$$\omega_i = \frac{2}{(1 - \xi_i^2) [P'_k(\xi_i)]^2} \quad (\text{C.4})$$

In practice, values of ξ_i and ω_i are tabulated for each degree k . Eq. (C.3) is exact if h is a polynomial of degree $2k-1$ or less. It is the best interest of using such quadrature rule since it is the only one that has this property. Other quadrature rules with k points cannot integrate exactly a polynomial of degree $2k-1$.

C.2.3. Recurrence formula

The Legendre polynomials can be obtained using the recurrence formula [317]:

$$(k+1)P_{k+1}(\xi) = (2k+1)\xi P_k(\xi) - kP_{k-1}(\xi) \text{ for } k \geq 2$$

$$P_0 = 1 \text{ and } P_1 = \xi \quad (\text{C.5})$$

Legendre polynomials and their corresponding Gauss-Legendre points up to $k = 4$ are represented in Figure C.1 on $I = [-1, 1]$. It can be seen from Figure C.1 that for any Legendre polynomials, its zeros,

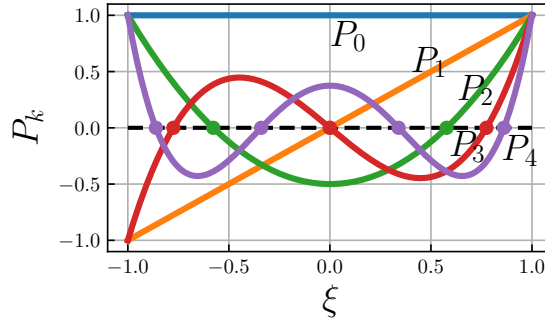


Figure C.1. – Legendre polynomials and Gauss-Legendre points on $[-1, 1]$

which are Gauss-Legendre points, are strictly inside I .

C.2.4. Some useful properties

At the end points of I , P_k and its derivative P'_k take the following values:

$$P_k(-1) = (-1)^k \text{ and } P_k(1) = 1 \quad (\text{C.6})$$

$$P'_k(-1) = \frac{(-1)^{k-1} k(k+1)}{2} \text{ and } P'_k(1) = \frac{k(k+1)}{2} \quad (\text{C.7})$$

Finally, the L^2 -norm of P_k is given by:

$$\|P_k\|_2 = \sqrt{\langle P_k, P_k \rangle} = \sqrt{\frac{2}{2k+1}} \quad (\text{C.8})$$

where one should note that by construction of Legendre polynomials: $\langle P_k, P_l \rangle = 0$ for $k \neq l$.

C.2.5. Shifted Legendre polynomials

Shifted Legendre polynomials correspond to Legendre polynomials defined on $[0, 1]$ rather than on $[-1, 1]$ as introduced above. It is recalled here for clarity since usually Legendre polynomials are introduced on $[-1, 1]$ but the SD method presented in this manuscript work on $[0, 1]$. The transformation

$$\xi(\tilde{\xi}) = 2\tilde{\xi} - 1 \iff \tilde{\xi}(\xi) = \frac{1+\xi}{2} \quad (\text{C.9})$$

with $\xi \in [-1, 1]$ and $\tilde{\xi} \in [0, 1]$ bijectly maps $[0, 1]$ into $[-1, 1]$ and vice-versa so that $P_k(\xi)$ can be evaluated from $\tilde{\xi}$ values using

$$P_k(\xi) = P_k(2\tilde{\xi} - 1) \quad (\text{C.10})$$

Therefore, the recurrence formula written on $[-1, 1]$ in paragraph C.2.3, can be used on $[0, 1]$ with $\tilde{\xi}$:

$$(k+1)P_{k+1}(2\tilde{\xi}-1) = (2k+1)(2\tilde{\xi}-1)P_k(2\tilde{\xi}-1) - kP_{k-1}(2\tilde{\xi}-1) \text{ for } k \geq 2$$

$$P_0 = 1 \text{ and } P_1 = 2\tilde{\xi} - 1 \quad (\text{C.11})$$

Eq. (C.9) also shows how to compute Gauss-Legendre points on $[0, 1]$ from their values on $[-1, 1]$. It is very useful for locating interior FP on $[0, 1]$ for the SD method as they correspond to these Gauss-Legendre points (see paragraph 4.1.2.2). Finally, the weights of integration given in Eq. (C.4) are simply divided by two if the quadrature rule is used with Shifted Legendre polynomials. It is due to the change of variable from the original integral done on $[-1, 1]$ to the new integral that will be considered on $[0, 1]$:

$$\int_{-1}^1 h(\xi) d\xi = \sum_{i=1}^k \omega_i h(\xi_i) = 2 \int_0^1 h(\tilde{\xi}) d\tilde{\xi} = 2 \sum_{i=1}^k \tilde{\omega}_i h(\tilde{\xi}_i)$$

$$\implies \tilde{\omega}_i = \frac{\omega_i}{2} \quad (\text{C.12})$$

C.3. Radau polynomials

Radau polynomials are used in some correction functions employed in the FR framework and also for gradient computation in the SD context in this work. That is why there are considered in this section. The notations follow the ones used by Huynh in his original FR paper [57].

C.3.1. Definition of Radau points based on a quadrature rule

Radau polynomials come from the Radau quadrature rule that uses a set of $k \geq 1$ points, where one of them is one of the two boundary points ($\xi = -1$ or $\xi = 1$), for the integration of a function on $I = [-1, 1]$. Thus, this quadrature rule is different from the Gauss-Legendre one where all the Gauss-Legendre points are strictly inside I as shown in paragraph C.2.3. If the right boundary is include in the quadrature points, meaning that $\xi_k = 1$, the right Radau quadrature rule can be written for integrating any smooth function h on I with [57]:

$$\int_{-1}^1 h(\xi) d\xi \approx \sum_{i=1}^k \omega_i h(\xi_i) \text{ where } \xi_k = 1 \quad (\text{C.13})$$

Thus, there are k weights and $k-1$ integration points to find. Consequently, a total of $2k-1$ unknowns have to be found which is solved by setting $h = \xi^i$, for $i = 0, 1, \dots, 2k-2$, in Eq. (C.13). The $k-1$ points obtained solving this system plus $\xi_k = 1$ are called the right Radau points [57]. Similarly, the left Radau points and quadrature are defined in the same manner but the right boundary point is replaced by the left one. The left Radau points are actually the symmetry points about 0 of the Right Radau points [57]. Both these quadrature rules using k points can integrate exactly a polynomial of degree $2k-2$ or less (one degree less than Gauss-Legendre quadrature rule).

C.3.2. Expressions of Radau polynomials

The right (respectively left) k Radau points are the roots of right (respectively left) Radau polynomial $R_{R,k}$ (respectively $R_{L,k}$) of degree k . It can be shown that Radau polynomials can be expressed using Legendre polynomials [57]:

$$R_{R,k} = \frac{(-1)^k}{2} (P_k - P_{k-1}) \quad (\text{C.14})$$

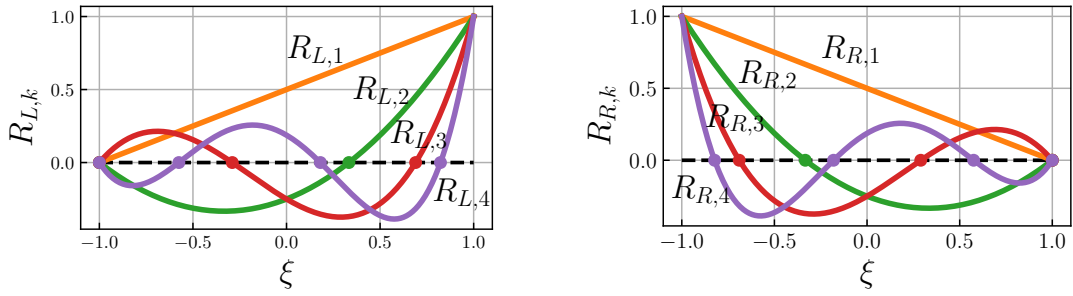
$$R_{R,k}(-1) = 1 \text{ and } R_{R,k}(1) = 0 \quad (\text{C.15})$$

and for the left Radau polynomial [57]:

$$R_{L,k} = \frac{P_k + P_{k-1}}{2} \quad (\text{C.16})$$

$$R_{L,k}(-1) = 0 \text{ and } R_{L,k}(1) = 1 \quad (\text{C.17})$$

Therefore, according to Eq. (C.14) and Eq. (C.16), Radau polynomials are polynomials of degree k which are orthogonal to \mathbf{P}^{k-2} space because of the presence of P_{k-1} in their expressions. Left and right Radau polynomials, along with their corresponding Radau points, up to $1 \leq k \leq 4$ are represented in Figures C.2a and C.2b.



(a) Left Radau polynomials and points

(b) Right Radau polynomials and points

Figure C.2. – Radau polynomials and points on $I = [-1, 1]$ for $1 \leq k \leq 4$.

Projection matrices for local polynomial adaptation

D.1. Projection matrices in 2D

D.1.1. Position of the problem

In 2D, the mortar element is a 1D line. Without loss of generality, an interface at constant ξ between two elements is considered as in Figure 4.16 with a left element at p_L and a right element at $p_R \neq p_L$. The process will be described here for the convective fluxes treatment at an interface since there are only two projection steps (one on Mortar and one from Mortar) whereas there are four projection steps (two on Mortar and two from Mortar) for the diffusive fluxes treatment. For the left (respectively right) element, there are $p_L + 1$ (respectively $p_R + 1$) FP on this interface which are located at the same ξ value but have different η coordinates. In each element, these η coordinates of interface FP are the same as the η coordinates of the SP within this element. Consequently, a 1D continuous polynomial of degree p_L (respectively p_R) along η direction can be built using the values of the solution at these interface FP and the Lagrange polynomial basis built at SP along η direction:

$$\mathbf{U}^{\Omega_L}(\eta) = \sum_{j_2=1}^{p_L+1} \mathbf{U}_{j_2}^{\Omega_L} l_{j_2}^{\Omega_L}(\eta), \quad \eta \in [0, 1] \quad (\text{D.1})$$

$$\mathbf{U}^{\Omega_R}(\eta) = \sum_{j_2=1}^{p_R+1} \mathbf{U}_{j_2}^{\Omega_R} l_{j_2}^{\Omega_R}(\eta), \quad \eta \in [0, 1] \quad (\text{D.2})$$

where $\mathbf{U}_{j_2}^{\Omega_L}$ (respectively $\mathbf{U}_{j_2}^{\Omega_R}$) are the values of the conservative variables in the interface at constant ξ in element Ω_L (respectively Ω_R). It is worth mentioning that since $p_L \neq p_R$, the Lagrange polynomials are necessary different: $l_{j_2}^{\Omega_L} \neq l_{j_2}^{\Omega_R}$.

D.1.2. Projection from the face to the mortar element

As mentioned in paragraph 4.5.4, the Riemann problem is solved on the Mortar interface between a left state $\mathbf{U}^{M,L}$ and a right state $\mathbf{U}^{M,R}$ both defined as:

$$\mathbf{U}^{M,L}(\eta) = \sum_{j_2=1}^{p_M+1} \mathbf{U}_{j_2}^{M,L} l_{j_2}^M(\eta), \quad \eta \in [0, 1] \quad (\text{D.3})$$

$$\mathbf{U}^{M,R}(\eta) = \sum_{j_2=1}^{p_M+1} \mathbf{U}_{j_2}^{M,R} l_{j_2}^M(\eta), \quad \eta \in [0, 1] \quad (\text{D.4})$$

where $p_M = \max(p_L, p_R)$ is the polynomial degree on the Mortar element and the $l_{j_2}^M$ are the Lagrange polynomials built at Mortar points. Still assuming that $p_L < p_R$, it can be written that

- $p_M = p_R$ and $l_{j_2}^M = l_{j_2}^{\Omega_R} \quad \forall j_2 \in \llbracket 1, p_M + 1 \rrbracket$.

- Right Mortar solution is equal to the solution on the right element interface: $\mathbf{U}_{j_2}^{M,R} = \mathbf{U}_{j_2}^{\Omega_R} \forall j_2 \in \llbracket 1, p_M + 1 \rrbracket$.
- Left Mortar solution is given by the unweighted L^2 -projection of \mathbf{U}^{Ω_L} onto the Mortar which reads as:

$$\begin{aligned}
 & \int_0^1 \left(\mathbf{U}^{\Omega_L}(\eta) - \mathbf{U}^{M,L}(\eta) \right) l_m^M(\eta) d\eta = \mathbf{0} \text{ for } m \in \llbracket 1, p_M + 1 \rrbracket \\
 & \implies \int_0^1 \sum_{j_2=1}^{p_L+1} \mathbf{U}_{j_2}^{\Omega_L} l_{j_2}^{\Omega_L}(\eta) l_m^M(\eta) d\eta = \int_0^1 \sum_{j_2=1}^{p_M+1} \mathbf{U}_{j_2}^{M,L} l_{j_2}^M(\eta) l_m^M(\eta) d\eta \text{ for } m \in \llbracket 1, p_M + 1 \rrbracket \\
 & \implies \sum_{j_2=1}^{p_L+1} \mathbf{U}_{j_2}^{\Omega_L} \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^M(\eta) d\eta = \sum_{j_2=1}^{p_M+1} \mathbf{U}_{j_2}^{M,L} \int_0^1 l_{j_2}^M(\eta) l_m^M(\eta) d\eta \text{ for } m \in \llbracket 1, p_M + 1 \rrbracket
 \end{aligned} \tag{D.5}$$

Eq. (D.5) is a system of $p_M + 1$ equations that is used to determine the $\mathbf{U}_{j_2}^{M,L}$ for $j_2 \in \llbracket 1, p_M + 1 \rrbracket$. Introducing rectangular matrix Σ^L and square matrix Π :

$$\Sigma_{m,j_2}^L = \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^M(\eta) d\eta \text{ for } (m, j_2) \in \llbracket 1, p_M + 1 \rrbracket \times \llbracket 1, p_L + 1 \rrbracket \tag{D.6}$$

$$\Pi_{m,j_2} = \int_0^1 l_{j_2}^M(\eta) l_m^M(\eta) d\eta \text{ for } (m, j_2) \in \llbracket 1, p_M + 1 \rrbracket^2 \tag{D.7}$$

Eq. (D.5) can be written in matrix form:

$$\begin{aligned}
 & \Sigma^L \cdot \mathbf{U}^{\Omega_L} = \Pi \cdot \mathbf{U}^{M,L} \\
 & \implies \mathbf{U}^{M,L} = \Pi^{-1} \cdot \Sigma^L \cdot \mathbf{U}^{\Omega_L}
 \end{aligned} \tag{D.8}$$

D.1.3. Back projection from the mortar element to the left face

Once $\mathbf{U}^{M,L}$ has been obtained, a Riemann problem is solved that ends up with the convective fluxes on the Mortar FP $\mathcal{F}_{c,n}^M$ and this convective flux is projected back on both elements. For the right element $\mathcal{F}_{c,n}^{\Omega_R} = \mathcal{F}_{c,n}^M$ since $p_M = p_R$. For the left element, a back unweighted L^2 -projection is done to find $\mathcal{F}_{c,n}^{\Omega_L}$:

$$\begin{aligned}
 & \int_0^1 \left(\mathcal{F}_{c,n}^{\Omega_L}(\eta) - \mathcal{F}_{c,n}^M(\eta) \right) l_m^{\Omega_L}(\eta) d\eta = \mathbf{0} \text{ for } m \in \llbracket 1, p_L + 1 \rrbracket \\
 & \implies \sum_{j_2=1}^{p_L+1} \mathcal{F}_{c,n,j_2}^{\Omega_L} \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^{\Omega_L}(\eta) d\eta = \sum_{j_2=1}^{p_M+1} \mathcal{F}_{c,n,j_2}^M \int_0^1 l_{j_2}^M(\eta) l_m^{\Omega_L}(\eta) d\eta \text{ for } m \in \llbracket 1, p_L + 1 \rrbracket
 \end{aligned} \tag{D.9}$$

Eq. (D.9) is a system of $p_L + 1$ equations that is used to determine the $\mathcal{F}_{c,n,j_2}^{\Omega_L}$ for $j_2 \in \llbracket 1, p_L + 1 \rrbracket$. Introducing square matrix Π^L and rectangular matrix Σ :

$$\Pi_{m,j_2}^L = \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^{\Omega_L}(\eta) d\eta \text{ for } (m, j_2) \in \llbracket 1, p_L + 1 \rrbracket^2 \tag{D.10}$$

$$\Sigma_{m,j_2} = \int_0^1 l_{j_2}^M(\eta) l_m^{\Omega_L}(\eta) d\eta \text{ for } (m, j_2) \in \llbracket 1, p_L + 1 \rrbracket \times \llbracket 1, p_M + 1 \rrbracket \tag{D.11}$$

Eq. (D.9) can be written in matrix form and solved for $\mathcal{F}_{c,n}^{\Omega_L}$:

$$\mathcal{F}_{c,n}^{\Omega_L} = \left(\Pi^L \right)^{-1} \cdot \Sigma \cdot \mathcal{F}_{c,n}^M \tag{D.12}$$

In Eq. (D.8) and Eq. (D.12), all matrices are calculated once in pre-processing using Gauss-Legendre quadrature rule at Gauss-Legendre points in $[0, 1]$ for computing integrals. Actually by construction, these Gauss-Legendre points are the locations of the p -interior FP on a given element of degree p so that integration points are already available for the evaluations of integrals in Eq. (D.8) and Eq. (D.12). Since minimum and maximum polynomial order are set by the user, all matrices are stored for $p_{min} \leq p_L, p_M \leq p_{max}$ to have all possible Mortar treatments that can happen between two elements whose degrees are between p_{min} and p_{max} . If an interface at constant η is considered, the same projection matrices are obtained since Legendre points are located at the same positions along ξ direction. Moreover, the process was described here for the convective fluxes where the solution on left element is first projected on the Mortar and then the Riemann flux is projected back on left element but for the diffusive fluxes where there are one more projection on Mortar and one more projection from the Mortar, the same matrices are employed.

D.2. Projection matrices in 3D

D.2.1. Position of the problem

In 3D, the mortar element is a 2D interface and again without loss of generality, it is assumed to be located at a constant ξ position between two elements with different degrees p_L and p_R still with $p_L < p_R$. Again the process is described for the convective fluxes treatment as in the 2D case for clarity. For the left (respectively right) element, there are $(p_L + 1)^2$ (respectively $(p_R + 1)^2$) FP on this interface which are located at the same ξ value but have different η and ζ coordinates. In each element, these η and ζ coordinates of interface FP are the same as the η and ζ coordinates of the SP within this element. Consequently, a continuous polynomial of degree p_L (respectively p_R) along η and ζ directions can be built using the values of the solution at these interface FP and the Lagrange polynomial basis built at SP along η and ζ directions:

$$\mathbf{U}^{\Omega_L}(\eta, \zeta) = \sum_{j_2=1}^{p_L+1} \sum_{j_3=1}^{p_L+1} \mathbf{U}_{(j_2, j_3)}^{\Omega_L} l_{j_2}^{\Omega_L}(\eta) l_{j_3}^{\Omega_L}(\zeta), \quad (\eta, \zeta) \in [0, 1]^2 \quad (\text{D.13})$$

$$\mathbf{U}^{\Omega_R}(\eta, \zeta) = \sum_{j_2=1}^{p_R+1} \sum_{j_3=1}^{p_R+1} \mathbf{U}_{(j_2, j_3)}^{\Omega_R} l_{j_2}^{\Omega_R}(\eta) l_{j_3}^{\Omega_R}(\zeta), \quad (\eta, \zeta) \in [0, 1]^2 \quad (\text{D.14})$$

The left and right solutions on the Mortar element are now defined by:

$$\mathbf{U}^{M,L}(\eta, \zeta) = \sum_{j_2=1}^{p_M+1} \sum_{j_3=1}^{p_M+1} \mathbf{U}_{(j_2, j_3)}^{M,L} l_{j_2}^M(\eta) l_{j_3}^M(\zeta), \quad (\eta, \zeta) \in [0, 1]^2 \quad (\text{D.15})$$

$$\mathbf{U}^{M,R}(\eta, \zeta) = \sum_{j_2=1}^{p_M+1} \sum_{j_3=1}^{p_M+1} \mathbf{U}_{(j_2, j_3)}^{M,R} l_{j_2}^M(\eta) l_{j_3}^M(\zeta), \quad (\eta, \zeta) \in [0, 1]^2 \quad (\text{D.16})$$

D.2.2. Projection from the left face to the mortar element

As in the 2D case for $p_L < p_R$, $\mathbf{U}_{(j_2, j_3)}^{M,R} = \mathbf{U}_{(j_2, j_3)}^{\Omega_R} \forall (j_2, j_3) \in \llbracket 1, p_M + 1 \rrbracket^2$ and the left Mortar solution is again obtained through an unweighted L^2 -projection of \mathbf{U}^{Ω_L} onto the Mortar given here

by:

$$\begin{aligned}
 & \int_0^1 \int_0^1 \left(\mathbf{U}^{\Omega_L}(\eta, \zeta) - \mathbf{U}^{M,L}(\eta, \zeta) \right) l_m^M(\eta) l_n^M(\zeta) d\eta d\zeta = \mathbf{0} \text{ for } (m, n) \in \llbracket 1, p_M + 1 \rrbracket^2 \\
 \Rightarrow & \int_0^1 \int_0^1 \left[\sum_{j_2=1}^{p_L+1} \sum_{j_3=1}^{p_L+1} \mathbf{U}_{(j_2, j_3)}^{\Omega_L} l_{j_2}^{\Omega_L}(\eta) l_{j_3}^{\Omega_L}(\zeta) \right] l_m^M(\eta) l_n^M(\zeta) d\eta d\zeta = \\
 & \int_0^1 \int_0^1 \left[\sum_{j_2=1}^{p_M+1} \sum_{j_3=1}^{p_M+1} \mathbf{U}_{(j_2, j_3)}^{M,L} l_{j_2}^M(\eta) l_{j_3}^M(\zeta) \right] l_m^M(\eta) l_n^M(\zeta) d\eta d\zeta \text{ for } (m, n) \in \llbracket 1, p_M + 1 \rrbracket^2 \\
 \Rightarrow & \sum_{j_2=1}^{p_L+1} \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^M(\eta) d\eta \sum_{j_3=1}^{p_L+1} \mathbf{U}_{(j_2, j_3)}^{\Omega_L} \int_0^1 l_{j_3}^{\Omega_L}(\zeta) l_n^M(\zeta) d\zeta = \\
 & \sum_{j_2=1}^{p_M+1} \int_0^1 l_{j_2}^M(\eta) l_m^M(\eta) d\eta \sum_{j_3=1}^{p_M+1} \mathbf{U}_{(j_2, j_3)}^{M,L} \int_0^1 l_{j_3}^M(\zeta) l_n^M(\zeta) d\zeta \text{ for } (m, n) \in \llbracket 1, p_M + 1 \rrbracket^2
 \end{aligned} \tag{D.17}$$

Eq. (D.17) is a system of $(p_M + 1)^2$ equations that is used to determine the $\mathbf{U}_{(j_2, j_3)}^{M,L}$ for $(j_2, j_3) \in \llbracket 1, p_M + 1 \rrbracket^2$. There are two more matrices, compared to the 2D case, that can be introduced since the two integrals in η correspond respectively to Σ^L and Π defined in Eq. (D.6) and Eq. (D.7) in the 2D case. The two new matrices needed in 3D are:

$$\Theta_{j_3, n}^L = \int_0^1 l_{j_3}^{\Omega_L}(\zeta) l_n^M(\zeta) d\zeta \text{ for } (j_3, n) \in \llbracket 1, p_L + 1 \rrbracket \times \llbracket 1, p_M + 1 \rrbracket \tag{D.18}$$

$$\Xi_{j_3, n} = \int_0^1 l_{j_3}^M(\zeta) l_n^M(\zeta) d\zeta \text{ for } (j_3, n) \in \llbracket 1, p_M + 1 \rrbracket^2 \tag{D.19}$$

Using Σ^L , Π , Θ^L and Ξ , Eq. (D.17) can be written in matrix form:

$$\begin{aligned}
 \Sigma^L \cdot \mathbf{U}^{\Omega_L} \cdot \Theta^L &= \Pi \cdot \mathbf{U}^{M,L} \cdot \Xi \\
 \Rightarrow \mathbf{U}^{M,L} &= \Pi^{-1} \cdot \Sigma^L \cdot \mathbf{U}^{\Omega_L} \cdot \Theta^L \cdot \Xi^{-1}
 \end{aligned} \tag{D.20}$$

Since the integration process is the same for η and ζ directions, it can be noticed that:

$$\Theta^L = \left[\Sigma^L \right]^T \text{ and } \Xi = \Pi \tag{D.21}$$

so that $\Theta^L \cdot \Xi^{-1}$ in Eq. (D.20) is actually:

$$\Theta^L \cdot \Xi^{-1} = \left[\Sigma^L \right]^T \cdot \Pi^{-1} = \left[\Sigma^L \right]^T \cdot \left[\Pi^{-1} \right]^T = \left[\Pi^{-1} \cdot \Sigma^L \right]^T \tag{D.22}$$

where the second equality comes from the fact that Π is a symmetric matrix. Therefore, Eq. (D.20) is computed using:

$$\mathbf{U}^{M,L} = \Pi^{-1} \cdot \Sigma^L \cdot \mathbf{U}^{\Omega_L} \cdot \left[\Pi^{-1} \cdot \Sigma^L \right]^T \tag{D.23}$$

D.2.3. Back projection from the mortar element to the left face

Like in the 2D case, when $\mathbf{U}^{M,L}$ has been obtained, a Riemann problem is solved that ends up with the convective fluxes on the Mortar FP $\mathcal{F}_{c,n}^M$ and this convective flux is projected back on both

elements. For the right element the projection is still identity: $\mathcal{F}_{c,n}^{\Omega_R} = \mathcal{F}_{c,n}^M$ since $p_M = p_R$. For the left element, a back unweighted L^2 -projection is again used to find $\mathcal{F}_{c,n}^{\Omega_L}$:

$$\begin{aligned} & \int_0^1 \int_0^1 \left(\mathcal{F}_{c,n}^{\Omega_L}(\eta, \zeta) - \mathcal{F}_{c,n}^M(\eta, \zeta) \right) l_m^{\Omega_L}(\eta) l_n^{\Omega_L}(\zeta) d\eta d\zeta = \mathbf{0} \text{ for } (m, n) \in \llbracket 1, p_L + 1 \rrbracket^2 \\ \Rightarrow & \sum_{j_2=1}^{p_L+1} \int_0^1 l_{j_2}^{\Omega_L}(\eta) l_m^{\Omega_L}(\eta) d\eta \sum_{j_3=1}^{p_L+1} \mathcal{F}_{c,n,(j_2,j_3)}^{\Omega_L} \int_0^1 l_{j_3}^{\Omega_L}(\zeta) l_n^{\Omega_L}(\zeta) d\zeta = \\ & \sum_{j_2=1}^{p_M+1} \int_0^1 l_{j_2}^M(\eta) l_m^{\Omega_L}(\eta) d\eta \sum_{j_3=1}^{p_M+1} \mathcal{F}_{c,n,(j_2,j_3)}^M \int_0^1 l_{j_3}^M(\zeta) l_n^{\Omega_L}(\zeta) d\zeta \text{ for } (m, n) \in \llbracket 1, p_L + 1 \rrbracket^2 \end{aligned} \quad (\text{D.24})$$

Eq. (D.24) is a system of $(p_L + 1)^2$ equations that is used to determine the $\mathcal{F}_{c,n,(j_2,j_3)}^{\Omega_L}$ for $(j_2, j_3) \in \llbracket 1, p_L + 1 \rrbracket^2$. As for the projection on Mortar, there are two more matrices compared to the 2D case that can be introduced in addition to Σ and Π^L defined in Eq. (D.11) and Eq. (D.10) in the 2D case. The two new matrices needed in 3D are:

$$\Xi_{j_3,n}^L = \int_0^1 l_{j_3}^{\Omega_L}(\zeta) l_n^{\Omega_L}(\zeta) d\zeta \text{ for } (j_3, n) \in \llbracket 1, p_L + 1 \rrbracket^2 \quad (\text{D.25})$$

$$\Theta_{j_3,n} = \int_0^1 l_{j_3}^M(\zeta) l_n^{\Omega_L}(\zeta) d\zeta \text{ for } (j_3, n) \in \llbracket 1, p_M + 1 \rrbracket \times \llbracket 1, p_L + 1 \rrbracket \quad (\text{D.26})$$

with also:

$$\Theta = \Sigma^T \text{ and } \Xi^L = \Pi^L \quad (\text{D.27})$$

Consequently, the matrix form of Eq. (D.24) is:

$$\begin{aligned} \Pi^L \cdot \mathcal{F}_{c,n}^{\Omega_L} \cdot \Pi^L &= \Sigma \cdot \mathcal{F}_{c,n}^M \cdot \Sigma^T \\ \Rightarrow \mathcal{F}_{c,n}^{\Omega_L} &= \left(\Pi^L \right)^{-1} \cdot \Sigma \cdot \mathcal{F}_{c,n}^M \cdot \left[\left(\Pi^L \right)^{-1} \cdot \Sigma \right]^T \end{aligned} \quad (\text{D.28})$$

Again as in 2D, the matrices appearing in Eq. (D.23) and Eq. (D.28) are computed once in pre-processing using Gauss-Legendre quadrature at Gauss points in $[0, 1]$. The main difference between the 2D case is that the transpose of matrices $\Pi^{-1} \cdot \Sigma^L$, for Eq. (D.23), and of $\left(\Pi^L \right)^{-1} \cdot \Sigma$, for Eq. (D.28), are also needed.

Coefficients of Runge-Kutta schemes used in this work

E.1. SSP-ERK(3,3) scheme of Gottlieb and Shu

The Shu-Osher matrices for this scheme are summed in Table E.1 [222].

| α_{qj} | β_{qj} |
|---|---|
| $\begin{pmatrix} 1 & 0 & 0 \\ 3/4 & 1/4 & 0 \\ 1/3 & 0 & 2/3 \end{pmatrix}$ | $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/4 & 0 \\ 0 & 0 & 2/3 \end{pmatrix}$ |

Table E.1. – Coefficients of matrices α and β of the SSP-ERK(3,3) scheme.

The equivalent Butcher tableau, obtained from its Shu-Osher representation, for this scheme is written in Table E.2.

| | |
|---|---|
| $\begin{pmatrix} 0 \\ 1 \\ 0.5 \end{pmatrix}$ | $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1/4 & 0 \\ 0 & 0 & 2/3 \end{pmatrix}$ |
| | $(1/6, 1/6, 2/3)^T$ |

Table E.2. – Butcher tableau of the SSP-ERK(3,3) scheme.

E.2. SSP-ERK(5,4) scheme of Spiteri and Ruuth

The Shu-Osher matrices for this scheme are summed in Table E.3 [226].

| | |
|---|--|
| $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.44437049406734 & 0.55562950593266 & 0 & 0 & 0 \\ 0.62010185138540 & 0 & 0.37989814861460 & 0 & 0 \\ 0.17807995410773 & 0 & 0 & 0.82192004589227 & 0 \\ 0.00683325884039 & 0 & 0.51723167208978 & 0.12759831133288 & 0.34833675773694 \end{pmatrix}$ | |
| $\begin{pmatrix} 0.39175222700392 & 0 & 0 & 0 & 0 \\ 0 & 0.36841059262959 & 0 & 0 & 0 \\ 0 & 0 & 0.25189177424738 & 0 & 0 \\ 0 & 0 & 0 & 0.54497475021237 & 0 \\ 0 & 0 & 0 & 0.08460416338212 & 0.22600748319395 \end{pmatrix}$ | |

Table E.3. – Coefficients of matrices α (top) and β (bottom) of the SSP-ERK(5,4) scheme.

The equivalent Butcher tableau, obtained from its Shu-Osher representation, for this scheme is written in Table E.4.

| | |
|---|--|
| $\begin{pmatrix} 0 \\ 0.39175223 \\ 0.58607969 \\ 0.47454236 \\ 0.93501063 \end{pmatrix}$ | $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.391752230 & 0 & 0 & 0 & \\ 0.2176691 & 0.36841059 & 0 & 0 & 0 \\ 0.08269209 & 0.1399585 & 0.25189177 & 0 & 0 \\ 0.06796628 & 0.1150347 & 0.2070349 & 0.54497475 & 0 \end{pmatrix}$ |
| | $(0.14681188, 0.24848291, 0.10425883, 0.2744389, 0.22600748)^T$ |

Table E.4. – Butcher tableau of the SSP-ERK(5,4) scheme.

Diagonalization of the 3D Euler equations for a multi-species gas in isoparametric coordinates

F.1. Governing equations for conservative variables in the physical space

The 3D Euler equations for a multi-species gas composed of N_s species can be written as:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}_c}{\partial x} + \frac{\partial \mathbf{F}_c}{\partial y} + \frac{\partial \mathbf{G}_c}{\partial z} = \mathbf{0} \quad (\text{F.1})$$

where $\mathbf{U} = (\rho, \rho u, \rho v, \rho w, \rho E, \rho Y_k)^T = (U_1, U_2, U_3, U_4, U_5, U_{5+k})^T$, with $k \in \llbracket 1, N_s \rrbracket$, is the vector of conservative variables and the convective fluxes as a function of \mathbf{U} are given by:

$$\begin{aligned} \mathbf{E}_c &= \begin{pmatrix} \rho u \\ P + \rho u^2 \\ \rho uv \\ \rho uw \\ u(P + \rho E) \\ \rho u Y_k \end{pmatrix} = \begin{pmatrix} U_2 \\ P + U_2^2/U_1 \\ U_2 U_3/U_1 \\ U_2 U_4/U_1 \\ U_2 (P + U_5)/U_1 \\ U_{5+k} U_2/U_1 \end{pmatrix}, \quad \mathbf{F}_c = \begin{pmatrix} \rho v \\ \rho uv \\ P + \rho v^2 \\ \rho vw \\ v(P + \rho E) \\ \rho v Y_k \end{pmatrix} = \begin{pmatrix} U_3 \\ U_2 U_3/U_1 \\ P + U_3^2/U_1 \\ U_3 U_4/U_1 \\ U_3 (P + U_5)/U_1 \\ U_{5+k} U_3/U_1 \end{pmatrix} \\ \mathbf{G}_c &= \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ P + \rho w^2 \\ w(P + \rho E) \\ \rho w Y_k \end{pmatrix} = \begin{pmatrix} U_4 \\ U_2 U_4/U_1 \\ U_3 U_4/U_1 \\ P + U_4^2/U_1 \\ U_4 (P + U_5)/U_1 \\ U_{5+k} U_4/U_1 \end{pmatrix} \end{aligned} \quad (\text{F.2})$$

ρE is the sum of sensible and kinetic energies per unit of volume whose expression for a thermally perfect gas (C_p and C_v depend on T) is:

$$\rho E = \rho \sum_{k=1}^{N_s} Y_k \left(\int_{T_0}^T C_{pk}(T') dT' \right) - P + \rho \frac{u^2 + v^2 + w^2}{2} = \rho \sum_{k=1}^{N_s} Y_k h_{sk}(T) - P + \rho \frac{u^2 + v^2 + w^2}{2} \quad (\text{F.3})$$

where the pressure is given by the perfect gas law introduced in Eq. (2.16).

F.2. Governing equations for primitive variables in the physical space

The diagonalization process is easier to do when working with primitive variables $\mathbf{Q} = (\rho, u, v, w, P, Y_k)^T = (Q_1, Q_2, Q_3, Q_4, Q_5, Q_{5+k})^T$. Applying the chain rule with respect to \mathbf{Q} to Eq. (F.1) allows to write:

$$\begin{aligned} & \frac{\partial \mathbf{U}}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{E}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial x} + \frac{\partial \mathbf{F}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial y} + \frac{\partial \mathbf{G}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial z} = \mathbf{0} \\ \Rightarrow & \frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{E}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial x} + \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{F}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial y} + \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{G}_c}{\partial \mathbf{Q}} \cdot \frac{\partial \mathbf{Q}}{\partial z} = \mathbf{0} \\ & \Rightarrow \frac{\partial \mathbf{Q}}{\partial t} + A \frac{\partial \mathbf{Q}}{\partial x} + B \frac{\partial \mathbf{Q}}{\partial y} + C \frac{\partial \mathbf{Q}}{\partial z} = \mathbf{0} \end{aligned} \quad (\text{F.4})$$

where $A = (\partial \mathbf{Q} / \partial \mathbf{U}) \cdot (\partial \mathbf{E}_c / \partial \mathbf{Q})$, $B = (\partial \mathbf{Q} / \partial \mathbf{U}) \cdot (\partial \mathbf{F}_c / \partial \mathbf{Q})$ and $C = (\partial \mathbf{Q} / \partial \mathbf{U}) \cdot (\partial \mathbf{G}_c / \partial \mathbf{Q})$ are three Jacobian matrices that have to be determined. Firstly, the passage matrix from conservative to primitive $\partial \mathbf{Q} / \partial \mathbf{U}$ is computed by inverting $\partial \mathbf{U} / \partial \mathbf{Q}$ (passage matrix from primitive to conservative) which is easier to obtain. Secondly, flux Jacobian matrices with respect to \mathbf{Q} namely $\partial \mathbf{E}_c / \partial \mathbf{Q}$, $\partial \mathbf{F}_c / \partial \mathbf{Q}$ and $\partial \mathbf{G}_c / \partial \mathbf{Q}$ are computed. Finally expressions of A , B and C are obtained.

F.3. Computation of Jacobian matrices A , B and C

F.3.1. Computation of $\partial \mathbf{U} / \partial \mathbf{Q}$ matrix

This matrix is defined by:

$$\frac{\partial \mathbf{U}}{\partial \mathbf{Q}} \equiv \frac{\partial (\rho, \rho u, \rho v, \rho w, \rho E, \rho Y_k)}{\partial (\rho, u, v, w, P, Y_k)} \quad (\text{F.5})$$

The four first lines and the $5 + k$ lines, with $k \in \llbracket 1, N_s \rrbracket$, can be easily computed: the only difficulties rely in the fifth line for the terms $\partial \rho E / \partial \rho$, $\partial \rho E / \partial P$ and $\partial \rho E / \partial Y_k$.

F.3.1.1. Computation of $\partial \rho E / \partial \rho$ for $\partial \mathbf{U} / \partial \mathbf{Q}$ matrix

Differentiating Eq. (F.3) with respect to ρ gives:

$$\begin{aligned} \frac{\partial \rho E}{\partial \rho} &= \underbrace{\sum_{k=1}^{N_s} Y_k h_{sk}(T)}_{h_s(T)} + \rho \sum_{k=1}^{N_s} Y_k C_{pk} \underbrace{\frac{\partial T}{\partial \rho}}_{\frac{-T}{\rho}} - \underbrace{\frac{\bar{R}}{W} \left(T + \rho \frac{\partial T}{\partial \rho} \right)}_{=0} + \underbrace{\frac{u^2 + v^2 + w^2}{2}}_{\frac{\|\mathbf{u}\|_2^2}{2}} \\ \Rightarrow \frac{\partial \rho E}{\partial \rho} &= h_s(T) - T \underbrace{\sum_{k=1}^{N_s} Y_k C_{pk}}_{C_p(T)} + \frac{\|\mathbf{u}\|_2^2}{2} \\ \Rightarrow \frac{\partial \rho E}{\partial \rho} &= h_s(T) - T C_p(T) + \frac{\|\mathbf{u}\|_2^2}{2} \end{aligned} \quad (\text{F.6})$$

F.3.1.2. Computation of $\partial \rho E / \partial P$ for $\partial \mathbf{U} / \partial \mathbf{Q}$ matrix

In that case, Eq. (F.3) is differentiated with respect to P to get:

$$\frac{\partial \rho E}{\partial P} = \rho \sum_{k=1}^{N_s} Y_k C_{pk} \underbrace{\frac{\partial T}{\partial P}}_{\frac{W}{\rho R}} - 1 = \frac{W}{R} C_p(T) - 1 = \frac{C_p(T)}{C_p(T) - C_v(T)} - 1 = \frac{1}{1 - \frac{1}{\gamma}} - 1 = \frac{\gamma - \gamma + 1}{\gamma - 1} = \frac{1}{\gamma - 1} \quad (\text{F.7})$$

which is the same results for $\partial \rho E / \partial P$ when the gas is calorically perfect (constant C_p and C_v).

F.3.1.3. Computation of $\partial \rho E / \partial Y_k$ for $\partial \mathbf{U} / \partial \mathbf{Q}$ matrix

In that case, Eq. (F.3) is differentiated with respect to Y_k to get:

$$\begin{aligned} \frac{\partial \rho E}{\partial Y_k} &= \rho \left(h_{sk}(T) + \sum_{k'=1}^{N_s} Y_{k'} C_{pk'} \underbrace{\frac{\partial T}{\partial Y_k}}_{\frac{-TW}{W_k}} \right) - \rho \bar{R} \underbrace{\left(T \underbrace{\frac{\partial \left(\frac{1}{W} \right)}{\partial Y_k}}_{\frac{1}{W_k}} + \frac{1}{W} \underbrace{\frac{\partial T}{\partial Y_k}}_{\frac{-TW}{W_k}} \right)}_{=0} \\ \Rightarrow \frac{\partial \rho E}{\partial Y_k} &= \rho \left(h_{sk}(T) - TC_p(T) \frac{W}{W_k} \right) \end{aligned} \quad (\text{F.8})$$

F.3.1.4. Final expression for $\partial \mathbf{U} / \partial \mathbf{Q}$ matrix

Following these previous calculations, the passage matrix from primitive to conservative variables $\partial \mathbf{U} / \partial \mathbf{Q}$ for a multi-species thermally perfect gas is given by:

$$\frac{\partial \mathbf{U}}{\partial \mathbf{Q}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 & 0 \\ v & 0 & \rho & 0 & 0 & 0 \\ w & 0 & 0 & \rho & 0 & 0 \\ h_s - TC_p + \frac{\|\mathbf{u}\|_2^2}{2} & \rho u & \rho v & \rho w & \frac{1}{\gamma - 1} & \rho \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ Y_k & 0 & 0 & 0 & 0 & \rho \end{bmatrix} \quad (\text{F.9})$$

F.3.2. Computation of $\partial \mathbf{Q} / \partial \mathbf{U}$ matrix

Matrix $\partial \mathbf{Q} / \partial \mathbf{U}$ is the inverse matrix of $\partial \mathbf{U} / \partial \mathbf{Q}$ introduced in Eq. (F.9):

$$\frac{\partial \mathbf{Q}}{\partial \mathbf{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -u & \frac{1}{\rho} & 0 & 0 & 0 & 0 \\ \frac{\rho}{-v} & 0 & \frac{1}{\rho} & 0 & 0 & 0 \\ \frac{\rho}{-w} & 0 & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{\rho}{(\gamma - 1) \frac{\|\mathbf{u}\|_2^2}{2} - Y_k} & (1 - \gamma) u & (1 - \gamma) v & (1 - \gamma) w & \gamma - 1 & (1 - \gamma) \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ -\frac{Y_k}{\rho} & 0 & 0 & 0 & 0 & \frac{1}{\rho} \end{bmatrix} \quad (\text{F.10})$$

F.3.3. Flux Jacobian matrices with respect to \mathbf{Q}

Differentiating \mathbf{E}_c , \mathbf{F}_c and \mathbf{G}_c , given in Eq. (4.51), with respect to \mathbf{Q} variables gives:

$$\frac{\partial \mathbf{E}_c}{\partial \mathbf{Q}} = \begin{bmatrix} u & \rho & 0 & 0 & 0 & 0 \\ u^2 & 2\rho u & 0 & 0 & 1 & 0 \\ uv & \rho v & \rho u & 0 & 0 & 0 \\ uw & \rho w & 0 & \rho u & 0 & 0 \\ u \left(h_s - TC_p + \frac{\|\mathbf{u}\|_2^2}{2} \right) & \rho \left(h_s + \frac{\|\mathbf{u}\|_2^2}{2} + u^2 \right) & \rho uv & \rho uw & \frac{\gamma u}{\gamma - 1} & \rho u \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ uY_k & \rho Y_k & 0 & 0 & 0 & \rho u \end{bmatrix} \quad (\text{F.11})$$

$$\frac{\partial \mathbf{F}_c}{\partial \mathbf{Q}} = \begin{bmatrix} v & 0 & \rho & 0 & 0 & 0 \\ uv & \rho v & \rho u & 0 & 0 & 0 \\ v^2 & 0 & 2\rho v & 0 & 1 & 0 \\ vw & 0 & \rho w & \rho v & 0 & 0 \\ v \left(h_s - TC_p + \frac{\|\mathbf{u}\|_2^2}{2} \right) & \rho uv & \rho \left(h_s + \frac{\|\mathbf{u}\|_2^2}{2} + v^2 \right) & \rho vw & \frac{\gamma v}{\gamma - 1} & \rho v \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ vY_k & 0 & \rho Y_k & 0 & 0 & \rho v \end{bmatrix} \quad (\text{F.12})$$

$$\frac{\partial \mathbf{G}_c}{\partial \mathbf{Q}} = \begin{bmatrix} w & 0 & 0 & \rho & 0 & 0 \\ uw & \rho w & 0 & \rho u & 0 & 0 \\ vw & 0 & \rho w & \rho v & 0 & 0 \\ w^2 & 0 & 0 & 2\rho w & 1 & 0 \\ w \left(h_s - TC_p + \frac{\|\mathbf{u}\|_2^2}{2} \right) & \rho uw & \rho vw & \rho \left(h_s + \frac{\|\mathbf{u}\|_2^2}{2} + w^2 \right) & \frac{\gamma w}{\gamma - 1} & \rho w \left(h_{sk} - TC_p \frac{W}{W_k} \right) \\ wY_k & 0 & 0 & \rho Y_k & 0 & \rho w \end{bmatrix} \quad (\text{F.13})$$

F.3.4. Final expressions of matrices A , B and C

Since passage matrices from conservative to or from primitive variables and flux Jacobian matrices with respect to \mathbf{Q} have been computed, the Jacobian matrices A , B and C can be obtained:

$$A \equiv \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{E}_c}{\partial \mathbf{Q}} = \begin{bmatrix} u & \rho & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 1/\rho & 0 \\ 0 & 0 & u & 0 & 0 & 0 \\ 0 & 0 & 0 & u & 0 & 0 \\ 0 & \gamma P & 0 & 0 & u & 0 \\ 0 & 0 & 0 & 0 & 0 & u \end{bmatrix} \quad (\text{F.14})$$

$$B \equiv \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{F}_c}{\partial \mathbf{Q}} = \begin{bmatrix} v & 0 & \rho & 0 & 0 & 0 \\ 0 & v & 0 & 0 & 0 & 0 \\ 0 & 0 & v & 0 & 1/\rho & 0 \\ 0 & 0 & 0 & v & 0 & 0 \\ 0 & 0 & \gamma P & 0 & v & 0 \\ 0 & 0 & 0 & 0 & 0 & v \end{bmatrix} \quad (\text{F.15})$$

$$C \equiv \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} \cdot \frac{\partial \mathbf{G}_c}{\partial \mathbf{Q}} = \begin{bmatrix} w & 0 & 0 & \rho & 0 & 0 \\ 0 & w & 0 & 0 & 0 & 0 \\ 0 & 0 & w & 0 & 0 & 0 \\ 0 & 0 & 0 & w & 1/\rho & 0 \\ 0 & 0 & 0 & \gamma P & w & 0 \\ 0 & 0 & 0 & 0 & 0 & w \end{bmatrix} \quad (\text{F.16})$$

where $\gamma P = \rho(\gamma - 1)TC_p$ even for a thermally multi-species perfect gas. These flux Jacobians are almost the same than the ones obtain for a calorically perfect gas.

F.4. Diagonalization and wave equation in isoparametric space

Now that 3D Euler equations are completely expressed with primitive variables in the physical space, the transformation to the isoparametric space can be done. Once the equations will be written in the isoparametric domain, the diagonalization will be considered.

F.4.1. Governing equations for primitive variables in isoparametric space

Starting from Eq. (F.4) and using the chain rule with respect to the isoparametric coordinates:

$$\begin{aligned} \frac{\partial \mathbf{Q}}{\partial t} + A \left(\xi_x \frac{\partial \mathbf{Q}}{\partial \xi} + \eta_x \frac{\partial \mathbf{Q}}{\partial \eta} + \zeta_x \frac{\partial \mathbf{Q}}{\partial \zeta} \right) + B \left(\xi_y \frac{\partial \mathbf{Q}}{\partial \xi} + \eta_y \frac{\partial \mathbf{Q}}{\partial \eta} + \zeta_y \frac{\partial \mathbf{Q}}{\partial \zeta} \right) + C \left(\xi_z \frac{\partial \mathbf{Q}}{\partial \xi} + \eta_z \frac{\partial \mathbf{Q}}{\partial \eta} + \zeta_z \frac{\partial \mathbf{Q}}{\partial \zeta} \right) &= \mathbf{0} \\ \Rightarrow \frac{\partial \mathbf{Q}}{\partial t} + A_\xi \frac{\partial \mathbf{Q}}{\partial \xi} + A_\eta \frac{\partial \mathbf{Q}}{\partial \eta} + A_\zeta \frac{\partial \mathbf{Q}}{\partial \zeta} &= \mathbf{0} \end{aligned} \quad (\text{F.17})$$

where:

$$A_\xi = \xi_x A + \xi_y B + \xi_z C, \quad A_\eta = \eta_x A + \eta_y B + \eta_z C, \quad A_\zeta = \zeta_x A + \zeta_y B + \zeta_z C \quad (\text{F.18})$$

To determine the waves traveling in ξ , η and ζ directions, the matrices A_ξ , A_η and A_ζ have to be diagonalized. Since this diagonalization is almost identical for these three directions, it will be explained here for A_ξ only but the reasoning is exactly equivalent for A_η and A_ζ . Thus, the diagonalization of A_ξ will give the expressions of the waves crossing a boundary with a ξ -normal direction.

F.4.2. Diagonalization of A_ξ

F.4.2.1. Expression of A_ξ

The matrix A_ξ can be explicitly expressed thanks to Eq. (F.18) and Eq. (F.14)-Eq. (F.16):

$$A_\xi = \begin{bmatrix} \hat{u} & \xi_x \rho & \xi_y \rho & \xi_z \rho & 0 & 0 \\ 0 & \hat{u} & 0 & 0 & \xi_x / \rho & 0 \\ 0 & 0 & \hat{u} & 0 & \xi_y / \rho & 0 \\ 0 & 0 & 0 & \hat{u} & \xi_z / \rho & 0 \\ 0 & \xi_x \gamma P & \xi_y \gamma P & \xi_z \gamma P & \hat{u} & 0 \\ 0 & 0 & 0 & 0 & 0 & \hat{u} \end{bmatrix} \quad (\text{F.19})$$

with $\hat{u} = \xi_x u + \xi_y v + \xi_z w$.

F.4.2.2. Compute eigenvalues of A_ξ

The computation of the eigenvalues of A_ξ , noted λ here, gives the following results:

$$\det(A_\xi - \lambda I) = (\hat{u} - \lambda)^4 \left[(\hat{u} - \lambda)^2 - c^2 (\xi_x^2 + \xi_y^2 + \xi_z^2) \right] = (\hat{u} - \lambda)^4 \left[(\hat{u} - \lambda)^2 - \hat{c}^2 \right] \quad (\text{F.20})$$

where $\hat{c} = c \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}$ and \det stands for the determinant application. Thus, the $5 + N_s$ eigenvalues are:

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_{5+k} = \hat{u} \text{ with } k \in \llbracket 1, N_s \rrbracket, \lambda_4 = \hat{u} + \hat{c} \text{ and } \lambda_5 = \hat{u} - \hat{c} \quad (\text{F.21})$$

Finally, the eigenvalues expressed in Eq. (F.21) are usually divided by $\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}$ in order to introduce the unit normal vector defined in Eq. (A.13). Then, Eq. (F.21) becomes:

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_{5+k} = u_n \text{ with } k \in \llbracket 1, N_s \rrbracket, \lambda_4 = u_n + c \text{ and } \lambda_5 = u_n - c \quad (\text{F.22})$$

That is why, the matrix A_ξ is usually written in terms of u_n by dividing it by $\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}$ and from now, A_ξ will be referred to as $A_\xi / \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}$.

F.4.2.3. Compute right eigenvectors of A_ξ for eigenvalue u_n

The eigenvalue u_n is of multiplicity $3 + N_s$ which means that $3 + N_s$ right eigenvectors associated to this eigenvalue have to be found. Starting from

$$A_\xi - u_n \mathbf{I} = \begin{bmatrix} 0 & n_x^u \rho & n_y^u \rho & n_z^u \rho & 0 & 0 \\ 0 & 0 & 0 & 0 & n_x^u / \rho & 0 \\ 0 & 0 & 0 & 0 & n_y^u / \rho & 0 \\ 0 & 0 & 0 & 0 & n_z^u / \rho & 0 \\ 0 & n_x^u \rho c^2 & n_y^u \rho c^2 & n_z^u \rho c^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{F.23})$$

where γP was changed into ρc^2 . The columns numbered $5 + 1$ to $5 + N_s$ in Eq. (F.23) are null which indicates that N_s right eigenvectors can be taken following:

$$\mathbf{R}^{(5+k)} = (0, \dots, 0, 1, 0, \dots, 0)^T \text{ for } k \in \llbracket 1, N_s \rrbracket \quad (\text{F.24})$$

where the 1 is located on the k -th component of $\mathbf{R}^{(5+k)}$. Consequently, it remains to find 3 right eigenvectors of $A_\xi - u_n \mathbf{I}$. Under the general form $\mathbf{R}^{(i)} = (r_1^{(i)}, r_2^{(i)}, r_3^{(i)}, r_4^{(i)}, r_5^{(i)}, r_{5+k}^{(i)})^T$ with $k \in \llbracket 1, N_s \rrbracket$

and $i \in \llbracket 1, 3 \rrbracket$. According to Eq. (F.23), these vectors satisfy (page 175 of Hirsch's book [318]):

$$r_1^{(i)} \text{ and } r_{5+k}^{(i)} \text{ are arbitrary} \quad (\text{F.25})$$

$$r_5^{(i)} = 0 \quad (\text{F.26})$$

$$r_2^{(i)} n_x^u + r_3^{(i)} n_y^u + r_4^{(i)} n_z^u = 0 \quad (\text{F.27})$$

The common choice is to consider [318]:

$$\mathbf{R}^{(1)} = \left(n_x^u, 0, (c/\rho) n_z^u, -(c/\rho) n_y^u, 0, 0, \dots, 0 \right)^T \quad (\text{F.28})$$

$$\mathbf{R}^{(2)} = \left(n_y^u, -(c/\rho) n_z^u, 0, (c/\rho) n_x^u, 0, 0, \dots, 0 \right)^T \quad (\text{F.29})$$

$$\mathbf{R}^{(3)} = \left(n_z^u, (c/\rho) n_y^u, -(c/\rho) n_x^u, 0, 0, 0, \dots, 0 \right)^T \quad (\text{F.30})$$

where the constant c/ρ was chosen for units reason as it is explained in paragraph F.4.3.

F.4.2.4. Compute right eigenvector of A_ξ for eigenvalue $u_n + c$

The eigenvalue $u_n + c$ is of multiplicity 1 which means that 1 right eigenvector associated to this eigenvalue has to be found. It corresponds to $\mathbf{R}^{(4)}$ and satisfies:

$$[A_\xi - (u_n + c) \mathbf{I}] \cdot \mathbf{R}^{(4)} = 0 \iff \begin{bmatrix} -c & n_x^u \rho & n_y^u \rho & n_z^u \rho & 0 & 0 \\ 0 & -c & 0 & 0 & n_x^u / \rho & 0 \\ 0 & 0 & -c & 0 & n_y^u / \rho & 0 \\ 0 & 0 & 0 & -c & n_z^u / \rho & 0 \\ 0 & n_x^u \rho c^2 & n_y^u \rho c^2 & n_z^u \rho c^2 & -c & 0 \\ 0 & 0 & 0 & 0 & 0 & -c \end{bmatrix} \cdot \begin{bmatrix} r_1^{(4)} \\ r_2^{(4)} \\ r_3^{(4)} \\ r_4^{(4)} \\ r_5^{(4)} \\ r_{5+k}^{(4)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{F.31})$$

From Eq. (F.31), it can be written that:

$$r_1^{(4)} = \frac{\rho \left(n_x^u r_2^{(4)} + n_y^u r_3^{(4)} + n_z^u r_4^{(4)} \right)}{c} \quad (\text{F.32})$$

$$r_2^{(4)} = \frac{n_x^u}{\rho c} r_5^{(4)} \quad (\text{F.33})$$

$$r_3^{(4)} = \frac{n_y^u}{\rho c} r_5^{(4)} \quad (\text{F.34})$$

$$r_4^{(4)} = \frac{n_z^u}{\rho c} r_5^{(4)} \quad (\text{F.35})$$

$$r_5^{(4)} = \rho c \left(n_x^u r_2^{(4)} + n_y^u r_3^{(4)} + n_z^u r_4^{(4)} \right) \quad (\text{F.36})$$

$$r_{5+k}^{(4)} = 0 \text{ for } k \in \llbracket 1, N_s \rrbracket \quad (\text{F.37})$$

Combining Eq. (F.32) and Eq. (F.36) gives:

$$r_5^{(4)} = c^2 r_1^{(4)} \quad (\text{F.38})$$

so that $r_2^{(4)}$, $r_3^{(4)}$ and $r_4^{(4)}$ can be expressed as function of $r_1^{(4)}$ only:

$$r_2^{(4)} = \frac{n_x^u c}{\rho} r_1^{(4)}, \quad r_3^{(4)} = \frac{n_y^u c}{\rho} r_1^{(4)}, \quad r_4^{(4)} = \frac{n_z^u c}{\rho} r_1^{(4)} \quad (\text{F.39})$$

The common choice for $r_1^{(4)}$ is $\rho/(\sqrt{2}c)$ [318]. Therefore, $\mathbf{R}^{(4)}$ has the following expression:

$$\mathbf{R}^{(4)} = \left(\rho/(\sqrt{2}c), n_x^u/\sqrt{2}, n_y^u/\sqrt{2}, n_z^u/\sqrt{2}, \rho c/\sqrt{2}, 0, \dots, 0 \right)^T \quad (\text{F.40})$$

F.4.2.5. Compute right eigenvector of A_ξ for eigenvalue $u_n - c$

As for eigenvalue $u_n + c$, the eigenvalue $u_n - c$ is of multiplicity 1 so that 1 right eigenvector associated to this eigenvalue has also to be found corresponding to $\mathbf{R}^{(5)}$ here. Following the same methodology detailed in F.4.2.4, $\mathbf{R}^{(5)}$ is taken as:

$$\mathbf{R}^{(5)} = \left(\rho/(\sqrt{2}c), -n_x^u/\sqrt{2}, -n_y^u/\sqrt{2}, -n_z^u/\sqrt{2}, \rho c/\sqrt{2}, 0, \dots, 0 \right)^T \quad (\text{F.41})$$

F.4.2.6. Synthesis

The right (its columns are right eigenvectors of A_ξ) and left (its rows are left eigenvectors of A_ξ) eigenvectors matrices are given by (extension of the work of Warming *et al.* [319] to a multi-species gas):

$$R_{A_\xi} = \left(\mathbf{R}^{(1)}, \mathbf{R}^{(2)}, \mathbf{R}^{(3)}, \mathbf{R}^{(4)}, \mathbf{R}^{(5)}, \mathbf{R}^{(5+k)} \right)^T = \begin{bmatrix} n_x^u & n_y^u & n_z^u & \rho/(\sqrt{2}c) & \rho/(\sqrt{2}c) & 0 \\ 0 & -n_z^u c/\rho & n_y^u c/\rho & n_x^u/\sqrt{2} & -n_x^u/\sqrt{2} & 0 \\ n_z^u c/\rho & 0 & -n_x^u c/\rho & n_y^u/\sqrt{2} & -n_y^u/\sqrt{2} & 0 \\ -n_y^u c/\rho & n_x^u c/\rho & 0 & n_z^u/\sqrt{2} & -n_z^u/\sqrt{2} & 0 \\ 0 & 0 & 0 & \rho c/\sqrt{2} & \rho c/\sqrt{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{F.42})$$

$$L_{A_\xi} = \left(\mathbf{L}^{(1)}, \mathbf{L}^{(2)}, \mathbf{L}^{(3)}, \mathbf{L}^{(4)}, \mathbf{L}^{(5)}, \mathbf{L}^{(5+k)} \right)^T = \begin{bmatrix} n_x^u & 0 & n_z^u \rho/c & -n_y^u \rho/c & -n_x^u/c^2 & 0 \\ n_y^u & -n_z^u \rho/c & 0 & n_x^u \rho/c & -n_y^u/c^2 & 0 \\ n_z^u & n_y^u \rho/c & -n_x^u \rho/c & 0 & -n_z^u/c^2 & 0 \\ 0 & n_x^u/\sqrt{2} & n_y^u/\sqrt{2} & n_z^u/\sqrt{2} & 1/(\sqrt{2}\rho c) & 0 \\ 0 & -n_x^u/\sqrt{2} & -n_y^u/\sqrt{2} & -n_z^u/\sqrt{2} & 1/(\sqrt{2}\rho c) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{F.43})$$

where $L_{A_\xi} = R_{A_\xi}^{-1}$. Therefore, A_ξ is expressed using L_{A_ξ} , R_{A_ξ} and $\Lambda_{A_\xi} = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_{5+k})$:

$$A_\xi = R_{A_\xi} \Lambda_{A_\xi} L_{A_\xi} \quad (\text{F.44})$$

F.4.3. Wave equation along ξ -direction for 3D Euler equations in isoparametric coordinates

Using the decomposition of A_ξ in Eq. (F.44), Eq. (F.17) is given by:

$$\frac{\partial \mathbf{Q}}{\partial t} + R_{A_\xi} \Lambda_{A_\xi} L_{A_\xi} \cdot \frac{\partial \mathbf{Q}}{\partial \xi} + A_\eta \frac{\partial \mathbf{Q}}{\partial \eta} + A_\zeta \frac{\partial \mathbf{Q}}{\partial \zeta} = \mathbf{0} \quad (\text{F.45})$$

Multiplying Eq. (F.45) on the left by L_{A_ξ} , it ends up with:

$$L_{A_\xi} \frac{\partial \mathbf{Q}}{\partial t} + \Lambda_{A_\xi} L_{A_\xi} \cdot \frac{\partial \mathbf{Q}}{\partial \xi} = -L_{A_\xi} \left(A_\eta \frac{\partial \mathbf{Q}}{\partial \eta} + A_\zeta \frac{\partial \mathbf{Q}}{\partial \zeta} \right) \quad (\text{F.46})$$

Eq. (F.46) is a wave equation of the form:

$$\frac{\partial \mathbf{W}}{\partial t} + \mathcal{L} = \mathcal{T} \quad (\text{F.47})$$

with $\partial \mathbf{W}$ the vector of characteristic variables, \mathcal{L} the wave strengths and \mathcal{T} representing transverse terms all defined as:

$$\partial \mathbf{W} = L_{A_\xi} \partial \mathbf{Q} = \begin{pmatrix} n_x^u \partial \rho + (n_z^u \rho / c) \partial v - (n_y^u \rho / c) \partial w - (n_x^u / c^2) \partial P \\ n_y^u \partial \rho - (n_z^u \rho / c) \partial u + (n_x^u \rho / c) \partial w - (n_y^u / c^2) \partial P \\ n_z^u \partial \rho + (n_y^u \rho / c) \partial u - (n_x^u \rho / c) \partial v - (n_z^u / c^2) \partial P \\ (n_x^u / \sqrt{2}) \partial u + (n_y^u / \sqrt{2}) \partial v + (n_z^u / \sqrt{2}) \partial w + (1 / \sqrt{2} \rho c) \partial P \\ - (n_x^u / \sqrt{2}) \partial u - (n_y^u / \sqrt{2}) \partial v - (n_z^u / \sqrt{2}) \partial w + (1 / \sqrt{2} \rho c) \partial P \\ \partial Y_k \end{pmatrix} = R_{A_\xi}^{-1} \partial \mathbf{Q} \quad (\text{F.48})$$

$$\mathcal{L} = \Lambda_{A_\xi} \frac{\partial \mathbf{W}}{\partial \xi} \quad (\text{F.49})$$

$$\mathcal{T} = -L_{A_\xi} \left(A_\eta \frac{\partial \mathbf{Q}}{\partial \eta} + A_\zeta \frac{\partial \mathbf{Q}}{\partial \zeta} \right) \quad (\text{F.50})$$

Vector $\partial \mathbf{W}$ is composed of $3 + N_s$ entropy waves namely $\partial W_1, \partial W_2, \partial W_3, \partial W_{5+1}, \dots, \partial W_{5+N_s}$, propagating at u_n and of two acoustic waves ∂W_+ and ∂W_- propagating respectively at $u_n + c$ and $u_n - c$. Note the presence of the ratio ρ/c in the 3 first lines of Eq. (F.48) coming from the choice of the constant for vectors $\mathbf{R}^{(1)}, \mathbf{R}^{(2)}$ and $\mathbf{R}^{(3)}$ done in paragraph F.4.2.3. Thanks to this choice, the units in the 3 first lines of Eq. (F.48) are correct. Finally, Eq. (F.48) allows to introduce the passage matrix from primitive to characteristic variables noted $P_{\mathbf{Q}}$ in the manuscript:

$$P_{\mathbf{Q}} \equiv \frac{\partial \mathbf{W}}{\partial \mathbf{Q}} = L_{A_\xi} \quad (\text{F.51})$$

Compute constants in VKP spectrum

The objective is to obtain analytical expressions for the two constants used in Eq. (6.62) namely α and $\kappa_{e,\text{VKP}}$ and also for ϵ_{VKP} .

G.1. Compute α

This constant will be obtained to satisfy:

$$\begin{aligned} & \int_0^\infty \left(\alpha \frac{u_{rms}^2}{\kappa_{e,\text{VKP}}} \frac{(\kappa/\kappa_{e,\text{VKP}})^4}{(1 + (\kappa/\kappa_{e,\text{VKP}})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] d\kappa \right) = \frac{3}{2} u_{rms}^2 \\ \Rightarrow & \alpha \int_0^\infty \left(\frac{1}{\kappa_{e,\text{VKP}}} \frac{(\kappa/\kappa_{e,\text{VKP}})^4}{(1 + (\kappa/\kappa_{e,\text{VKP}})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] d\kappa \right) = \frac{3}{2} \end{aligned} \quad (\text{G.1})$$

Now, let's use the following change of variable in the integral: $x = \kappa/\kappa_{e,\text{VKP}} \iff d\kappa = \kappa_{e,\text{VKP}} dx$. By introducing $z = 2(\kappa_{e,\text{VKP}}/\kappa_{Kol})^2$, Eq. (G.1) becomes:

$$\alpha \int_0^\infty \left(\frac{x^4}{(1 + x^2)^{17/6}} \exp [-zx^2] dx \right) = \frac{3}{2} \quad (\text{G.2})$$

Then, another change of variable is used by setting: $y = x^2 \iff dx = dy/(2\sqrt{y})$ and the new integral expressed for variable y is:

$$\begin{aligned} & \alpha \int_0^\infty \left(\frac{y^2}{2\sqrt{y} (1 + y)^{17/6}} \exp [-zy] dy \right) = \frac{3}{2} \\ \Rightarrow & \alpha \int_0^\infty \left(y^{3/2} (1 + y)^{-17/6} \exp [-zy] dy \right) = 3 \end{aligned} \quad (\text{G.3})$$

The integral defined in Eq. (G.3) can be calculated under certain assumptions using a mathematical result concerning the gamma function [259]:

$$\int_0^\infty y^{a-1} (1 + y)^{b-a-1} \exp [-zy] dy = \begin{cases} \frac{\Gamma(a) \Gamma(1-b)}{\Gamma(1+a-b)} & \text{if } a > 0, b < 1 \text{ and } z \rightarrow 0 \\ \Gamma(b-1) z^{1-b} & \text{if } a > 0, b > 1 \text{ and } z \rightarrow 0 \end{cases} \quad (\text{G.4})$$

In the case of Eq. (G.3): $a = 5/2$ and $b - a - 1 = -17/6 \iff b = 2/3$. Consequently, if it is assumed that $(k_{e,\text{VKP}} k_{Kol})^2 \rightarrow 0 \iff z \rightarrow 0$, Eq. (G.3) can be approximated by:

$$\begin{aligned}
 & \alpha \frac{\Gamma(5/2) \Gamma(1/3)}{\Gamma(17/6)} = 3 \\
 \Rightarrow & \alpha \frac{\frac{3}{2} \frac{1}{2} \Gamma(1/2) \Gamma(1/3)}{\frac{11}{6} \frac{5}{6} \Gamma(5/6)} = 3 \\
 \Rightarrow & \alpha \frac{\frac{3}{2} \frac{1}{2} \Gamma(1/2) \Gamma(1/3)}{\frac{11}{6} \frac{5}{6} \Gamma(5/6)} = 3 \\
 \Rightarrow & \alpha \frac{27 \sqrt{\pi} \Gamma(1/3)}{55 \Gamma(5/6)} = 3 \\
 \Rightarrow & \alpha = \frac{55 \Gamma(5/6)}{9 \sqrt{\pi} \Gamma(1/3)}
 \end{aligned} \tag{G.5}$$

where the following properties of the gamma function were employed: $\Gamma(1+x) = x\Gamma(x) \ \forall x > 0$ and $\Gamma(1/2) = \sqrt{\pi}$.

G.2. Compute $\kappa_{e,\text{VKP}}$

This constant will be obtained to satisfy:

$$\frac{\pi}{2u_{rms}^2} \int_0^\infty \left(\alpha \frac{u_{rms}^2}{\kappa_{e,\text{VKP}}} \frac{(\kappa/\kappa_{e,\text{VKP}})^4}{\kappa (1 + (\kappa/\kappa_{e,\text{VKP}})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] d\kappa \right) = L_e \tag{G.6}$$

$$\Rightarrow \frac{\pi\alpha}{2\kappa_{e,\text{VKP}}} \int_0^\infty \left(\frac{(\kappa/\kappa_{e,\text{VKP}})^4}{\kappa (1 + (\kappa/\kappa_{e,\text{VKP}})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] d\kappa \right) = L_e \tag{G.7}$$

$$\Rightarrow \frac{\pi\alpha}{2\kappa_{e,\text{VKP}}} \int_0^\infty \left(\frac{x^3}{(1+x^2)^{17/6}} \exp[-zx^2] dx \right) = L_e \tag{G.8}$$

still by setting $x = \kappa/\kappa_{e,\text{VKP}}$ and $z = 2(\kappa_{e,\text{VKP}}/\kappa_{Kol})^2$. Then, the same change of variable as in Section G.1 $y = x^2$ is used to have an integral that can be computed with gamma functions:

$$\begin{aligned}
 & \frac{\pi\alpha}{2\kappa_{e,\text{VKP}}} \int_0^\infty \left(\frac{y^{3/2}}{2\sqrt{y}(1+y)^{17/6}} \exp[-zy] dy \right) = L_e \\
 \Rightarrow & \frac{\pi\alpha}{4\kappa_{e,\text{VKP}}} \int_0^\infty \left(y(1+y)^{-17/6} \exp[-zy] dy \right) = L_e
 \end{aligned} \tag{G.9}$$

Consequently, applying Eq. (G.4) this time with $a = 2$ and $b = 1/6$ and under the assumption that $z \rightarrow 0$, Eq. (G.9) becomes:

$$\begin{aligned}
 & \frac{\pi\alpha}{4\kappa_{e,\text{VKP}}} \frac{\Gamma(2) \Gamma(5/6)}{\Gamma(17/6)} = L_e \\
 \Rightarrow & \frac{\pi\alpha}{4\kappa_{e,\text{VKP}}} \frac{1 \times \Gamma(5/6)}{\frac{55}{36} \Gamma(5/6)} = L_e \\
 \Rightarrow & \kappa_{e,\text{VKP}} = \frac{9\pi\alpha}{55L_e}
 \end{aligned} \tag{G.10}$$

G.3. Compute ϵ_{VKP}

Starting from the definition of the turbulent dissipation rate for HIT:

$$\begin{aligned}
\epsilon_{\text{VKP}} &= 2\nu \int_0^\infty \left(\kappa^2 \alpha \frac{u_{rms}^2}{\kappa_{e,\text{VKP}}} \frac{(\kappa/\kappa_{e,\text{VKP}})^4}{(1 + (\kappa/\kappa_{e,\text{VKP}})^2)^{17/6}} \exp \left[-2 \left(\frac{\kappa}{\kappa_{Kol}} \right)^2 \right] d\kappa \right) \\
\Rightarrow \epsilon_{\text{VKP}} &= 2\nu \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^2 \int_0^\infty \left(\frac{x^6}{(1 + x^2)^{17/6}} \exp [-zx^2] dx \right) \\
\Rightarrow \epsilon_{\text{VKP}} &= 2\nu \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^2 \int_0^\infty \left(\frac{y^3}{2\sqrt{y}(1 + y)^{17/6}} \exp [-zy] dy \right) \\
\Rightarrow \epsilon_{\text{VKP}} &= \nu \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^2 \int_0^\infty \left(y^{5/2} (1 + y)^{-17/6} \exp [-zy] dy \right)
\end{aligned} \tag{G.11}$$

Thus in that case Eq. (G.4) is used with $a = 7/2$ and $b = 5/3 > 1$ and still assuming $z \rightarrow 0$:

$$\begin{aligned}
\epsilon_{\text{VKP}} &= \nu \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^2 \Gamma(2/3) \left[2 \left(\frac{\kappa_{e,\text{VKP}}}{\kappa_{Kol}} \right)^2 \right]^{-2/3} \\
\Rightarrow \epsilon_{\text{VKP}} &= \nu \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^{2/3} \Gamma(2/3) 2^{-2/3} \kappa_{Kol}^{4/3}
\end{aligned} \tag{G.12}$$

Finally, the link between ϵ_{VKP} and κ_{Kol} is employed to isolate κ_{Kol} :

$$\kappa_{Kol} = \frac{\epsilon_{\text{VKP}}^{1/4}}{\nu^{3/4}} \Rightarrow \kappa_{Kol}^{4/3} = \frac{\epsilon_{\text{VKP}}^{1/3}}{\nu} \tag{G.13}$$

Therefore, Eq. (G.12) becomes:

$$\begin{aligned}
\epsilon_{\text{VKP}} &= \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^{2/3} \Gamma(2/3) 2^{-2/3} \epsilon_{\text{VKP}}^{1/3} \\
\Rightarrow \epsilon_{\text{VKP}}^{2/3} &= \alpha u_{rms}^2 \kappa_{e,\text{VKP}}^{2/3} \Gamma(2/3) 2^{-2/3} \\
\Rightarrow \epsilon_{\text{VKP}} &= [\alpha \Gamma(2/3)]^{3/2} \frac{\kappa_{e,\text{VKP}}^3 u_{rms}^3}{2} \sim \frac{u_{rms}^3}{L_e}
\end{aligned} \tag{G.14}$$

List of Figures

| | |
|---|-----|
| 2.1. Evolution of PEA functions f_1 and f_2 with equivalence ratio for the CH ₄ /Air-2S-BFER. | 25 |
| 3.1. Sketch of the energy cascade showing mass kinetic energy density E [$\text{m}^3.\text{s}^{-2}$] versus wavenumber κ [m^{-1}], in log-log scale, for homogeneous isotropic turbulence. | 37 |
| 3.2. Results of flame/turbulence interactions with non-thickened and thickened flames simulated in [163, 177]. Left: non-thickened flame. Right: thickened flame. | 45 |
| 4.1. Linear isoparametric and physical domains in 1D. | 53 |
| 4.2. Quadratic isoparametric and physical domains in 1D. | 53 |
| 4.3. Locations of SP (▲) and FP (■) for $p = 2$ in isoparametric domain \mathcal{L} | 55 |
| 4.4. Locations of SP (▲) and FP (■) for $p = 3$ in isoparametric domain \mathcal{L} | 55 |
| 4.5. 1 st and 2 nd steps of the SD process in 1D for a hyperbolic equation. | 57 |
| 4.6. 3 rd and 4 th steps of the SD process in 1D for a hyperbolic equation. | 57 |
| 4.7. 5 th and 6 th steps of the SD process in 1D for a hyperbolic equation. | 57 |
| 4.8. Defining common values for an element Ω_e | 58 |
| 4.9. Correction functions used in this work for the SDLIFT approach shown at two different polynomial degrees. | 62 |
| 4.10. Illustration of the stencil of the BR1 diffusion scheme. | 63 |
| 4.11. Illustration of the stencil of the BR2 diffusion scheme. | 64 |
| 4.12. Isoparametric transformation between a linear standard hexahedron and a linear hexahedron. | 65 |
| 4.13. Locations of SP (▲), ξ -FP (■), η -FP (■) and ζ -FP (■) for $p = 1$ in isoparametric domain \mathcal{H} | 69 |
| 4.14. Illustration of the Riemann problem in a 1D general case. | 72 |
| 4.15. Situation of the Riemann problem solved with the HLLC solver when $\mathbf{n}^u = [1, 0, 0]^T$. The star region consists of two constant states separated by a contact wave of speed S_* . | 73 |
| 4.16. Illustration of the mortar element method at an interface between a left element with $p_L = 1$ and a right element with $p_R = 2$. Red triangles represent SP and blue squares represent FP in the ξ direction. | 80 |
| 5.1. Illustration of the computation of a function f at a boundary FP to impose $(\partial f / \partial \xi)$ at this boundary FP. The polynomial degree is set to $p = 3$ with five FP along direction ξ . | 89 |
| 5.2. Waves crossing NSCBC inlet and outlet. | 94 |
| 6.1. Geometry of a channel flow. | 103 |
| 6.2. Geometry of a pipe flow. | 107 |
| 6.3. Non-dimensionalized velocity profiles \bar{w}_{in}/U_d for different values of n . The case $n = 1/2$ corresponds to the laminar Poiseuille's flow shown here for comparison. | 111 |
| 6.4. Geometry of the n -th Fourier mode. | 117 |
| 6.5. Artificial tripping using a source term added to momentum equations. | 120 |
| 6.6. Computational domain and boundary conditions for the injection of turbulence inside a 3D periodic channel. | 121 |
| 6.7. Passot-Pouquet turbulent spectrum injected at $z = 0$ | 123 |

| | |
|---|-----|
| 6.8. Contours of Q-criterion colored by vorticity magnitude. The flow is going from the left to the right. | 124 |
| 6.9. Evolution of the TKE and of the RMS velocities along channel axis. | 124 |
| 6.10. Comparison of injected PP and VKP spectrums at $z = 0$ | 125 |
| 6.11. VKP and PP spectrums represented for different ranges of κ . Blue curves, green curves and orange curves correspond respectively to κ such that $E(\kappa) > 10^{-12} \text{ m}^3 \cdot \text{s}^{-2}$, $\kappa \in [\kappa_{min}, \kappa_{Kol}]$ and $\kappa \in [\kappa_{min}, \kappa_{max}]$ | 125 |
| 6.12. Comparison of TKE decays between PP and VKP obtained with JAGUAR and RMS velocities decay in the channel for the VKP spectrum. | 126 |
| 6.13. Evolution of statistics using Passot-Pouquet energy spectrum for turbulence injection. | 127 |
| 6.14. Evolution of statistics using von Karman-Pao energy spectrum for turbulence injection. | 128 |
| 6.15. Computational domain and boundary conditions for the turbulent wall-bounded channel flow. | 129 |
| 6.16. Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at different axial positions in the channel when only HIT is injected at channel inlet. DNS results of Del Alamo <i>et al.</i> [241] are also represented for comparison. | 131 |
| 6.17. Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at different axial positions in the channel. DNS results of Del Alamo <i>et al.</i> [241] are also represented for comparison. | 132 |
| 6.18. Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at $x = 7 \times 2h$ for three values of C_D . DNS results of Del Alamo <i>et al.</i> [241] and results with only HIT injected at the inlet are also represented for comparison. | 133 |
| 6.19. Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at $x = 7 \times 2h$ for three values of C_D . DNS results of Del Alamo <i>et al.</i> [241] and results with only HIT injected at the inlet are also represented for comparison. | 134 |
| 6.20. Evolution of $\bar{U}^+ = \bar{u}/u_\tau$ as a function of y^+ at $x = 7 \times 2h$ for three values of L_e when injecting non-homogenous and anisotropic turbulence. DNS results of Del Alamo <i>et al.</i> [241] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison. | 135 |
| 6.21. Evolution of RMS velocities non-dimensionalized by u_τ as a function of y^+ at $x = 7 \times 2h$ for three values of L_e when injecting non-homogenous and anisotropic turbulence. DNS results of Del Alamo <i>et al.</i> [241] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison. | 136 |
| 6.22. Evolution of mean pressure and RMS of pressure at channel center ($y = 0$) along channel axis for HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous anisotropic turbulence with $L_e = 0.85 \text{ mm}$ cases. | 136 |
| 6.23. Computational domain, boundary conditions and cross-sectional view of a quarter-section of the mesh elements used for the turbulent wall-bounded pipe flow. | 137 |
| 6.24. Evolution of $\bar{U}^+ = \bar{w}/u_\tau$ as a function of r^+ at different axial positions in the pipe. DNS results of El Khoury <i>et al.</i> [247] are also represented for comparison. | 139 |
| 6.25. Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at different axial positions in the pipe when only HIT is injected at pipe inlet. DNS results of El Khoury <i>et al.</i> [247] are also represented for comparison. | 140 |
| 6.26. Evolution of $\bar{U}^+ = \bar{w}/u_\tau$ as a function of r^+ at $z = 7D$ for three values of C_D . DNS results of El Khoury <i>et al.</i> [247] and results with only HIT injected at the inlet are also represented for comparison. | 141 |
| 6.27. Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at $z = 7D$ for three values of C_D . DNS results of El Khoury <i>et al.</i> [247] and results with only HIT injected at the inlet are also represented for comparison. | 141 |

| | |
|--|-----|
| 6.28. Evolution of $\bar{U}^+ = \bar{w}/u_\tau$ as a function of r^+ at $z = 7D$ for three values of L_e when injecting non-homogeneous and anisotropic turbulence. DNS results of El Khoury <i>et al.</i> [247] and results with HIT injected at the inlet coupled with artificial tripping for $C_D = 1.5$ are also represented for comparison. | 142 |
| 6.29. Evolution of RMS velocities non-dimensionalized by u_τ as a function of r^+ at $z = 7D$ for three values of L_e when injecting non-homogeneous and anisotropic turbulence. DNS results of El Khoury <i>et al.</i> [247] and results with only HIT injected at the inlet are also represented for comparison. | 143 |
| 6.30. Evolution of mean pressure and RMS of pressure at channel center ($r = 0$) along pipe axis for HIT only, HIT coupled with artificial tripping for $C_D = 1.5$ and non-homogeneous anisotropic turbulence with $L_e = D/4$ mm cases. | 144 |
| 7.1. Initial values of T , u and P at SP for both CONS and TUPY approaches. | 152 |
| 7.2. Profiles of T , u and P at SP after 10^4 iterations for multi-species cases with either CONS or TUPY approaches, and mono-species case with CONS approach. | 153 |
| 7.3. Initial profiles of T , u and P at OP for multi-species cases with either CONS or TUPY approaches and mono-species case with CONS approach. | 154 |
| 7.4. Profiles of T , u and P at SP at 0, 1000 and 2000 time iterations for the multi-species case with the TUPY approach when $u_c = 20 \text{ m.s}^{-1}$ | 154 |
| 7.5. Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4/Air-2S-BFER chemical scheme at $\phi = 0.8$ | 156 |
| 7.6. Comparison of major species mass fractions profiles between CANTERA and JAGUAR at $p = 4$ for a 1D methane-air premixed flame using the CH4/Air-2S-BFER chemical scheme at $\phi = 0.8$ | 157 |
| 7.7. Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$ | 159 |
| 7.8. Comparison of species mass fractions profiles of H_2 , H , O and OH between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$ | 159 |
| 7.9. Comparison of species mass fractions profiles of HO_2 , H_2O_2 , CH_2O and C_2H_4 between CANTERA, AVBP and JAGUAR for a 1D methane-air premixed flame using the CH4_16_250_10_QC ARC mechanism at $\phi = 1.0$. Profiles are shown between $x_{min} = 5.85 \text{ mm}$ and $x_{max} = 8.5 \text{ mm}$ for clarity. | 160 |
| 7.10. Comparison of ρ , u , P and T profiles between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$ | 161 |
| 7.11. Comparison of species mass fractions profiles of H_2 , H , O_2 and OH between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$ | 161 |
| 7.12. Comparison of species mass fractions profiles of O , H_2O , HO_2 and H_2O_2 between CANTERA, AVBP and JAGUAR for a 1D hydrogen-air premixed flame using the H2AIR_9_21_0_SD skeletal mechanism at $\phi = 1.0$. Profiles of HO_2 and H_2O_2 are shown between $x_{min} = 3 \text{ mm}$ and $x_{max} = 4 \text{ mm}$ for clarity reasons. | 162 |
| 7.13. Computational domain and boundary conditions for the 2D burner case. | 163 |
| 7.14. 2D heat release rate field obtained with JAGUAR for the 2D burner case. | 164 |
| 7.15. Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $x = 10.1 \text{ mm}$ along y -axis for the 2D burner case. | 164 |
| 7.16. Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $x = 12 \text{ mm}$ along y -axis for the 2D burner case. | 165 |

| | |
|--|-----|
| 7.17. Comparison between JAGUAR and AVBP of temperature, heat release rate, axial and vertical velocity profiles at $y = 0$ mm along x -axis for the 2D burner case. | 165 |
| 7.18. Computational domain and boundary conditions for the 2D circular flame case. | 166 |
| 7.19. 2D temperature fields obtained with JAGUAR (top) and AVBP (middle), and comparison of a longitudinal cut of temperature at $y = 0$ along x -axis, at $t = 2$ ms, $t = 4$ ms and $t = 6$ ms. | 167 |
| 7.20. Evolution with time of the consumption speed for JAGUAR and AVBP simulations. | 168 |
| 7.21. Comparison of the steady 2D heat release rate field between JAGUAR (bottom) and AVBP (top) fine solutions for the 2D burner case. | 169 |
| 7.22. Comparison of the steady heat release rate profiles at $x = 12$ mm between JAGUAR and AVBP fine solutions for the 2D burner case. | 169 |
| 7.23. Evolution of $\epsilon^{\omega_T}/\omega_{T,ref}^{max}$ with respect to p for two numbers of DOF. The values obtained with the LW and TTGC schemes at these numbers of DOF are also shown. | 170 |
| 7.24. Field of $\tanh(\alpha\tilde{\theta}_e)$ obtained from a uniform $p = 2$ simulation on the mesh with 1236 elements and $\alpha = 100$ | 173 |
| 7.25. Polynomial degree distribution for a calculation with $p_{min} = 2$ and $p_{max} = 4$ on the mesh with 1236 elements. | 174 |
| 7.26. Steady 2D heat release rate field obtained with a polynomial order distribution $2 \leq p \leq 4$ on the mesh with 1236 elements. | 174 |
| 7.27. Comparison of the steady heat release rate profiles at $x = 12$ mm, obtained on the same mesh with 1236 elements, between a computation at uniform $p = 2$, varying $2 \leq p \leq 4$ and uniform $p = 4$ | 175 |
| 7.28. Comparison of HRR and Y_{CO} between JAGUAR calculations at $p = 4$, $p = 5$ and $p = 6$ when $n_{pts}^{ff} \approx 4$ and the original diffusion scheme is employed and the AVBP simulation. CANTERA solution is added as reference. | 177 |
| 7.29. Comparison of the CO mass fraction between calculations at $p = 4$, $p = 5$ and $p = 6$ with either TUPY or ρ UPY approach at FP on a 1D premixed flame with $n_{pts}^{ff} \approx 4$ | 177 |
| 7.30. Comparison of the CO mass fraction obtained with different pairs of diffusion schemes and correction functions when using the SDLIFT formulation on a 1D premixed flame with $n_{pts}^{ff} \approx 4$ | 179 |
| 7.31. Comparison of Y_{CH_2O} , $Y_{C_2H_4}$, Y_H and Y_{HO_2} profiles between original and several SDLIFT formulations on a 1D CH_4 /Air premixed flame using an ARC mechanism with $n_{pts}^{ff} \approx 4$. CANTERA solution is added as reference and AVBP solution is also shown for comparison. | 180 |
| 7.32. Evolution of $S_c(t)$ for JAGUAR and AVBP simulations when $n_{pts}^{ff} = 16$ | 180 |
| 7.33. Cut of temperature profile at $y = 0$ along x -axis obtained with JAGUAR and AVBP at $t = 6$ ms when $n_{pts}^{ff} = 2$, $n_{pts}^{ff} = 4$ and $n_{pts}^{ff} = 8$ | 181 |
| 7.34. Evolution of $S_c(t)$ for JAGUAR and AVBP simulations when $n_{pts}^{ff} = 2$, $n_{pts}^{ff} = 4$ and $n_{pts}^{ff} = 8$ | 182 |
| 8.1. Comparison of heat release rate profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH_4 /Air-2S-BFER chemical scheme at $\phi = 0.8$. Non-thickened CANTERA solution is also plotted for comparison. | 186 |
| 8.2. Comparison of temperature profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH_4 /Air-2S-BFER chemical scheme at $\phi = 0.8$ in the x^* -space. Non-thickened CANTERA solution in the x -space is plotted for comparison. | 186 |
| 8.3. Comparison of fictive species mass fraction profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using the CH_4 /Air-2S-BFER chemical mechanism at $\phi = 0.8$ | 187 |

| | |
|--|-----|
| 8.4. Comparison of temperature profiles between AVBP and JAGUAR for a thickened 1D premixed methane-air flame using an ARC chemical mechanism at $\phi = 1.0$ in the x^* -space. Non-thickened CANTERA solution in the x -space is plotted for comparison. | 188 |
| 8.5. Comparison between JAGUAR (top) and AVBP (bottom) of the 2D heat release rate field for the 2D burner case using the TF model. | 189 |
| 8.6. Comparison between JAGUAR and AVBP of temperature and heat release rate profiles at $x = 12$ mm along y -axis for the 2D burner case using the TF model. | 189 |
| 8.7. Contour plot (left) and radial profile along $y = 0$ for $r \geq 0$ (right) of u_θ when $R_c = 6$ mm and $u_\theta^{max} = 1$ m.s $^{-1}$ | 190 |
| 8.8. Comparison between $\nabla_{\mathbf{x}}\mathbf{u}$ obtained with the methodology of paragraph 8.1.5.3 and the L^2 -norm of Eq. (8.8). | 192 |
| 8.9. Sketch of the Cambridge burner with associated dimensions and injected streams. The white area corresponds to the computational domain. | 194 |
| 8.10. Radial profiles of the statistics of axial and radial velocities at different distances from the burner exit for the non-reacting case SwBc. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Mercier [286] (—) and experimental data [298] (•). | 197 |
| 8.11. Evolution of Pope's criterion, computed with the Vreman model [157] using $C_{SGS} = 0.10$, on the $p = 2$ discretization in the burner mid-section. | 198 |
| 8.12. Contour plots in the burner mid-section of instantaneous temperature T (top-left), heat release rate $\dot{\omega}_T$ (top-right), equivalence ratio ϕ (bottom-left) and axial velocity (bottom-right) obtained with JAGUAR for the $p = 3$ case. | 198 |
| 8.13. Radial profiles of the statistics of axial and radial velocities at different distances from the burner exit. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Proch and Kempf [300] (—) and experimental data [298] (•). | 200 |
| 8.14. Radial profiles of the statistics of temperature and of mean equivalence ratio at different distances from the burner exit. Legends: JAGUAR $p = 2$ (—), JAGUAR $p = 3$ (—), numerical results of Proch and Kempf [300] (—) and experimental data [296](•). | 201 |
| 8.15. Comparison of temperature fluctuations $\sqrt{T'^2}$ at different distances from the burner exit in the $p = 2$ discretization case for both $n_{pts}^{ff} = 7$ (original computation) and $n_{pts}^{ff} = 10$. Legends: JAGUAR $n_{pts}^{ff} = 7$ (—), JAGUAR $n_{pts}^{ff} = 10$ (—), numerical results of Proch and Kempf [300] (—) and experimental data [296] (•). | 203 |
| 8.16. Comparison of polynomial distributions obtained with different sensors on the Cambridge burner when $p_{min} = 2$ and $p_{max} = 4$. The computation of sensor values is done using a first run at constant $p = 2$ | 203 |
| 8.17. Sketch of the VOLVO burner in the xy -plane with associated dimensions and locations of experimental data in both x and x^* spaces. | 204 |
| 8.18. Vortex shedding structure for the non-reacting case visualized using an iso-contour of Q -criterion at 7×10^7 s $^{-1}$ colored by instantaneous transverse velocity. | 207 |
| 8.19. \bar{u}/U_b along $y = 0$ in the non-reacting case. | 207 |
| 8.20. y -profiles of \bar{u}/U_b in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends. | 207 |
| 8.21. y -profiles of \bar{v}/U_b in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends. | 208 |
| 8.22. y -profiles of $\sqrt{u'^2}/U_b$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends. | 208 |
| 8.23. y -profiles of $\sqrt{v'^2}/U_b$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends. | 209 |
| 8.24. y -profiles of $u'v'/U_b^2$ in the non-reacting case at $x_1 - x_5$. See Figure 8.19 for legends. | 209 |
| 8.25. Evolution of Pope's criterion, computed with the Vreman model [157] using $C_{SGS} = 0.10$, on the $p = 3$ discretization near the bluff-body region. | 209 |
| 8.26. View of the mesh elements near the bluff-body region for the reacting case in the xy -plane. | 210 |

| | |
|--|-----|
| 8.27. Polynomial degree distribution near the bluff-body region for the reacting case in the xy -plane. | 210 |
| 8.28. Iso-surface of progress variable $c = 0.5$ colored by instantaneous wrinkling factor. . . . | 211 |
| 8.29. \bar{u}/U_b along $y = 0$ in the reacting case. | 212 |
| 8.30. y -profiles of \bar{u}/U_b in the reacting case at $x_1 - x_5$ obtained with JAGUAR. | 213 |
| 8.31. y -profiles of \bar{v}/U_b in the reacting case at $x_1 - x_5$ obtained with JAGUAR. | 213 |
| 8.32. y -profiles of $\sqrt{u'^2}/U_b$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR. | 213 |
| 8.33. y -profiles of $\sqrt{v'^2}/U_b$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR. | 214 |
| 8.34. y -profiles of $u'v'/U_b^2$ in the reacting case at $x_1 - x_5$ obtained with JAGUAR. | 214 |
| 8.35. y -profiles of \bar{T}/T_{in} in the reacting case at x_4, x'_4 and x'_5 obtained with JAGUAR. . . . | 214 |
| C.1. Legendre polynomials and Gauss-Legendre points on $[-1, 1]$ | 236 |
| C.2. Radau polynomials and points on $I = [-1, 1]$ for $1 \leq k \leq 4$ | 238 |

List of Tables

| | | |
|------|--|-----|
| 2.1. | Pre-exponential constant, activation energy and reaction exponents for the CH ₄ /Air-2S-BFER chemical scheme. | 24 |
| 2.2. | Coefficient values for $f_1(\phi)$ and $f_2(\phi)$ for the CH ₄ /Air-2S-BFER. | 24 |
| 2.3. | Laws for dynamic viscosity as a function of temperature T only. | 31 |
| 2.4. | Values of μ_{ref} , T_{ref} and n used in the power law formula for CH ₄ /Air-2S-BFER, CH4_16_250_10_QC and H2AIR_9_21_0_SD mechanisms. | 31 |
| 2.5. | Values of Prandtl numbers for CH ₄ /Air-2S-BFER, CH4_16_250_10_QC and H2AIR_9_21_0_SD mechanisms. | 32 |
| 4.1. | General form of a Butcher tableau. | 82 |
| 6.1. | Regions of the flow and their characteristics in a turbulent channel based on y^+ values. Adapted from Table 7.1 in Pope [150]. | 106 |
| 6.2. | Summary of the values taken by κ_e , ϵ and L_f for both Passot-Pouquet and von Karman-Pao spectrums. | 115 |
| 6.3. | Riemann solver, diffusion scheme and time-integration scheme used for all simulations of injection of turbulence through a periodic channel. | 122 |
| 6.4. | Turbulent parameters used for the validation of the injection of HIT in a periodic channel. | 122 |
| 6.5. | Turbulent parameters used for the injection of HIT in a periodic channel with the VKP spectrum. | 123 |
| 6.6. | Iteration cost as a function of N in the case of the periodic channel. All computations were done on 600 processors provided by the same machine. | 126 |
| 6.7. | Turbulent parameters used for the HIT injection in a turbulent wall-bounded channel flow. | 131 |
| 6.8. | Turbulent parameters used for the HIT injection in a turbulent wall-bounded pipe flow. | 138 |
| 7.1. | Characteristics of the 1D methane-air premixed flame using the CH ₄ /Air-2S-BFER scheme. T^b , S_L^0 and δ_L^0 are the values given by CANTERA. | 156 |
| 7.2. | Values of N_e for each degree p to keep the DOF number around 400. | 156 |
| 7.3. | Comparison of flame speed obtained with AVBP and JAGUAR (abbreviated by JAG) at different polynomial degrees p for a 1D methane-air premixed flame using the CH ₄ /Air-2S-BFER chemical scheme at $\phi = 0.8$ | 157 |
| 7.4. | Characteristics of the 1D methane-air premixed flame using an ARC mechanism. T^b , S_L^0 and δ_L^0 are the values given by the CANTERA solution. | 158 |
| 7.5. | Characteristics of the 1D hydrogen-air premixed flame using a skeletal mechanism. T^b , S_L^0 and δ_L^0 are the values given by the CANTERA solution. | 160 |
| 7.6. | Comparison between JAGUAR and AVBP of burnt gas temperatures T^b and maximum heat release rates $\dot{\omega}_T^{max}$ obtained in the 2D burner case using the CH ₄ /Air-2S-BFER scheme at $\phi = 0.8$. CANTERA values obtained on a 1D flame in the same conditions are also shown for reference. | 164 |
| 7.7. | Values of κ [$\mu s/ite/DOF$] and τ [s] for both AVBP and JAGUAR simulations with different values of N_e and p to keep the number of DOF around 30900 ($\Delta_{DOF} = 5.09 \times 10^{-5}$ m and $n_{pts}^{ff} = 8$). | 172 |

| | |
|--|-----|
| 7.8. Values of κ [$\mu s/ite/DOF$] and τ [s] for both AVBP and JAGUAR simulations with different values of N_e and p to keep the number of DOF around 20100 ($\Delta_{DOF} = 6.31 \times 10^{-5}$ m and $n_{pts}^{ff} = 6$). | 172 |
| 7.9. Turnaround times for $\epsilon^{\dot{\omega}_T}/\dot{\omega}_{T,ref}^{max} = 0.02$ with different discretization. | 173 |
| 7.10. Values of DOF, κ [$\mu s/ite/DOF$] and τ [s] for JAGUAR simulations of the 2D burner on the same mesh but with different polynomial distributions within elements. | 175 |
| 7.11. Pairs of (N_e, p) values to keep DOF ≈ 185 | 176 |
| 8.1. Flame sensor parameters for the dynamic thickening of the one-dimensional methane-air premixed flame using the CH4/Air-2S-BFER chemical mechanism at $\phi = 0.8$ | 185 |
| 8.2. Comparison of flame speeds between JAGUAR and AVBP for dynamic, constant thickening and without any thickening for a 1D premixed methane-air flame using the CH4/Air-2S-BFER chemical mechanism at $\phi = 0.8$ | 187 |
| 8.3. Flame sensor parameters for the dynamic thickening of the one-dimensional methane-air premixed flame using an ARC chemical mechanism at $\phi = 1.0$ | 187 |
| 8.4. Comparison of flame speeds between JAGUAR and AVBP for dynamic, constant thickening and without any thickening for a thickened 1D premixed methane-air flame using an ARC chemical mechanism at $\phi = 1.0$ | 188 |
| 8.5. Operating conditions considered in this work for the simulations of the Cambridge burner. | 194 |
| 8.6. Iteration cost κ [$\mu s.ite^{-1}.DOF^{-1}$] for SwBc and SwB1 cases for $p = 2$ and $p = 3$ discretizations with different numerical treatments. The SwBc case with $N_s = 0$ was added to estimate the cost of adding one transport equation. All computations were done on 760 processors provided by the same cluster. | 199 |
| 8.7. Values of N_e , p and DOF for the three different discretizations used for the non-reacting case of the VOLVO burner at $\phi_{in} = 0.65$ | 205 |
| 8.8. Summary numerical methods used in previous works and this work for comparison on the reacting VOLVO burner. Question tags indicate that no information is available. | 212 |
| 8.9. Iteration costs κ [$\mu s.ite^{-1}.DOF^{-1}$] for non-reacting and reacting VOLVO simulations with JAGUAR for the $p = 3$ discretization and AVBP. All computations were done on 760 processors provided by the same cluster. | 215 |
| E.1. Coefficients of matrices α and β of the SSP-ERK(3,3) scheme. | 245 |
| E.2. Butcher tableau of the SSP-ERK(3,3) scheme. | 245 |
| E.3. Coefficients of matrices α (top) and β (bottom) of the SSP-ERK(5,4) scheme. | 245 |
| E.4. Butcher tableau of the SSP-ERK(5,4) scheme. | 246 |

List of Algorithms

| | | |
|----|--|-----|
| 1. | Algorithm to compute \mathbf{M} at a given point | 229 |
| 2. | Algorithm to compute \mathbf{q} at a given point | 229 |
| 3. | Computation of gradients in Cartesian coordinates at ξ -FP with SDLIFT formulation | 230 |
| 4. | Kim and Lee algorithm to apply characteristic boundary conditions in generalized co-ordinates | 231 |
| 5. | Fievet et <i>al.</i> algorithm to apply characteristic boundary conditions in generalized co-ordinates | 231 |
| 6. | Algorithm for computing the turbulent velocity vector using a synthetic random Fourier method | 232 |
| 7. | TFLES model using relaxation flame sensor for a SD discretization | 233 |

Bibliography

- [1] H.-O. Pörtner, D. C. Roberts, H. Adams, C. Adler, P. Aldunce, E. Ali, R. A. Begum, R. Betts, R. B. Kerr, R. Biesbroek, et al., “Climate change 2022: Impacts, adaptation and vulnerability,” IPCC Sixth Assessment Report, 2022. pages 2
- [2] “IEA (2021), net zero by 2050, IEA, paris.” <https://www.iea.org/reports/net-zero-by-2050>. pages 2
- [3] T. Poinso and D. Veynante, Theoretical and numerical combustion. RT Edwards, Inc., 2005. pages 3, 18, 19, 26, 27, 28, 29, 37, 38, 39, 41, 95, 157
- [4] D. Anderson, J. C. Tannehill, and R. H. Pletcher, Computational fluid mechanics and heat transfer. Taylor & Francis, 2016. pages 3, 66, 67
- [5] Z. J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H. T. Huynh, et al., “High-order CFD methods: current status and perspective,” International Journal for Numerical Methods in Fluids, vol. 72, no. 8, pp. 811–845, 2013. pages 3, 5, 54, 170
- [6] S. K. Lele, “Compact finite difference schemes with spectral-like resolution,” Journal of Computational Physics, vol. 103, no. 1, pp. 16–42, 1992. pages 4
- [7] K. Richards, P. Senecal, and E. Pomraning, “Convergent science.” <https://convergecfcd.com/>, 2023. Version 3.0*. pages 4
- [8] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, “Uniformly high order accurate essentially non-oscillatory schemes, III,” in Upwind and high-resolution schemes, pp. 218–290, Springer, 1987. pages 5
- [9] C.-W. Shu and S. Osher, “Efficient implementation of essentially non-oscillatory shock-capturing schemes,” Journal of Computational Physics, vol. 77, no. 2, pp. 439–471, 1988. pages 83
- [10] R. Abgrall, “On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation,” Journal of Computational Physics, vol. 114, no. 1, pp. 45–58, 1994. pages 5
- [11] X.-D. Liu, S. Osher, and T. Chan, “Weighted essentially non-oscillatory schemes,” Journal of Computational Physics, vol. 115, no. 1, pp. 200–212, 1994. pages 5
- [12] T. Barth and P. Frederickson, “Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction,” in 28th aerospace sciences meeting, p. 13, 1990. pages 5
- [13] M. Delanaye and Y. Liu, “Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral grids,” in 14th Computational Fluid Dynamics Conference, p. 3259, 1999. pages 5
- [14] L. Fu, X. Y. Hu, and N. A. Adams, “A family of high-order targeted ENO schemes for compressible-fluid simulations,” Journal of Computational Physics, vol. 305, pp. 333–359, 2016. pages 5
- [15] J.-B. Chapelier, M. De La Llave Plata, F. Renac, and E. Lamballais, “Evaluation of a high-order discontinuous Galerkin method for the DNS of turbulent flows,” Computers & Fluids, vol. 95, pp. 210–226, 2014. pages 5

-
- [16] M. de la Llave Plata, V. Couaillier, and M.-C. Le Pape, “On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence,” Computers & Fluids, vol. 176, pp. 320–337, 2018. pages 5
- [17] W. H. Reed and T. Hill, “Triangular mesh methods for the neutron transport equation,” tech. rep., Los Alamos Scientific Lab., N. Mex.(USA), 1973. pages 5
- [18] B. Cockburn, S. Hou, and C.-W. Shu, “The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case,” Mathematics of Computation, vol. 54, no. 190, pp. 545–581, 1990. pages 5
- [19] B. Cockburn and C.-W. Shu, “The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems,” Journal of Computational Physics, vol. 141, no. 2, pp. 199–224, 1997. pages
- [20] B. Cockburn and C.-W. Shu, “Runge-Kutta discontinuous Galerkin methods for convection-dominated problems,” Journal of Scientific Computing, vol. 16, no. 3, pp. 173–261, 2001. pages 5, 172
- [21] C. Anghan, S. Dave, S. Saincher, and J. Banerjee, “Direct numerical simulation of transitional and turbulent round jets: Evolution of vortical structures and turbulence budget,” Physics of Fluids, vol. 31, no. 6, p. 065105, 2019. pages 5
- [22] A. D. Beck, T. Bolemann, D. Flad, H. Frank, G. J. Gassner, F. Hindenlang, and C.-D. Munz, “High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations,” International Journal for Numerical Methods in Fluids, vol. 76, no. 8, pp. 522–548, 2014. pages 5
- [23] F. Renac, M. de la Llave Plata, E. Martin, J.-B. Chapelier, and V. Couaillier, “Aghora: a high-order DG solver for turbulent flow simulations,” in IDIHOM: Industrialization of High-Order Methods-A Top-Down Approach, pp. 315–335, Springer, 2015. pages 5
- [24] G. Billet, J. Ryan, and M. Borrel, “A Runge Kutta discontinuous Galerkin approach to solve reactive flows on conforming hybrid grids: the parabolic and source operators,” Computers & Fluids, vol. 95, pp. 98–115, 2014. pages 5, 8, 83
- [25] Y. Lv and M. Ihme, “Discontinuous Galerkin method for multicomponent chemically reacting flows and combustion,” Journal of Computational Physics, vol. 270, pp. 105–137, 2014. pages 8, 75, 150
- [26] Y. Lv and M. Ihme, “High-order discontinuous Galerkin method for applications to multicomponent and chemically reacting flows,” Acta Mechanica Sinica, vol. 33, no. 3, pp. 486–499, 2017. pages 8, 204, 211, 212
- [27] R. F. Johnson and A. D. Kercher, “A conservative discontinuous Galerkin discretization for the chemically reacting Navier-Stokes equations,” Journal of Computational Physics, vol. 423, p. 109826, 2020. pages 8, 9, 75, 83, 150, 168
- [28] J. Du and Y. Yang, “High-order bound-preserving discontinuous Galerkin methods for multicomponent chemically reacting flows,” Journal of Computational Physics, vol. 469, p. 111548, 2022. pages 5, 7, 8
- [29] R. Biswas, K. D. Devine, and J. E. Flaherty, “Parallel, adaptive finite element methods for conservation laws,” Applied Numerical Mathematics, vol. 14, no. 1-3, pp. 255–283, 1994. pages 5

- [30] M. Yu, Z. J. Wang, and Y. Liu, “On the accuracy and efficiency of discontinuous Galerkin, Spectral Difference and Correction Procedure via Reconstruction methods,” Journal of Computational Physics, vol. 259, pp. 70–95, 2014. pages 5, 54, 171
- [31] J. S. Hesthaven and T. Warburton, Nodal discontinuous Galerkin methods: algorithms, analysis, and applications. Springer Science & Business Media, 2007. pages 5
- [32] D. A. Kopriva and G. Gassner, “On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods,” Journal of Scientific Computing, vol. 44, no. 2, pp. 136–155, 2010. pages 6
- [33] W. Li, J. Pan, and Y.-X. Ren, “The discontinuous Galerkin spectral element methods for compressible flows on two-dimensional mixed grids,” Journal of Computational Physics, vol. 364, pp. 314–346, 2018. pages 6
- [34] Z. J. Wang, “Spectral (finite) volume method for conservation laws on unstructured grids: Basic formulation,” Journal of Computational Physics, vol. 178, no. 1, pp. 210–251, 2002. pages 6
- [35] Z. J. Wang and Y. Liu, “Spectral (finite) volume method for conservation laws on unstructured grids: II. Extension to two-dimensional scalar equation,” Journal of Computational Physics, vol. 179, no. 2, pp. 665–697, 2002. pages
- [36] Z. J. Wang, L. Zhang, and Y. Liu, “Spectral (finite) volume method for conservation laws on unstructured grids IV: Extension to two-dimensional systems,” Journal of Computational Physics, vol. 194, no. 2, pp. 716–741, 2004. pages 6
- [37] Y. Liu, M. Vinokur, and Z. J. Wang, “Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems,” Journal of Computational Physics, vol. 212, no. 2, pp. 454–472, 2006. pages 6
- [38] Y. Sun, Z. J. Wang, and Y. Liu, “Spectral (finite) volume method for conservation laws on unstructured grids VI: Extension to viscous flow,” Journal of Computational Physics, vol. 215, no. 1, pp. 41–58, 2006. pages 6
- [39] M. Yang, R. Harris, Z.-J. Wang, and Y. Liu, “Efficient quadrature-free 3D high-order spectral volume method on unstructured grids,” in 18th AIAA Computational Fluid Dynamics Conference, p. 4325, 2007. pages 6
- [40] R. Harris, Z. J. Wang, and Y. Liu, “Efficient quadrature-free high-order spectral volume method on unstructured grids: Theory and 2D implementation,” Journal of Computational Physics, vol. 227, no. 3, pp. 1620–1642, 2008. pages 6
- [41] K. Van den Abeele and C. Lacor, “An accuracy and stability study of the 2D spectral volume method,” Journal of Computational Physics, vol. 226, no. 1, pp. 1007–1026, 2007. pages 6
- [42] D. A. Kopriva and J. H. Kolas, “A conservative staggered-grid chebyshev multidomain method for compressible flows,” Journal of Computational Physics, vol. 125, no. 1, pp. 244–261, 1996. pages 6, 54
- [43] D. A. Kopriva, “A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method,” Journal of Computational Physics, vol. 128, no. 2, pp. 475–488, 1996. pages 6, 80, 81, 84
- [44] Y. Liu, M. Vinokur, and Z. Wang, “Spectral Difference method for unstructured grids I: basic formulation,” Journal of Computational Physics, vol. 216, no. 2, pp. 780–801, 2006. pages 6

-
- [45] Z. J. Wang, Y. Liu, G. May, and A. Jameson, "Spectral Difference method for unstructured grids II: extension to the Euler equations," Journal of Scientific Computing, vol. 32, no. 1, pp. 45–71, 2007. pages 6
- [46] G. May and A. Jameson, "A Spectral Difference method for the Euler and Navier-Stokes equations on unstructured meshes," in 44th AIAA aerospace sciences meeting and exhibit, p. 304, 2006. pages 6
- [47] Y. Sun, Z. J. Wang, and Y. Liu, "High-order multidomain Spectral Difference method for the Navier-Stokes equations on unstructured hexahedral grids," Communications in Computational Physics, vol. 2, no. 2, pp. 310–333, 2007. pages 6, 59, 65, 75, 76, 84, 176, 182, 195, 206, 216, 218
- [48] W. Chen, Y. Ju, and C. Zhang, "A collocated-grid Spectral Difference method for compressible flows," Computers & Fluids, vol. 196, p. 104341, 2020. pages 6
- [49] K. Van den Abeele, C. Lacor, and Z. Wang, "On the stability and accuracy of the Spectral Difference method," Journal of Scientific Computing, vol. 37, no. 2, pp. 162–188, 2008. pages 6, 54, 55
- [50] A. Jameson, "A proof of the stability of the Spectral Difference method for all orders of accuracy," Journal of Scientific Computing, vol. 45, no. 1-3, pp. 348–358, 2010. pages 6, 55
- [51] G. May and J. Schöberl, "Analysis of a Spectral Difference scheme with flux interpolation on Raviart-Thomas elements," Aachen Institute for Advanced Study in Computational Engineering Science, Aachen, 2010. pages 6
- [52] A. Balan, G. May, and J. Schöberl, "A stable high-order Spectral Difference method for hyperbolic conservation laws on triangular elements," Journal of Computational Physics, vol. 231, no. 5, pp. 2359–2375, 2012. pages 6, 176
- [53] M. Li, Z. Qiu, C. Liang, M. Sprague, M. Xu, and C. A. Garriss, "A new high-order Spectral Difference method for simulating viscous flows on unstructured grids with mixed-element meshes," Computers & Fluids, vol. 184, pp. 187–198, 2019. pages
- [54] Z. Qiu, B. Zhang, C. Liang, and M. Xu, "A high-order solver for simulating vortex-induced vibrations using the sliding-mesh Spectral Difference method and hybrid grids," International Journal for Numerical Methods in Fluids, vol. 90, no. 4, pp. 171–194, 2019. pages 6
- [55] A. Veilleux, G. Puigt, H. Deniau, and G. Daviller, "A stable Spectral Difference approach for computations with triangular and hybrid grids up to the 6th order of accuracy," Journal of Computational Physics, p. 110774, 2021. pages 7, 10
- [56] A. Veilleux, G. Puigt, H. Deniau, and G. Daviller, "Stable Spectral Difference Approach Using Raviart-Thomas Elements for 3D Computations on Tetrahedral Grids," Journal of Scientific Computing, vol. 91, no. 1, pp. 1–22, 2022. pages 7, 10
- [57] H. T. Huynh, "A Flux Reconstruction approach to high-order schemes including discontinuous Galerkin methods," in 18th AIAA Computational Fluid Dynamics Conference, p. 4079, 2007. pages 7, 60, 61, 237, 238
- [58] Y. Allaneau and A. Jameson, "Connections between the filtered discontinuous Galerkin method and the Flux Reconstruction approach to high-order discretizations," Computer Methods in Applied Mechanics and Engineering, vol. 200, no. 49-52, pp. 3628–3636, 2011. pages 7

- [59] P. E. Vincent, P. Castonguay, and A. Jameson, “A new class of high-order energy stable flux reconstruction schemes,” Journal of Scientific Computing, vol. 47, no. 1, pp. 50–72, 2011. pages 7, 62
- [60] A. Sheshadri and A. Jameson, “On the stability of the flux reconstruction schemes on quadrilateral elements for the linear advection equation,” Journal of Scientific Computing, vol. 67, no. 2, pp. 769–790, 2016. pages 7
- [61] A. Jameson, P. E. Vincent, and P. Castonguay, “On the non-linear stability of flux reconstruction schemes,” Journal of Scientific Computing, vol. 50, no. 2, pp. 434–445, 2012. pages 7, 54
- [62] P. E. Vincent, P. Castonguay, and A. Jameson, “Insights from von Neumann analysis of high-order flux reconstruction schemes,” Journal of Computational Physics, vol. 230, no. 22, pp. 8134–8154, 2011. pages 7
- [63] K. Asthana and A. Jameson, “High-order flux reconstruction schemes with minimal dispersion and dissipation,” Journal of Scientific Computing, vol. 62, no. 3, pp. 913–944, 2015. pages 7
- [64] P. Castonguay, P. E. Vincent, and A. Jameson, “A new class of high-order energy stable flux reconstruction schemes for triangular elements,” Journal of Scientific Computing, vol. 51, no. 1, pp. 224–256, 2012. pages 7
- [65] D. Williams and A. Jameson, “Energy stable flux reconstruction schemes for advection–diffusion problems on tetrahedra,” Journal of Scientific Computing, vol. 59, no. 3, pp. 721–759, 2014. pages 7
- [66] H. Gao and Z. Wang, “A high-order lifting collocation penalty formulation for the Navier-Stokes equations on 2D mixed grids,” in 19th AIAA Computational Fluid Dynamics, p. 3784, American Institute of Aeronautics and Astronautics, 2009. pages 7
- [67] T. Haga, H. Gao, and Z. J. Wang, “A high-order unifying discontinuous formulation for the Navier-Stokes equations on 3D mixed grids,” Mathematical Modelling of Natural Phenomena, vol. 6, no. 3, pp. 28–56, 2011. pages 7
- [68] V. Singh and S. Frankel, “Wall-Modeled Implicit LES of Transitional Flows using Variable-Order Flux Reconstruction Method,” in AIAA Scitech 2019 Forum, p. 0645, 2019. pages 7
- [69] T. Haga, S. Tsutsumi, S. Kawai, and R. Takaki, “Large-eddy simulation of a supersonic jet using high-order Flux Reconstruction scheme,” in 53rd AIAA Aerospace Sciences Meeting, p. 0831, 2015. pages 7
- [70] J. Romero, K. Asthana, and A. Jameson, “A simplified formulation of the flux reconstruction method,” Journal of Scientific Computing, vol. 67, no. 1, pp. 351–374, 2016. pages 7
- [71] J. Romero, F. D. Witherden, and A. Jameson, “A direct flux reconstruction scheme for advection-diffusion problems on triangular grids,” Journal of Scientific Computing, vol. 73, no. 2, pp. 1115–1144, 2017. pages 7
- [72] B. Larrouturou, “How to preserve the mass fractions positivity when computing compressible multi-component flows,” Journal of Computational Physics, vol. 95, no. 1, pp. 59–84, 1991. pages 7
- [73] G. Strang, “On the construction and comparison of difference schemes,” SIAM journal on numerical analysis, vol. 5, no. 3, pp. 506–517, 1968. pages 7, 83

-
- [74] S. Blanchard, Q. Cazères, and B. Cuenot, “Chemical modeling for methane oxy-combustion in liquid rocket engines,” *Acta Astronautica*, vol. 190, pp. 98–111, 2022. pages 7, 83
- [75] R. Abgrall, “How to prevent pressure oscillations in multicomponent flow calculations: a quasi conservative approach,” *Journal of Computational Physics*, vol. 125, no. 1, pp. 150–160, 1996. pages 8, 150, 153
- [76] R. Abgrall and S. Karni, “Computations of compressible multifluids,” *Journal of Computational Physics*, vol. 169, no. 2, pp. 594–623, 2001. pages 8
- [77] G. Billet and R. Abgrall, “An adaptive shock-capturing algorithm for solving unsteady reactive flows,” *Computers & fluids*, vol. 32, no. 10, pp. 1473–1495, 2003. pages 8
- [78] K.-M. Shyue, “An efficient shock-capturing algorithm for compressible multicomponent problems,” *Journal of Computational Physics*, vol. 142, no. 1, pp. 208–242, 1998. pages 8
- [79] K.-M. Shyue, “A fluid-mixture type algorithm for compressible multicomponent flow with van der Waals equation of state,” *Journal of Computational Physics*, vol. 156, no. 1, pp. 43–88, 1999. pages
- [80] K.-M. Shyue, “A fluid-mixture type algorithm for compressible multicomponent flow with Mie–Grüneisen equation of state,” *Journal of Computational Physics*, vol. 171, no. 2, pp. 678–707, 2001. pages
- [81] M. T. H. de Frahan, S. Varadan, and E. Johnsen, “A new limiting procedure for discontinuous Galerkin methods applied to compressible multiphase flows with shocks and interfaces,” *Journal of Computational Physics*, vol. 280, pp. 489–509, 2015. pages
- [82] T. Nonomura, S. Morizawa, H. Terashima, S. Obayashi, and K. Fujii, “Numerical (error) issues on compressible multicomponent flows using a high-order differencing scheme: weighted compact nonlinear scheme,” *Journal of Computational Physics*, vol. 231, no. 8, pp. 3181–3210, 2012. pages
- [83] T. Nonomura and K. Fujii, “Robust explicit formulation of weighted compact nonlinear scheme,” *Computers & Fluids*, vol. 85, pp. 8–18, 2013. pages
- [84] T. Nonomura and K. Fujii, “Characteristic finite-difference WENO scheme for multicomponent compressible fluid analysis: Overestimated quasi-conservative formulation maintaining equilibriums of velocity, pressure, and temperature,” *Journal of Computational Physics*, vol. 340, pp. 358–388, 2017. pages 8, 153
- [85] G. Billet and J. Ryan, “A Runge–Kutta discontinuous Galerkin approach to solve reactive flows: The hyperbolic operator,” *Journal of Computational Physics*, vol. 230, no. 4, pp. 1064–1083, 2011. pages 8, 75, 83, 150
- [86] J. Du and Y. Yang, “Third-order conservative sign-preserving and steady-state-preserving time integrations and applications in stiff multispecies and multireaction detonations,” *Journal of Computational Physics*, vol. 395, pp. 489–510, 2019. pages 8
- [87] S. Gottlieb, C.-W. Shu, and E. Tadmor, “Strong stability-preserving high-order time discretization methods,” *SIAM review*, vol. 43, no. 1, pp. 89–112, 2001. pages 8, 83
- [88] P. Gupta, K. Schwinn, G. Lodato, C. Slabaugh, and C. Scalo, “Numerical investigation of sustained planar detonation waves in a periodic domain,” in *2018 Fluid Dynamics Conference*, p. 3240, 2018. pages 8

- [89] H. Tofaili, G. Lodato, L. Vervisch, and P. Clavin, “One-dimensional dynamics of gaseous detonations revisited,” Combustion and Flame, vol. 232, p. 111535, 2021. pages 8, 191, 193
- [90] G. Lodato, “Characteristic modal shock detection for discontinuous finite element methods,” Computers & Fluids, vol. 179, pp. 309–333, 2019. pages 9, 83, 191, 192, 193
- [91] X. Zhang and C.-W. Shu, “On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes,” Journal of Computational Physics, vol. 229, no. 23, pp. 8918–8934, 2010. pages 9, 191, 192, 193
- [92] A. Peyvan, D. Li, J. Komperda, and F. Mashayek, “Oscillation-free nodal discontinuous spectral element method for the simulation of compressible multicomponent flows,” Journal of Computational Physics, vol. 452, p. 110921, 2022. pages 9, 75, 150
- [93] J. Bell, “AMR for low mach number reacting flow,” in Adaptive Mesh Refinement-Theory and Applications, pp. 203–221, Springer, 2005. pages 9
- [94] R. Hartmann and T. Leicht, “Higher order and adaptive DG methods for compressible flows,” 37th Advanced CFD Lecture Series: Recent developments in higher order methods and industrial application in aeronautics, vol. 2014, no. 3, pp. 1–156, 2014. pages 9, 168, 174
- [95] F. Naddei, M. de la Llave Plata, and V. Couaillier, “A comparison of refinement indicators for p-adaptive discontinuous Galerkin methods for the Euler and Navier-Stokes equations,” in 2018 AIAA Aerospace Sciences Meeting, p. 0368, 2018. pages 9, 79, 168
- [96] C. Hirsch, K. Hillewaert, R. Hartmann, V. Couaillier, J.-F. Boussuge, F. Chalot, S. Bosniakov, and W. Haase, “Tilda: Towards industrial LES/DNS in Aeronautics,” Springer International Publishing, 2021. pages 9, 10, 79, 80, 168, 174
- [97] G. R. McNamara and G. Zanetti, “Use of the Boltzmann equation to simulate lattice-gas automata,” Physical review letters, vol. 61, no. 20, p. 2332, 1988. pages 9
- [98] X. He and L.-S. Luo, “A priori derivation of the lattice Boltzmann equation,” Physical Review E, vol. 55, no. 6, p. R6333, 1997. pages 9
- [99] X. He and L.-S. Luo, “Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation,” Physical review E, vol. 56, no. 6, p. 6811, 1997. pages 9
- [100] M. E. Gleason, B. Duncan, J. Walter, A. Guzman, and Y.-C. Cho, “Comparison of computational simulation of automotive spinning wheel flow field with full width moving belt wind tunnel results,” SAE International Journal of Passenger Cars-Mechanical Systems, vol. 8, no. 2015-01-1556, pp. 275–293, 2015. pages 9
- [101] D. Casalino, A. Hazir, and A. Mann, “Turbofan broadband noise prediction using the lattice Boltzmann method,” AIAA Journal, vol. 56, no. 2, pp. 609–628, 2018. pages 9
- [102] M. R. Khorrami and E. Fares, “Simulation-based airframe noise prediction of a full-scale, full aircraft,” in 22nd AIAA/CEAS aeroacoustics conference, p. 2706, 2016. pages 9
- [103] A. Aniello, D. Schuster, P. Werner, J. Boussuge, M. Gatti, C. Mirat, L. Selle, T. Schuller, T. Poinot, and U. R  de, “Comparison of a finite volume and two Lattice Boltzmann solvers for swirled confined flows,” Computers & Fluids, vol. 241, p. 105463, 2022. pages 9
- [104] H. Touil, D. Ricot, and E. L  v  que, “Direct and large-eddy simulation of turbulent flows on composite multi-resolution grids by the lattice Boltzmann method,” Journal of Computational Physics, vol. 256, pp. 220–233, 2014. pages 9

-
- [105] F. Schornbaum and U. Rude, “Massively parallel algorithms for the lattice Boltzmann method on nonuniform grids,” SIAM Journal on Scientific Computing, vol. 38, no. 2, pp. C96–C126, 2016. pages 9
- [106] S. Marié, D. Ricot, and P. Sagaut, “Comparison between lattice Boltzmann method and Navier-Stokes high order schemes for computational aeroacoustics,” Journal of Computational Physics, vol. 228, no. 4, pp. 1056–1070, 2009. pages 9
- [107] F. Renard, Y. Feng, J.-F. Boussuge, and P. Sagaut, “Improved compressible hybrid lattice Boltzmann method on standard lattice for subsonic and supersonic flows,” Computers & Fluids, vol. 219, p. 104867, 2021. pages 9
- [108] S. Guo, Y. Feng, J. Jacob, F. Renard, and P. Sagaut, “An efficient lattice Boltzmann method for compressible aerodynamics on D3Q19 lattice,” Journal of Computational Physics, vol. 418, p. 109570, 2020. pages 9
- [109] Y. Feng, M. Tayyab, and P. Boivin, “A Lattice-Boltzmann model for low-Mach reactive flows,” Combustion and Flame, vol. 196, pp. 249–254, 2018. pages 9, 211
- [110] M. Tayyab, S. Zhao, Y. Feng, and P. Boivin, “Hybrid regularized lattice-Boltzmann modelling of premixed and non-premixed combustion processes,” Combustion and Flame, vol. 211, pp. 173–184, 2020. pages 9, 88, 211
- [111] M. Tayyab, S. Zhao, and P. Boivin, “Lattice-Boltzmann modeling of a turbulent bluff-body stabilized flame,” Physics of Fluids, vol. 33, no. 3, p. 031701, 2021. pages 9, 204, 205, 206, 211, 212
- [112] C. Liang, C. Cox, and M. Plesniak, “A comparison of computational efficiencies of Spectral Difference method and Correction Procedure via Reconstruction,” Journal of Computational Physics, vol. 239, pp. 138–146, 2013. pages 10, 168
- [113] C. Cox, W. Trojak, T. Dzanic, F. Witherden, and A. Jameson, “Accuracy, stability, and performance comparison between the Spectral Difference and Flux Reconstruction schemes,” Computers & Fluids, vol. 221, p. 104922, 2021. pages 10, 168
- [114] A. Cassagne, J.-F. Boussuge, and G. Puigt, High-order method for a new generation of Large Eddy Simulation solver. PhD thesis, PRACE, 2015. pages 10
- [115] A. Cassagne, J.-F. Boussuge, N. Villedieu, G. Puigt, I. d’Ast, and A. Genot, “JAGUAR: a new CFD code dedicated to massively parallel high-order LES computations on complex geometry,” in The 50th 3AF International Conference on Applied Aerodynamics (AERO 2015), 2015. pages 10, 202
- [116] A. Veilleux, Extension of the Spectral Difference method to simplex cells and hybrid grids. PhD thesis, Université de Toulouse, 2021. pages 10, 52, 53, 176
- [117] R. Fiévet, H. Deniau, and E. Piot, “Strong compact formalism for characteristic boundary conditions with discontinuous spectral methods,” Journal of Computational Physics, vol. 408, p. 109276, 2020. pages 10, 88, 91, 93, 94, 100, 231
- [118] R. Fievet, H. Deniau, J.-P. Brazier, and E. Piot, “Numerical study of hypersonic boundary-layer transition delay through second-mode absorption,” in AIAA SciTech 2020 Forum, p. 2061, 2020. pages 10

- [119] M. Sweeney, S. Hochgreb, M. Dunn, and R. Barlow, "A comparative analysis of flame surface density metrics in premixed and stratified flames," Proceedings of the Combustion Institute, vol. 33, no. 1, pp. 1419–1427, 2011. pages 11, 193
- [120] M. Sweeney, S. Hochgreb, and R. Barlow, "The structure of premixed and stratified low turbulence flames," Combustion and Flame, vol. 158, no. 5, pp. 935–948, 2011. pages 11, 193
- [121] A. Sjunnesson, C. Nelsson, and E. Max, "LDA measurements of velocities and turbulence in a bluff body stabilized flame," tech. rep., Swedish National Board for Industrial and Technical Development (NUTEK), 1991. pages 11, 204, 211
- [122] A. Sjunnesson, P. Henrikson, and C. Lofstrom, "CARS measurements and visualization of reacting flows in a bluff body stabilized flame," in 28th Joint Propulsion Conference and Exhibit, p. 3650, 1992. pages 11, 204, 211
- [123] R. Bilger, S. Stårner, and R. Kee, "On reduced mechanisms for methane air combustion in nonpremixed flames," Combustion and Flame, vol. 80, no. 2, pp. 135–149, 1990. pages 18, 19
- [124] R. J. Blint, "The relationship of the laminar flame width to flame speed," Combustion Science and Technology, vol. 49, no. 1-2, pp. 79–92, 1986. pages 20
- [125] R. J. Kee, F. M. Rupley, and J. A. Miller, "Chemkin-ii: A fortran chemical kinetics package for the analysis of gas-phase chemical kinetics," tech. rep., Sandia National Lab.(SNL-CA), Livermore, CA (United States), 1989. pages 21
- [126] D. G. Goodwin, H. K. Moffat, and R. L. Speth, "Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes." <http://www.cantera.org>, 2017. Version 2.3.0. pages 21, 155
- [127] C. Bowman, R. Hanson, D. Davidson, W. Gardiner Jr, V. Lissianski, G. Smith, D. Golden, M. Frenklach, H. Wang, and M. Goldenberg, "GRI-Mech version 2.11," tech. rep., 11/3, 1995. pages 23
- [128] C. Bowman, M. Frenklach, W. Gardiner, and G. Smith, "The GRI 3.0 chemical kinetic mechanism," 1999. pages 23
- [129] E. Ranzi, A. Frassoldati, A. Stagni, M. Pelucchi, A. Cuoci, and T. Faravelli, "Reduced kinetic schemes of complex reaction systems: fossil and biomass-derived transportation fuels," International Journal of Chemical Kinetics, vol. 46, no. 9, pp. 512–542, 2014. pages 23
- [130] B. Franzelli, E. Riber, M. Sanjosé, and T. Poinsot, "A two-step chemical scheme for kerosene–air premixed flames," Combustion and Flame, vol. 157, no. 7, pp. 1364–1373, 2010. pages 23, 24, 155, 167, 195
- [131] B. Franzelli, Chemical kinetics in Large Eddy Simulation of turbulent combustion. PhD thesis, PhD thesis, Université de Toulouse-Ecole doctorale MEGeP, CERFACS-CFD Team, 2010. pages 23, 24, 43
- [132] T. Turányi, "Sensitivity analysis of complex kinetic systems. tools and applications," Journal of mathematical chemistry, vol. 5, no. 3, pp. 203–248, 1990. pages 25
- [133] J. C. Sutherland and A. Parente, "Combustion modeling using principal component analysis," Proceedings of the Combustion Institute, vol. 32, no. 1, pp. 1563–1570, 2009. pages 25
- [134] T. Lu and C. K. Law, "A directed relation graph method for mechanism reduction," Proceedings of the Combustion Institute, vol. 30, no. 1, pp. 1333–1341, 2005. pages 25

-
- [135] P. Pepiot-Desjardins and H. Pitsch, “An efficient error-propagation-based reduction method for large chemical kinetic mechanisms,” Combustion and Flame, vol. 154, no. 1-2, pp. 67–81, 2008. pages 25
- [136] “San diego mechanism website.” <https://web.eng.ucsd.edu/mae/groups/combustion/mechanism.html>. pages 25, 160
- [137] S. Lam, “Singular perturbation for stiff equations using numerical methods,” in Recent advances in the aerospace sciences, pp. 3–19, Springer, 1985. pages 25
- [138] S. Lam and D. Coussis, “Conventional asymptotics and computational singular perturbation for simplified kinetics modelling,” in Reduced kinetic mechanisms and asymptotic approximations for methane-air flames, pp. 227–242, Springer, 1991. pages
- [139] S. Lam, “Using CSP to understand complex chemical kinetics,” Combustion Science and Technology, vol. 89, no. 5-6, pp. 375–404, 1993. pages 25
- [140] T. Lu, C. K. Law, C. S. Yoo, and J. H. Chen, “Dynamic stiffness removal for direct numerical simulations,” Combustion and Flame, vol. 156, no. 8, pp. 1542–1551, 2009. pages 25
- [141] T. Løvås, P. Amnéus, F. Mauss, and E. Mastorakos, “Comparison of automatic reduction procedures for ignition chemistry,” Proceedings of the Combustion Institute, vol. 29, no. 1, pp. 1387–1393, 2002. pages 25
- [142] “Gri-3.0 website.” [GregoryP.Smith,DavidM.Golden,MichaelFrenklach,NigelW.Moriarty,BorisEiteneer,MikhailGoldenberg,C.ThomasBowman,RonaldK.Hanson,SoonhoSong,WilliamC.Gardiner,Jr.,VitaliV.Lissianski,andZhiweiQinhttp://combustion.berkeley.edu/gri-mech/](http://combustion.berkeley.edu/gri-mech/). pages 26
- [143] Q. Cazères, P. Pepiot, E. Riber, and B. Cuenot, “A fully automatic procedure for the analytical reduction of chemical kinetics mechanisms for computational fluid dynamics applications,” Fuel, vol. 303, p. 121247, 2021. pages 26
- [144] B. Lewis and G. Von Elbe, “Combustion, explosions and flame in gases,” New York, 1987. pages 26
- [145] F. A. Williams, Combustion theory. CRC Press, 2018. pages 27, 42
- [146] J. O. Hirschfelder, C. F. Curtiss, R. B. Bird, and M. G. Mayer, Molecular theory of gases and liquids, vol. 165. Wiley New York, 1964. pages 28
- [147] C. Wilke, “A viscosity equation for gas mixtures,” The journal of chemical physics, vol. 18, no. 4, pp. 517–519, 1950. pages 30
- [148] S. Mathur, P. Tondon, and S. Saxena, “Thermal conductivity of binary, ternary and quaternary mixtures of rare gases,” Molecular physics, vol. 12, no. 6, pp. 569–579, 1967. pages 31
- [149] D. Stull and H. Prophet, “JANAF Thermochemical Tables,” Nat. Stand. Ref. Data Ser., Nat. Bur. Stds.(US), IL, vol. 1141, 1971. pages 32
- [150] S. B. Pope, Turbulent flows. Cambridge university press, 2000. pages 36, 37, 103, 104, 105, 106, 108, 112, 113, 142, 267
- [151] L. F. Richardson, Weather prediction by numerical process. University Press, 1922. pages 36, 37

- [152] A. N. Kolmogorov, "The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers," Cr Acad. Sci. URSS, vol. 30, pp. 301–305, 1941. pages 36
- [153] S. Ghosal, "Analysis and control of errors in the numerical simulation of turbulence," in Turbulent flow computation, pp. 101–140, Springer, 2002. pages 39
- [154] V. Moureau, G. Lartigue, Y. Sommerer, C. Angelberger, O. Colin, and T. Poinso, "Numerical methods for unsteady compressible multi-component reacting flows on fixed and moving grids," Journal of Computational Physics, vol. 202, no. 2, pp. 710–736, 2005. pages 39
- [155] J. Smagorinsky, "General circulation experiments with the primitive equations: I. The basic experiment," Monthly weather review, vol. 91, no. 3, pp. 99–164, 1963. pages 40
- [156] F. Nicoud and F. Ducros, "Subgrid-scale stress modelling based on the square of the velocity gradient tensor," Flow, Turbulence and Combustion, vol. 62, no. 3, pp. 183–200, 1999. pages 40
- [157] A. Vreman, "An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications," Physics of Fluids, vol. 16, no. 10, pp. 3670–3681, 2004. pages 40, 196, 198, 209, 211, 265
- [158] F. Nicoud, H. B. Toda, O. Cabrit, S. Bose, and J. Lee, "Using singular values to build a subgrid-scale model for large eddy simulations," Physics of Fluids, vol. 23, no. 8, p. 085106, 2011. pages 40, 211
- [159] J.-B. Chapelier and G. Lodato, "A spectral-element dynamic model for the large-eddy simulation of turbulent flows," Journal of Computational Physics, vol. 321, pp. 279–302, 2016. pages 40, 41, 54, 216
- [160] A. R. Kerstein, W. T. Ashurst, and F. A. Williams, "Field equation for interface propagation in an unsteady homogeneous flow field," Physical Review A, vol. 37, no. 7, p. 2728, 1988. pages 41
- [161] M. Boger, D. Veynante, H. Boughanem, and A. Trouvé, "Direct numerical simulation analysis of flame surface density concept for large eddy simulation of turbulent premixed combustion," in Symposium (International) on Combustion, vol. 27, pp. 917–925, Elsevier, 1998. pages 41
- [162] V. Robin, A. Mura, M. Champion, and P. Plion, "A multi-Dirac presumed PDF model for turbulent reactive flows with variable equivalence ratio," Combustion science and technology, vol. 178, no. 10-11, pp. 1843–1870, 2006. pages 41
- [163] O. Colin, F. Ducros, D. Veynante, and T. Poinso, "A thickened flame model for large eddy simulations of turbulent premixed combustion," Physics of Fluids, vol. 12, no. 7, pp. 1843–1863, 2000. pages 41, 44, 45, 46, 261
- [164] F. F. Grinstein, L. G. Margolin, and W. J. Rider, Implicit large eddy simulation, vol. 10. Cambridge university press Cambridge, 2007. pages 41
- [165] S. Hickel and N. Adams, "On implicit subgrid-scale modeling in wall-bounded flows," Physics of Fluids, vol. 19, no. 10, p. 105106, 2007. pages 41
- [166] A. Jameson and G. Lodato, "A note on the numerical dissipation from high-order discontinuous finite element schemes," Computers & Fluids, vol. 98, pp. 186–195, 2014. pages 41, 54
- [167] A. H. Mohammad, Z. Wang, and C. Liang, "Large eddy simulation of flow over a cylinder using high-order Spectral Difference method," Adv. Appl. Math. Mech, vol. 2, no. 4, pp. 451–466, 2010. pages 41

-
- [168] P. Castonguay, C. Liang, and A. Jameson, "Simulation of transitional flow over airfoils using the spectral difference method," in 40th fluid dynamics conference and exhibit, p. 4626, 2010. pages
- [169] C. Liang, S. Premasuthan, A. Jameson, and Z. Wang, "Large eddy simulation of compressible turbulent channel flow with spectral difference method," in 47th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition, p. 402, 2009. pages 41
- [170] J.-B. Chapelier, G. Lodato, and A. Jameson, "A study on the numerical dissipation of the spectral difference method for freely decaying and wall-bounded turbulence," Computers & Fluids, vol. 139, pp. 261–280, 2016. pages 41
- [171] G. J. Gassner and A. D. Beck, "On the accuracy of high-order discretizations for underresolved turbulence simulations," Theoretical and Computational Fluid Dynamics, vol. 27, no. 3-4, pp. 221–237, 2013. pages 41
- [172] S. B. Pope, "Ten questions concerning the large-eddy simulation of turbulent flows," New journal of Physics, vol. 6, no. 1, p. 35, 2004. pages 41
- [173] T. Butler and P. O'rourke, "A numerical method for two dimensional unsteady reacting flows," in Symposium (international) on combustion, vol. 16, pp. 1503–1515, Elsevier, 1977. pages 42
- [174] J.-P. Legier, T. Poinso, and D. Veynante, "Dynamically thickened flame LES model for premixed and non-premixed turbulent combustion," in Proceedings of the summer program, vol. 12, pp. 157–168, Citeseer, 2000. pages 43, 211
- [175] T. Jaravel, Prediction of pollutants in gas turbines using Large Eddy Simulation. PhD thesis, Université de Toulouse, 2016. pages 43, 44, 184, 185, 225, 226
- [176] B. Rochette, E. Riber, B. Cuenot, and O. Vermorel, "A generic and self-adapting method for flame detection and thickening in the thickened flame model," Combustion and Flame, vol. 212, pp. 448–458, 2020. pages 44, 211
- [177] C. Angelberger, D. Veynante, F. Egolfopoulos, and T. Poinso, "Large eddy simulations of combustion instabilities in premixed flames," in Proc. of the Summer Program, pp. 61–82, Citeseer, 1998. pages 44, 45, 261
- [178] F. Charlette, C. Meneveau, and D. Veynante, "A power-law flame wrinkling model for LES of premixed turbulent combustion Part I: non-dynamic formulation and initial tests," Combustion and Flame, vol. 131, no. 1-2, pp. 159–180, 2002. pages 45, 46, 47, 211
- [179] O. Colin and M. Rudgyard, "Development of high-order Taylor-Galerkin schemes for LES," Journal of Computational Physics, vol. 162, no. 2, pp. 338–371, 2000. pages 45, 122, 155, 171, 211
- [180] F. Charlette, C. Meneveau, and D. Veynante, "A power-law flame wrinkling model for LES of premixed turbulent combustion Part II: dynamic formulation," Combustion and Flame, vol. 131, no. 1-2, pp. 181–197, 2002. pages 45, 46, 47
- [181] G. Wang, M. Boileau, and D. Veynante, "Implementation of a dynamic thickened flame model for large eddy simulations of turbulent premixed combustion," Combustion and Flame, vol. 158, no. 11, pp. 2199–2213, 2011. pages 46, 47, 211

- [182] R. Mercier, T. Schmitt, D. Veynante, and B. Fiorina, “The influence of combustion SGS sub-models on the resolved flame propagation. Application to the LES of the Cambridge stratified flames,” Proceedings of the Combustion Institute, vol. 35, no. 2, pp. 1259–1267, 2015. pages 46, 48, 193, 195, 199
- [183] F. C. Gouldin, “An application of fractals to modeling premixed turbulent flames,” Combustion and flame, vol. 68, no. 3, pp. 249–266, 1987. pages 47
- [184] F. Gouldin, S. Hilton, and T. Lamb, “Experimental evaluation of the fractal geometry of flamelets,” in Symposium (International) on Combustion, vol. 22, pp. 541–550, Elsevier, 1989. pages 47
- [185] V. Moureau, P. Domingo, and L. Vervisch, “From large-eddy simulation to direct numerical simulation of a lean premixed swirl flame: Filtered laminar flame-pdf modeling,” Combustion and Flame, vol. 158, no. 7, pp. 1340–1357, 2011. pages 47
- [186] D. Veynante and V. Moureau, “Analysis of dynamic models for large eddy simulations of turbulent premixed combustion,” Combustion and Flame, vol. 162, no. 12, pp. 4622–4642, 2015. pages 47
- [187] P. Volpiani, T. Schmitt, and D. Veynante, “A posteriori tests of a dynamic thickened flame model for large eddy simulations of turbulent premixed combustion,” Combustion and Flame, vol. 174, pp. 166–178, 2016. pages
- [188] S. Mouriaux, O. Colin, and D. Veynante, “Adaptation of a dynamic wrinkling model to an engine configuration,” Proceedings of the Combustion Institute, vol. 36, no. 3, pp. 3415–3422, 2017. pages 47
- [189] M. Abramowitz and I. A. Stegun, “Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.,” National Bureau of Standards Applied Mathematics Series 55. Tenth Printing., 1972. pages 54, 55, 235
- [190] “Legendre polynomial.” https://en.wikipedia.org/wiki/Legendre_polynomials. pages
- [191] A. N. Lowan, N. Davids, and A. Levenson, “Table of the zeros of the Legendre polynomials of order 1-16 and the weight coefficients for Gauss’ mechanical quadrature formula,” Bulletin of the American Mathematical Society, vol. 48, no. 10, pp. 739–743, 1942. pages 54, 55
- [192] E. F. Toro, Riemann solvers and numerical methods for fluid dynamics: a practical introduction. Springer Science & Business Media, 2013. pages 56, 71, 72, 73
- [193] D. A. Kopriva, “A staggered-grid multidomain spectral method for the compressible Navier–Stokes equations,” Journal of Computational Physics, vol. 143, no. 1, pp. 125–158, 1998. pages 59
- [194] H. T. Huynh, “A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion,” in 47th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition, p. 403, 2009. pages 59, 60, 61, 62, 64, 78
- [195] F. Bassi and S. Rebay, “A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations,” Journal of Computational Physics, vol. 131, no. 2, pp. 267–279, 1997. pages 62
- [196] F. Bassi and S. Rebay, “A high order discontinuous Galerkin method for compressible turbulent flows,” in Discontinuous Galerkin Methods, pp. 77–88, Springer, 2000. pages 62

-
- [197] M. Vinokur, “Conservation equations of gas dynamics in curvilinear coordinate systems,” Journal of Computational Physics, vol. 14, no. 2, pp. 105–125, 1974. pages 67
- [198] H. Viviand, “Conservation forms of gas dynamic equations,” in In its Aerospace Research, Bi-monthly Bull. No. 1974-1 (ESRO-TT-144) p 153-159 (SEE N75-29988 20-99) Transl. into ENGLISH from La Rech. Aerospatiale, Bull. Bimestriel (Paris), no. 1974-1, Jan.-Feb. 1974 p 65-66, vol. 1974, 1975. pages 67
- [199] P. D. Lax, Hyperbolic systems of conservation laws and the mathematical theory of shock waves. SIAM, 1973. pages 71, 170
- [200] S. K. Godunov, “A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics,” Matematicheskii Sbornik, vol. 89, no. 3, pp. 271–306, 1959. pages 72
- [201] P. Batten, N. Clarke, C. Lambert, and D. M. Causon, “On the choice of wavespeeds for the HLLC Riemann solver,” SIAM Journal on Scientific Computing, vol. 18, no. 6, pp. 1553–1570, 1997. pages 72, 75, 84, 196, 206
- [202] P. L. Roe, “Approximate Riemann solvers, parameter vectors, and difference schemes,” Journal of Computational Physics, vol. 43, no. 2, pp. 357–372, 1981. pages 74, 75
- [203] F. Alauzet and L. Frazza, “Feature-based and goal-oriented anisotropic mesh adaptation for RANS applications in aeronautics and aerospace,” Journal of Computational Physics, vol. 439, p. 110340, 2021. pages 75
- [204] L. Lecomte, R. Vicquelin, S. Kudriakov, E. Studer, and C. Tenaud, “High-order numerical scheme for compressible multi-component real gas flows using an extension of the Roe approximate Riemann solver and specific Monotonicity-Preserving constraints,” Journal of Computational Physics, vol. 450, p. 110821, 2022. pages 75
- [205] E. F. Toro, M. Spruce, and W. Speares, “Restoration of the contact surface in the HLL-Riemann solver,” Shock waves, vol. 4, no. 1, pp. 25–34, 1994. pages 75
- [206] P. Castonguay, D. M. Williams, P. E. Vincent, and A. Jameson, “Energy stable flux reconstruction schemes for advection-diffusion problems,” Computer Methods in Applied Mechanics and Engineering, vol. 267, pp. 400–417, 2013. pages 77
- [207] A. Sheshadri and A. Jameson, “An analysis of stability of the flux reconstruction formulation on quadrilateral elements for the linear advection-diffusion equation,” Journal of Scientific Computing, vol. 74, no. 3, pp. 1757–1785, 2018. pages 77
- [208] F. Naddei, Simulation adaptative des grandes échelles d’écoulements turbulents fondée sur une méthode Galerkin discontinue. PhD thesis, Université Paris Saclay (COMUE), 2019. pages 79
- [209] E. Ching, Y. Lv, and M. Ihme, “Entropy residual as a feature-based adaptation indicator for simulations of unsteady flow,” in 54th AIAA Aerospace Sciences Meeting, p. 0837, 2016. pages 79
- [210] G. Karypis, K. Schloegel, and V. Kumar, “Parmetis: Parallel graph partitioning and sparse matrix ordering library,” University of Minnesota, Department of Computer Science, 1997. pages 79
- [211] Y. Maday, C. Mavriplis, and A. Patera, “Nonconforming mortar element methods: Application to spectral discretizations,” tech. rep., National Aeronautics and Space Administration, 1988. pages 80, 84

- [212] F. Chipman, “A-stable Runge-Kutta processes,” BIT Numerical Mathematics, vol. 11, no. 4, pp. 384–388, 1971. pages 81
- [213] “L-stability.” <https://en.wikipedia.org/wiki/L-stability>. pages 81
- [214] J. C. Butcher, Numerical methods for ordinary differential equations. John Wiley & Sons, 2016. pages 82
- [215] A. Najafi-Yazdi and L. Mongeau, “A low-dispersion and low-dissipation implicit Runge-Kutta scheme,” Journal of Computational Physics, vol. 233, pp. 315–323, 2013. pages 82
- [216] “List of Runge-Kutta methods.” https://en.wikipedia.org/wiki/List_of_Runge%E2%80%93Kutta_methods. pages 82
- [217] C. Bogey and C. Bailly, “A family of low dispersive and low dissipative explicit schemes for flow and noise computations,” Journal of Computational Physics, vol. 194, no. 1, pp. 194–214, 2004. pages 82
- [218] “Optimised Runge-Kutta time integration for the Spectral Difference method.” https://cerfacs.fr/wp-content/uploads/2018/12/CFD_AIAA2017_VANHAREN.pdf. pages 82
- [219] D. Stanescu and W. Habashi, “2N-storage low dissipation and dispersion Runge-Kutta schemes for computational acoustics,” Journal of Computational Physics, vol. 143, no. 2, pp. 674–681, 1998. pages 82
- [220] M. H. Carpenter and C. A. Kennedy, “Fourth-order 2N-storage Runge-Kutta schemes,” tech. rep., National Aeronautics and Space Administration, 1994. pages 82
- [221] J. Williamson, “Low-storage Runge-Kutta schemes,” Journal of Computational Physics, vol. 35, no. 1, pp. 48–56, 1980. pages 82
- [222] S. Gottlieb and C.-W. Shu, “Total variation diminishing Runge-Kutta schemes,” Mathematics of Computation, vol. 67, no. 221, pp. 73–85, 1998. pages 83, 155, 172, 176, 245
- [223] L. Ferracina and M. Spijker, “An extension and analysis of the Shu-Osher representation of Runge-Kutta methods,” Mathematics of Computation, vol. 74, no. 249, pp. 201–219, 2005. pages 83
- [224] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu, “High order strong stability preserving time discretizations,” Journal of Scientific Computing, vol. 38, no. 3, pp. 251–289, 2009. pages 83
- [225] “Python module for Runge-Kutta methods.” https://nodepy.readthedocs.io/en/latest/modules/runge_kutta_method.html. pages 83
- [226] R. J. Spiteri and S. J. Ruuth, “A new class of optimal high-order strong-stability-preserving time discretization methods,” SIAM Journal on Numerical Analysis, vol. 40, no. 2, pp. 469–491, 2002. pages 83, 196, 206, 245
- [227] K. S. Jung, S. O. Kim, T. Lu, J. H. Chen, and C. S. Yoo, “On the flame stabilization of turbulent lifted hydrogen jet flames in heated coflows near the autoignition limit: A comparative DNS study,” Combustion and Flame, vol. 233, p. 111584, 2021. pages 88
- [228] C. Pérez Arroyo, J. Dombard, F. Duchaine, L. Gicquel, N. Odier, G. Exilard, S. Richard, N. Buffaz, and J. Démolis, “Large-eddy simulation of an integrated high-pressure compressor and combustion chamber of a typical turbine engine architecture,” in Turbo Expo: Power for Land, Sea, and Air, vol. 84089, p. V02CT35A058, American Society of Mechanical Engineers, 2020. pages 88

-
- [229] I. Touloupoulos and J. A. Ekaterinaris, “Artificial boundary conditions for the numerical solution of the Euler equations by the discontinuous galerkin method,” Journal of Computational Physics, vol. 230, no. 15, pp. 5974–5995, 2011. pages 88
- [230] K. W. Thompson, “Time dependent boundary conditions for hyperbolic systems,” Journal of Computational Physics, vol. 68, no. 1, pp. 1–24, 1987. pages 88
- [231] T. J. Poinso and S. Lele, “Boundary conditions for direct simulations of compressible viscous flows,” Journal of Computational Physics, vol. 101, no. 1, pp. 104–129, 1992. pages 88, 94, 96
- [232] M. Baum, T. Poinso, and D. Thévenin, “Accurate boundary conditions for multicomponent reactive flows,” Journal of Computational Physics, vol. 116, no. 2, pp. 247–261, 1995. pages 88
- [233] J. C. Sutherland and C. A. Kennedy, “Improved boundary conditions for viscous, reacting, compressible flows,” Journal of Computational Physics, vol. 191, no. 2, pp. 502–524, 2003. pages 88, 96
- [234] J. W. Kim and D. J. Lee, “Generalized characteristic boundary conditions for computational aeroacoustics,” AIAA Journal, vol. 38, no. 11, pp. 2040–2049, 2000. pages 88, 89, 91, 100, 231
- [235] J. W. Kim and D. J. Lee, “Generalized characteristic boundary conditions for computational aeroacoustics, part 2,” AIAA Journal, vol. 42, no. 1, pp. 47–55, 2004. pages 88, 89, 91, 100, 231
- [236] A. Fosso P, H. Deniau, N. Lamarque, and T. Poinso, “Comparison of outflow boundary conditions for subsonic aeroacoustic simulations,” International Journal for Numerical Methods in Fluids, vol. 68, no. 10, pp. 1207–1233, 2012. pages 94
- [237] C. S. Yoo, Y. Wang, A. Trouvé, and H. G. Im, “Characteristic boundary conditions for direct simulations of turbulent counterflow flames,” Combustion Theory and Modelling, vol. 9, no. 4, pp. 617–646, 2005. pages 94
- [238] C. S. Yoo and H. G. Im, “Characteristic boundary conditions for simulations of compressible reacting flows with multi-dimensional, viscous and reaction effects,” Combustion Theory and Modelling, vol. 11, no. 2, pp. 259–286, 2007. pages 94
- [239] G. Mengaldo, D. De Grazia, F. Witherden, A. Farrington, P. Vincent, S. Sherwin, and J. Peiro, “A guide to the implementation of boundary conditions in compact high-order methods for compressible aerodynamics,” in 7th AIAA Theoretical Fluid Mechanics Conference, p. 2923, 2014. pages 98, 100
- [240] J. Kim, P. Moin, and R. Moser, “Turbulence statistics in fully developed channel flow at low Reynolds number,” Journal of Fluid Mechanics, vol. 177, pp. 133–166, 1987. pages 103
- [241] J. C. Del Alamo and J. Jiménez, “Spectra of the very large anisotropic scales in turbulent channels,” Physics of Fluids, vol. 15, no. 6, pp. L41–L44, 2003. pages 129, 130, 131, 132, 133, 134, 135, 136, 262
- [242] S. Hoyas and J. Jiménez, “Scaling of the velocity fluctuations in turbulent channels up to $Re_\tau=2003$,” Physics of Fluids, vol. 18, no. 1, p. 011702, 2006. pages
- [243] J. Jimenez and S. Hoyas, “Turbulent fluctuations above the buffer layer of wall-bounded flows,” Journal of Fluid Mechanics, vol. 611, pp. 215–236, 2008. pages 135
- [244] J. Jiménez, S. Hoyas, M. P. Simens, and Y. Mizuno, “Turbulent boundary layers and channels at moderate Reynolds numbers,” Journal of Fluid Mechanics, vol. 657, pp. 335–360, 2010. pages 103

- [245] C. B. Millikan, “A critical discussion of turbulent flow in channels and circular tubes,” in Proc. 5th Int. Congress on Applied Mechanics (Cambridge, MA, 1938), pp. 386–392, Wiley, 1939. pages 106
- [246] J. G. Eggels, F. Unger, M. Weiss, J. Westerweel, R. J. Adrian, R. Friedrich, and F. T. Nieuwstadt, “Fully developed turbulent pipe flow: a comparison between direct numerical simulation and experiment,” Journal of Fluid Mechanics, vol. 268, pp. 175–210, 1994. pages 107
- [247] G. K. El Khoury, P. Schlatter, A. Noorani, P. F. Fischer, G. Brethouwer, and A. V. Johansson, “Direct numerical simulation of turbulent pipe flow at moderately high Reynolds numbers,” Flow, Turbulence and Combustion, vol. 91, no. 3, pp. 475–495, 2013. pages 109, 137, 138, 139, 140, 141, 142, 143, 262, 263
- [248] D. Feldmann, C. Bauer, and C. Wagner, “Computational domain length and Reynolds number effects on large-scale coherent motions in turbulent pipe flow,” Journal of Turbulence, vol. 19, no. 3, pp. 274–295, 2018. pages 107, 109
- [249] C. Benocci and A. Pinelli, “The role of the forcing term in the large eddy simulation of equilibrium channel flow,” Engineering Turbulence Modeling and Experiments, 1990. pages 109
- [250] G. Barenblatt, “Scaling laws for fully developed turbulent shear flows. Part 1. Basic hypotheses and analysis,” Journal of Fluid Mechanics, vol. 248, pp. 513–520, 1993. pages 110
- [251] G. Barenblatt and V. Prostokishin, “Scaling laws for fully developed turbulent shear flows. Part 2. Processing of experimental data,” Journal of Fluid Mechanics, vol. 248, pp. 521–529, 1993. pages
- [252] G. Barenblatt, A. Chorin, and V. Prostokishin, “Scaling laws for fully developed turbulent flow in pipes,” American Society of Mechanical Engineers, vol. 50, no. 7, pp. 414–429, 1997. pages
- [253] M. Zagarola and A. Smits, “Scaling of the mean velocity profile for turbulent pipe flow,” Physical review letters, vol. 78, no. 2, p. 239, 1997. pages
- [254] N. Afzal, A. Seenaa, and A. Bushra, “Power law velocity profile in fully developed turbulent pipe and channel flows,” Journal of Hydraulic Engineering, vol. 133, no. 9, pp. 1080–1086, 2007. pages 110
- [255] D. Cooper, D. Jackson, B. E. Launder, and G. Liao, “Impinging jet studies for turbulence model assessment—I. Flow-field experiments,” International Journal of Heat and Mass Transfer, vol. 36, no. 10, pp. 2675–2684, 1993. pages 111
- [256] F. M. White and J. Majdalani, Viscous fluid flow, vol. 3. McGraw-Hill New York, 2006. pages 111
- [257] T. Passot and A. Pouquet, “Numerical simulation of compressible homogeneous flows in the turbulent regime,” Journal of Fluid Mechanics, vol. 181, pp. 441–466, 1987. pages 113
- [258] Y.-H. Pao, “Structure of turbulent velocity and scalar fields at large wavenumbers,” The Physics of Fluids, vol. 8, no. 6, pp. 1063–1075, 1965. pages 113
- [259] “Gamma function.” https://en.wikipedia.org/wiki/Gamma_function. pages 114, 257
- [260] G. R. Tabor and M. Baba-Ahmadi, “Inlet conditions for large eddy simulation: A review,” Computers & Fluids, vol. 39, no. 4, pp. 553–567, 2010. pages 115

-
- [261] X. Wu, “Inflow turbulence generation methods,” Annu. Rev. Fluid Mech, vol. 49, no. 1, pp. 23–49, 2017. pages 115, 117
- [262] N. S. Dhamankar, G. A. Blaisdell, and A. S. Lyrintzis, “Overview of turbulent inflow boundary conditions for large-eddy simulations,” AIAA Journal, vol. 56, no. 4, pp. 1317–1334, 2018. pages 115
- [263] R. H. Kraichnan, “Diffusion by a random velocity field,” The Physics of Fluids, vol. 13, no. 1, pp. 22–31, 1970. pages 115, 116, 122, 144
- [264] W. Bechara, C. Bailly, P. Lafon, and S. M. Candel, “Stochastic approach to noise modeling for free turbulent flows,” AIAA Journal, vol. 32, no. 3, pp. 455–463, 1994. pages 115, 116, 144
- [265] M. Klein, A. Sadiki, and J. Janicka, “A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations,” Journal of Computational Physics, vol. 186, no. 2, pp. 652–665, 2003. pages 115
- [266] N. Jarrin, S. Benhamadouche, D. Laurence, and R. Prosser, “A synthetic-eddy-method for generating inflow conditions for large-eddy simulations,” International Journal of Heat and Fluid Flow, vol. 27, no. 4, pp. 585–593, 2006. pages 115
- [267] A. Spille-Kohoff and H.-J. Kaltenbach, “Generation of turbulent inflow data with a prescribed shear-stress profile,” Proceedings of the Third AFOSR International Conference on DNS/LES, 2001. pages 116
- [268] A. Keating, U. Piomelli, E. Balaras, and H.-J. Kaltenbach, “A priori and a posteriori tests of inflow conditions for large-eddy simulation,” Physics of Fluids, vol. 16, no. 12, pp. 4696–4712, 2004. pages 116
- [269] M. Karweit, P. Blanc-Benon, D. Juvé, and G. Comte-Bellot, “Simulation of the propagation of an acoustic wave through a turbulent velocity field: A study of phase variance,” The Journal of the Acoustical Society of America, vol. 89, no. 1, pp. 52–62, 1991. pages 116, 117, 144
- [270] “Subroutine to generate random numbers.” <http://www.math.sci.hiroshima-u.ac.jp/m-mat/MT/VERSIONS/FORTRAN/mt95.f90>. pages 117
- [271] “Chapter Two GENERAL PRINCIPLES IN RANDOM VARIATE GENERATION.” http://luc.devroye.org/chapter_two.pdf. pages 117
- [272] T. Saad, D. Cline, R. Stoll, and J. C. Sutherland, “Scalable tools for generating synthetic isotropic turbulence with arbitrary spectra,” AIAA Journal, vol. 55, no. 1, pp. 327–331, 2017. pages 117, 118, 135
- [273] A. Richards, T. Saad, and J. C. Sutherland, “A fast turbulence generator using graphics Processing Units,” in 2018 Fluid Dynamics Conference, p. 3559, 2018. pages 117
- [274] C. Bailly and D. Juve, “A stochastic approach to compute subsonic noise using linearized Euler’s equations,” in 5th AIAA/CEAS aeroacoustics conference and exhibit, p. 1872, 1999. pages 118
- [275] N. Guezennec and T. Poinso, “Acoustically nonreflecting and reflecting boundary conditions for vorticity injection in compressible solvers,” AIAA Journal, vol. 47, no. 7, pp. 1709–1722, 2009. pages 118
- [276] G. Daviller, G. Oztarlik, and T. Poinso, “A generalized non-reflecting inlet boundary condition for steady and forced compressible flows with injection of vortical and acoustic waves,” Computers & Fluids, vol. 190, pp. 503–513, 2019. pages 118, 123, 135

- [277] T. S. Lund, X. Wu, and K. D. Squires, “Generation of turbulent inflow data for spatially-developing boundary layer simulations,” Journal of Computational Physics, vol. 140, no. 2, pp. 233–258, 1998. pages 119
- [278] A. Smirnov, S. Shi, and I. Celik, “Random flow generation technique for large eddy simulations and particle-dynamics modeling,” Journal of Fluids Engineering, vol. 123, no. 2, pp. 359–371, 2001. pages 119
- [279] M. Billson, L.-E. Eriksson, L. Davidson, and P. Jordan, “Modeling of synthesized anisotropic turbulence and its sound emission,” in 10th AIAA/CEAS aeroacoustics conference, p. 2857, 2004. pages 119
- [280] M. Billson, L.-E. Eriksson, and L. Davidson, “Jet noise modeling using synthetic anisotropic turbulence,” in 10th AIAA/CEAS aeroacoustics conference, p. 3028, 2004. pages 119
- [281] M. L. Shur, P. R. Spalart, M. K. Strelets, and A. K. Travin, “Synthetic turbulence generators for RANS-LES interfaces in zonal simulations of aerodynamic and aeroacoustic problems,” Flow, Turbulence and Combustion, vol. 93, no. 1, pp. 63–92, 2014. pages 119
- [282] J. Boudet, J.-F. Monier, and F. Gao, “Implementation of a roughness element to trip transition in large-eddy simulation,” Journal of Thermal Science, vol. 24, no. 1, pp. 30–36, 2015. pages 120, 132
- [283] T. Schonfeld and M. Rudgyard, “Steady and unsteady flow simulations using the hybrid flow solver AVBP,” AIAA Journal, vol. 37, no. 11, pp. 1378–1385, 1999. pages 122, 155, 211
- [284] N. Gourdain, L. Gicquel, M. Montagnac, O. Vermorel, M. Gazaix, G. Staffelbach, M. Garcia, J. Boussuge, and T. Poinso, “High performance parallel computing of flows in complex geometries: I. methods,” Computational Science & Discovery, vol. 2, no. 1, p. 015003, 2009. pages 122, 155
- [285] N. Odier, M. Sanjosé, L. Gicquel, T. Poinso, S. Moreau, and F. Duchaine, “A characteristic inlet boundary condition for compressible, turbulent, multispecies turbomachinery flows,” Computers & Fluids, vol. 178, pp. 41–55, 2019. pages 123
- [286] R. Mercier, Turbulent combustion modeling for Large Eddy Simulation of non-adiabatic stratified flames. PhD thesis, Ecole Centrale Paris, 2015. pages 131, 139, 194, 196, 197, 265
- [287] P. Aillaud, Simulations aux grandes échelles pour le refroidissement d’aubages de turbine haute-pressure. PhD thesis, Université de Toulouse, 2017. pages 131, 139
- [288] T. M. Farabee and M. J. Casarella, “Spectral features of wall pressure fluctuations beneath turbulent boundary layers,” Physics of Fluids A: Fluid Dynamics, vol. 3, no. 10, pp. 2410–2420, 1991. pages 135
- [289] W. Trojak, A. Scillitoe, and R. Watson, “Effect of Flux Function Order and Working Precision in Spectral Element Methods,” in AIAA Scitech 2020 Forum, p. 0566, 2020. pages 155
- [290] S. Nambully, P. Domingo, V. Moureau, and L. Vervisch, “A filtered-laminar-flame PDF sub-grid scale closure for LES of premixed turbulent flames. Part I: Formalism and application to a bluff-body burner with differential diffusion,” Combustion and Flame, vol. 161, no. 7, pp. 1756–1774, 2014. pages 163, 193, 197, 199
- [291] A. Bonhomme, L. Selle, and T. Poinso, “Curvature and confinement effects for flame speed measurements in laminar spherical and cylindrical flames,” Combustion and Flame, vol. 160, no. 7, pp. 1208–1214, 2013. pages 165, 166

-
- [292] D. Bradley, P. H. Gaskell, and X.-J. Gu, "Burning velocities, Markstein lengths, and flame quenching for spherical methane-air flames: a computational study," Combustion and flame, vol. 104, no. 1-2, pp. 176–198, 1996. pages 166
- [293] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut, "Revisiting the spectral analysis for high-order spectral discontinuous methods," Journal of Computational Physics, vol. 337, pp. 379–402, 2017. pages 172
- [294] T. Poinso, D. Veynante, and S. Candel, "Quenching processes and premixed turbulent combustion diagrams," Journal of Fluid Mechanics, vol. 228, pp. 561–606, 1991. pages 190
- [295] C. Ridders, "A new algorithm for computing a single root of a real continuous function," IEEE Transactions on circuits and systems, vol. 26, no. 11, pp. 979–980, 1979. pages 192
- [296] M. S. Sweeney, S. Hochgreb, M. J. Dunn, and R. S. Barlow, "The structure of turbulent stratified and premixed methane/air flames I: Non-swirling flows," Combustion and flame, vol. 159, no. 9, pp. 2896–2911, 2012. pages 193, 199, 201, 203, 265
- [297] M. S. Sweeney, S. Hochgreb, M. J. Dunn, and R. S. Barlow, "The structure of turbulent stratified and premixed methane/air flames II: Swirling flows," Combustion and Flame, vol. 159, no. 9, pp. 2912–2929, 2012. pages
- [298] R. Zhou, S. Balusamy, M. S. Sweeney, R. S. Barlow, and S. Hochgreb, "Flow field measurements of a series of turbulent premixed and stratified methane/air flames," Combustion and Flame, vol. 160, no. 10, pp. 2017–2028, 2013. pages 193, 196, 197, 198, 200, 265
- [299] S. Nambully, P. Domingo, V. Moureau, and L. Vervisch, "A filtered-laminar-flame PDF sub-grid-scale closure for LES of premixed turbulent flames: II. Application to a stratified bluff-body burner," Combustion and flame, vol. 161, no. 7, pp. 1775–1791, 2014. pages 193
- [300] F. Proch and A. M. Kempf, "Numerical analysis of the Cambridge stratified flame series using artificial thickened flame LES with tabulated premixed flame chemistry," Combustion and Flame, vol. 161, no. 10, pp. 2627–2646, 2014. pages 193, 195, 198, 199, 200, 201, 203, 265
- [301] F. Proch, P. Domingo, L. Vervisch, and A. M. Kempf, "Flame resolved simulation of a turbulent premixed bluff-body burner experiment. Part I: Analysis of the reaction zone dynamics with tabulated chemistry," Combustion and Flame, vol. 180, pp. 321–339, 2017. pages 193, 195, 197
- [302] F. Proch, P. Domingo, L. Vervisch, and A. M. Kempf, "Flame resolved simulation of a turbulent premixed bluff-body burner experiment. Part II: A-priori and a-posteriori investigation of sub-grid scale wrinkling closures in the context of artificially thickened flame modeling," Combustion and Flame, vol. 180, pp. 340–350, 2017. pages 193, 195
- [303] P. Lindstedt, "month progree report 1," tech. rep., Technical report, Brite Euram Program Project BE 95 1523. Report No. TR-96 009, 1997. pages 195
- [304] R. Mercier, B. Fiorina, F. Proch, and A. M. Kempf, "Numerical and modeling strategies for the simulation of the Cambridge stratified flame series," in Eighth International Symposium on Turbulence and Shear Flow Phenomena, Begel House Inc., 2013. pages 195, 197
- [305] G. P. Smith, D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. T. Bowman, R. K. Hanson, S. Song, W. C. Gardiner Jr, et al., "GRImech 3.0 reaction mechanism," Sandia National Laboratory, vol. 20, no. 0, p. 0, 2000. pages 195
- [306] V. Moureau, P. Domingo, and L. Vervisch, "Design of a massively parallel CFD code for complex geometries," Comptes Rendus Mécanique, vol. 339, no. 2-3, pp. 141–148, 2011. pages 196

- [307] A. Kempf, B. J. Geurts, and J. Oefelein, “Error analysis of large-eddy simulation of the turbulent non-premixed sydney bluff-body flame,” Combustion and flame, vol. 158, no. 12, pp. 2408–2419, 2011. pages 198
- [308] F. C. Marincola, T. Ma, and A. Kempf, “Large eddy simulations of the Darmstadt turbulent stratified flame series,” Proceedings of the Combustion Institute, vol. 34, no. 1, pp. 1307–1315, 2013. pages 198
- [309] M. Philip, M. Boileau, R. Vicquelin, E. Riber, T. Schmitt, B. Cuenot, D. Durox, and S. Candel, “Large Eddy Simulations of the ignition sequence of an annular multiple-injector combustor,” Proceedings of the Combustion Institute, vol. 35, no. 3, pp. 3159–3166, 2015. pages 199
- [310] P. Gruhlke, E. Inanc, R. Mercier, B. Fiorina, and A. M. Kempf, “A simple post-processing method to correct species predictions in artificially thickened turbulent flames,” Proceedings of the Combustion Institute, vol. 38, no. 2, pp. 2977–2984, 2021. pages 199
- [311] A. Ghani, T. Poinso, L. Gicquel, and G. Staffelbach, “LES of longitudinal and transverse self-excited combustion instabilities in a bluff-body stabilized turbulent premixed flame,” Combustion and Flame, vol. 162, no. 11, pp. 4075–4083, 2015. pages 204, 205, 206, 211, 212
- [312] B. Rochette, F. Collin-Bastiani, L. Gicquel, O. Vermorel, D. Veynante, and T. Poinso, “Influence of chemical schemes, numerical method and dynamic turbulent combustion modeling on LES of premixed turbulent flames,” Combustion and Flame, vol. 191, pp. 417–430, 2018. pages 204, 205, 206, 211, 212
- [313] C. Fureby and S.-I. Moller, “Large eddy simulation of reacting flows applied to bluff body stabilized flames,” AIAA journal, vol. 33, no. 12, pp. 2339–2347, 1995. pages
- [314] N. Zettervall, K. Nordin-Bates, E. Nilsson, and C. Fureby, “Large eddy simulation of a premixed bluff body stabilized flame using global and skeletal reaction mechanisms,” Combustion and Flame, vol. 179, pp. 1–22, 2017. pages 204, 212
- [315] P. A. Cocks, M. C. Soteriou, and V. Sankaran, “Impact of numerics on the predictive capabilities of reacting flow LES,” Combustion and Flame, vol. 162, no. 9, pp. 3394–3411, 2015. pages 204, 206
- [316] C. Geuzaine and J.-F. Remacle, “Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities,” International journal for numerical methods in engineering, vol. 79, no. 11, pp. 1309–1331, 2009. pages 210
- [317] F. B. Hildebrand, Introduction to numerical analysis. Courier Corporation, 1987. pages 236
- [318] C. Hirsch, Numerical computation of internal & external flows: fundamentals of numerical discretization. John Wiley & Sons, Inc., 1988. pages 253, 254
- [319] R. Warming, R. M. Beam, and B. Hyett, “Diagonalization and simultaneous symmetrization of the gas-dynamic matrices,” Mathematics of Computation, vol. 29, no. 132, pp. 1037–1045, 1975. pages 254

