

Apprentissage non-supervisé pour la simulation d'écoulements de fluide parfait

M. DUGUEY¹,

Encadrants: J. PETER² et S. HEIB³

Référent universitaire: E. AUDUSSE⁴

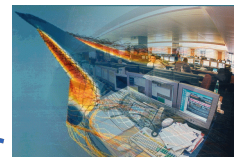
¹ *Élève ingénieur, École Sup Galilée, spécialité Mathématiques Appliquées et Calcul Scientifique, Villetaneuse 93430, FRANCE*
email: martin.duguey@gmail.com

² *Ingénieur de recherche, ONERA, Département DAAA, Unité DEFI, Châtillon 92320, FRANCE*
email: jacques.peter@onera.fr

³ *Ingénieur de recherche, ONERA, Département DAAA, Unité CLEF, Châtillon 92320, FRANCE*
email: sebastien.heib@onera.fr

⁴ *Maître de Conférences, Université Sorbonne Paris Nord, LAGA, UMR 7539, Villetaneuse 93430, FRANCE*
email: audusse@math.univ-paris13.fr

Mars - Août 2022



Abstract

The use of artificial intelligence for Computational Fluid Dynamics tends to show benefits in a context where fast results are needed and database are larger than ever. Moreover, it can be shown that AI really helps to deeply understand physical and numerical phenomenon. Due to this perspective and according to steady Euler's equation, we have designed a process which helps us to find numerical values of parameters for selected space discretizations. This process, involving machine learning approach, is based on computational data from a finite volume method.

Key words: Computational Fluid Dynamics, Space Discretization, Machine Learning, Finite Volume Method.

Résumé

L'utilisation de l'intelligence artificielle pour la dynamique des fluides numérique montre de réels bénéfices dans une période où des faibles coûts de calcul sont recherchés et où les bases de données sont de plus en plus grandes. Par ailleurs, l'intelligence artificielle permet une meilleure compréhension des phénomènes physiques et numériques liés à la simulation. Dans cette perspective et en considérant les équations d'Euler stationnaires, on a élaboré une démarche nous permettant de trouver des valeurs numériques optimales de paramètres de la discrétisation spatiale. Le processus comprenant une étape d'apprentissage automatique est basé sur des données numériques issues d'un solveur volumes finis.

Mots clés: Dynamique des fluides numérique, Discrétisation Spatiale, Apprentissage Automatique, Méthode des Volumes Finis.

Remerciements

J'aimerais remercier Jacques PETER pour ce stage et les nombreuses discussions autour du sujet. Ce fut un plaisir d'échanger quotidiennement et d'avoir pu profiter de son savoir-faire. Plus généralement, durant ces six mois et grâce à l'expertise des personnes que j'ai côtoyées, j'ai pu apprendre et véritablement compléter ma formation. Je remercie donc tous les membres de l'unité DAAA-DEFI de l'ONERA et particulièrement Alexandre SUSS avec qui j'ai partagé mon bureau et mes nombreuses questions pratiques.

Je remercie aussi Sébastien HEIB, Pascal RAUD, Christophe BENOIT et Sébastien BOURASSEAU pour leur aide précieuse durant ce stage. Les résultats obtenus n'auraient pas été envisageable sans les ressources qu'ils ont mises à ma disposition et le temps qu'ils m'ont consacré.

Je remercie aussi les personnes avec qui j'ai été en contact à distance et qui m'ont permis de mieux comprendre les outils mis à ma disposition notamment Mathieu BALESSENT, Loic BREVAULT et Christian MUSSO pour la partie apprentissage avec *OpenTURNS*. Mais aussi Lionel CASTILLON pour m'avoir fourni un script me permettant une meilleure visualisation des calculs numériques réalisés.

Sommaire

| | |
|--|----|
| Notations | 5 |
| Introduction | 6 |
| 1. L'apprentissage automatique au service de la modélisation et de la simulation d'écoulements. | 7 |
| 1.1 L'apprentissage automatique pour la paramétrisation de modèle d'écoulements, une méthode générale. | 7 |
| 1.2 L'apprentissage automatique pour la paramétrisation de modèle d'écoulements, une méthode spécifique pour les schémas Galerkin discontinus. | 10 |
| 1.3 L'apprentissage automatique pour la résolution de problèmes physiques. | 14 |
| 2. Une approche volumes finis pour la résolution d'un problème Euler compressible stationnaire. | 18 |
| 2.1 Le problème Euler compressible stationnaire. | 18 |
| 2.2 La méthode volumes finis du code <i>elsA</i> . | 22 |
| 2.3 Étude de la convergence du code <i>elsA</i> . | 25 |
| 3. Une méthodologie automatique pour l'identification de bons paramètres. | 28 |
| 3.1 Définition d'un critère d'intérêt. | 28 |
| 3.2 Discussion autour des paramètres influents du schéma Jameson-Schmidt-Turkel (JST). | 34 |
| 3.3 Modélisation et recherche optimale des paramètres. | 35 |
| 4. Résultats numériques sur des simulations 2D profil NACA0012. | 40 |
| 4.1 Espace de paramètres à deux dimensions. | 40 |
| 4.2 Espace de paramètres à trois dimensions. | 45 |
| Conclusion | 50 |
| Références | 51 |

Notations

| | |
|-----------------------------|---|
| d | dimension spatiale. |
| t | variable de temps. |
| \bar{x} | variables d'espace ($\bar{x} := (x, y, z)$ pour $d = 3$). |
| Ω | domaine spatial d'intérêt. |
| M | nombre de Mach. |
| ρ | masse volumique. |
| \bar{V} | vecteur vitesse ($\bar{V} := [u, v, w]$ pour $d = 3$). |
| $\ \bar{V}\ $ | norme vecteur vitesse $\ \bar{V}\ := \sqrt{u^2 + v^2 + w^2}$. |
| E | énergie totale. |
| e | densité massique d'énergie interne. |
| p_s | pression statique donnée par l'équation des gaz parfaits $p_s := (\gamma - 1)(\rho E - \frac{1}{2}\rho(u^2 + v^2 + w^2))$. |
| H | enthalpie totale, $H := E + \frac{p_s}{\rho}$. |
| h | densité massique d'enthalpie libre. |
| W | vecteur des variables d'état du système des équations d'Euler. |
| \mathcal{F} | vecteur de flux physique du système des équations d'Euler ($\mathcal{F} := [f, g, h]$ pour $d = 3$). |
| α | angle d'attaque. |
| Γ_b | paroi ou profil étudié sur le domaine Ω . |
| $\tilde{\Omega}$ | maillage du domaine Ω . |
| \mathcal{N} | nombre de points sur la paroi pour définir le maillage $\tilde{\Omega}$. |
| c_i | cellule quelconque de $\tilde{\Omega}$. |
| F_{c_i} | ensemble des cellules partageant une face avec la cellule c_i . |
| $\ \sigma_f\ $ | surface de la face f . |
| λ_f | rayon spectral de la matrice jacobienne associée au flux \mathcal{F} . |
| $\psi_{\frac{1}{2},c}$ | vecteur des variables d'état discrètes du système des équations d'Euler au centre cellule. |
| $\psi_{\frac{1}{2},f}$ | vecteur des variables d'état discrètes du système des équations d'Euler au centre face. |
| \mathcal{V}_{c_i} | volume de la cellule c_i . |
| $(\cdot)_{\frac{1}{2},c}$ | variable d'état au centre cellule de la cellule courante. |
| $(\cdot)_{\frac{1}{2},c}^f$ | variable d'état au centre cellule de la cellule voisine par la face f de la cellule courante. |
| ε | critère d'identification de bonnes solutions, $\varepsilon \in \mathbb{R}$. |
| D | dimension de l'espace des paramètres. |
| p | point de l'espace des paramètres. |
| f | fonction définie sur l'espace des paramètres et à valeurs dans \mathbb{R} donnant la valeur de ε associée. |
| \hat{f} | estimation de f par régression par processus gaussien. |
| $\mathcal{GP}(m, k)$ | processus gaussien de moyenne m et de fonction de covariance k . |
| M_p | nombre de réalisation de f pour le plan d'expérience initial. |
| f_i | réalisation de f , donc de ε au point i . |
| Φ | fonction de répartition de la loi gaussienne centrée réduite. |
| ϕ | fonction de densité de probabilité de la loi gaussienne centrée réduite. |

Introduction

La simulation numérique de manière générale fait intervenir de nombreux paramètres, qu'ils soient liés intrinsèquement à la discrétisation spatiale ou temporelle ou simplement des variables nécessaires à la convergence du code d'intérêt. De ce fait, plusieurs choix peuvent être laissés libres au savoir-faire de la personne développant le code numérique ou à des recommandations de la littérature issues de l'expérimentation numérique. On peut alors tout à fait imaginer que des jeux de paramètres différents conduisent à des solutions numériques significativement différentes. Dans un contexte de dynamique des fluides numérique, cette notion est d'autant plus importante qu'on peine à pleinement théoriser les problèmes physiques qu'on cherche à modéliser, et donc à garantir l'unicité d'une solution. On peut penser aux équations de Navier-Stokes ou d'Euler par exemple. Donc ce ne peut être que l'expertise et les propriétés du système des équations continues qui nous permettent d'identifier des résultats de référence en terme de simulation numérique. Dès lors, on peut considérer le problème inverse et se demander si on peut identifier, à partir de ces propriétés physiques, des valeurs de paramètres bien adaptées?

C'est de cette démarche que peut naître le recours à l'apprentissage automatique et à l'intelligence artificielle. Le raisonnement est d'ailleurs usuel dans ce cas, on dispose d'une solution de référence ou d'intérêt et on souhaite identifier les valeurs permettant de recréer cette solution par un apprentissage supervisé par exemple. On pourrait très bien comparer les solutions obtenues et classer les valeurs numériques en fonction du résultat, c'est ce qui est fait en classification et identification de cas. De cette façon plusieurs démarches sont possibles. Dans un contexte de simulation numérique pour les écoulements, des approches différentes ont été mises en oeuvre dans la littérature, et rendre compte de ces démarches et résultats obtenus constitue la première partie de ce rapport.

Le coeur du travail réalisé vise à proposer un début de réponse à la question suivante: Peut-on initier une démarche automatique pour l'identification de paramètres optimaux menant à l'obtention de bonnes solutions numériques d'écoulements de fluide parfait?

La seconde partie est consacrée à la définition du contexte d'étude. On spécifie l'utilisation des équations d'Euler sans dépendance temporelle et on présente un schéma potentiellement intéressant à la fois dans une démarche automatique mais aussi et surtout du point de vue d'une résolution numérique par méthode volumes finis. Le dernier point de cette partie est consacré au contexte numérique associé à notre démarche et à une étude de la convergence du code numérique *elsA* ([5]) à partir duquel on réalise nos simulations. Dans une troisième partie, une démarche automatique est présentée associée à nos simulations. On établit un critère non-supervisé nous permettant d'identifier de bonnes solutions. Il est aussi question, dans un second point, d'identifier des paramètres des équations discrètes qui nous semblent intéressants et de préciser leur implémentation dans le code numérique de manière à étudier les effets de leurs variations sur le critère précédemment établi. Enfin dans un dernier point, on explique concrètement, à travers une méthode d'apprentissage utilisant une régression par processus gaussien et une méthode d'optimisation efficace, comment on peut retrouver des valeurs numériques optimales de nos paramètres pour notre critère. La dernière partie est consacrée aux résultats obtenus pour des simulations réalisées autour d'un profil *NACA0012*.

1 L'apprentissage automatique au service de la modélisation et de la simulation d'écoulements.

Ces dernières années l'apprentissage automatique et l'intelligence artificielle ont contribué à aider à la compréhension et la résolution de problèmes autour de la dynamique des fluides comme en témoigne les dates de parution des articles étudiés dans cette section. Bien qu'ils ne constituent pas une liste exhaustive de ce qui a pu être fait sur les techniques automatiques dans un contexte de simulation, ils témoignent néanmoins d'une tendance actuelle. Le plus ancien date de 2016, ce qui est relativement récent et trois d'entre eux ont été écrits après 2019. Au coeur des motivations de ce stage, cette partie bibliographique est aussi l'occasion de rendre compte de l'utilisation de différentes formes d'apprentissage automatique permettant de nouvelles approches de la résolution d'équations type Euler ou Navier-Stokes. Les méthodes et résultats présentés sont issus des travaux de Volpiani *et al.* ([28]), de Parish et Duraisamy ([20]), de Singh, Medida et Duraisamy ([25]), de Discacciati, Hesthaven et Ray ([6]), et de Raissi, Perdikaris et Karniadakis ([23]).

1.1 L'apprentissage automatique pour la paramétrisation de modèles d'écoulements, une méthode générale.

On s'intéresse, dans un premier temps, aux résultats obtenus dans [28],[20] et [25]. Les articles font état d'une méthodologie permettant de corriger un modèle d'écoulement de turbulence (RANS) selon deux grandes étapes. D'abord une étape d'assimilation de données permettant un ajustement du modèle à un cas particulier pour lequel on dispose de calculs très précis. Ensuite une étape de généralisation, visant à ajuster le modèle cette fois-ci sur un cas pour lequel on ne dispose que de données non-corrigées.

1.1.1 Un modèle physique à corriger

Ces modèles RANS aux grands nombres de Reynolds, ne traduisent que partiellement le phénomène physique qu'ils tentent de décrire (mise en défaut pour la prédiction de la transition laminaire turbulent et des décollements notamment). L'article [28] présente un exemple concret de correction des équations RANS incompressibles dans un code numérique. En effet, il est question dans ce cas, en étudiant l'écoulement d'un fluide visqueux, de montrer les limites de l'hypothèse de Boussinesq en proposant une correction d'un modèle moyenné des équations de Navier Stokes (RANS), stationnaire, avec la relation de fermeture de Spalart-Allmaras. Ce modèle s'écrit pour i, j dans $\{1, 2, 3\}$ et avec les conventions de sommation usuelles, sous la forme

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{1.1 a}$$

$$\frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial 2\nu S_{ij}}{\partial x_j} + \frac{\partial \tau_{ij}}{\partial x_j} + f_i, \tag{1.1 b}$$

$$\tau_{ij} = 2\nu_t S_{ij} - \frac{2}{3}k\delta_{ij}. \tag{1.1 c}$$

où les u_i sont les composantes de la vitesse dans l'espace, p la pression statique, $S_{ij} := (u_{ij} + u_{ji})/2$, τ le tenseur de Reynolds fait intervenir la viscosité turbulente ν_t qui est déterminée par l'équation de Spalart-Allmaras, δ_{ij} est le symbole de Kroenecker et les f_i sont les composantes des forces extérieures volumiques s'appliquant dans le problème considéré. Les auteurs de [28] ajoutent un terme de correction

$\mathbf{f}^c := (f_1^c, f_2^c, f_3^c)$ au problème de telle sorte que l'on puisse chercher les composantes optimales de \mathbf{f}^c satisfaisant les équations

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (1.2 a)$$

$$\frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial 2\nu S_{ij}}{\partial x_j} + \frac{\partial 2\nu_t S_{ij}}{\partial x_j} + f_i + f_i^c. \quad (1.2 b)$$

et permettant de retrouver les données de haute fidélité. Bien que le contexte soit plus général dans les articles [20] et [25], le principe proposé est le même, on ajoute un terme de correction aux équations (en pratique du modèle de turbulence) pour pouvoir se recaler sur les données. Mais comment trouver les composantes de ce terme de correction \mathbf{f}^c ?

1.1.2 Assimilation de données

Dans la suite, les auteurs considèrent une solution de référence pour un cas d'étude. Par exemple cette solution peut être un calcul numérique jugé suffisamment précis, typiquement un calcul direct (DNS) pour les équations de Navier-Stokes. Cette référence sert d'objectif pour construire un modèle corrigé. C'est l'étape d'assimilation de données. Les articles présentent différentes façons d'identifier les termes à déterminer. Une première approche d'optimisation classique est proposée et vise à chercher \mathbf{f}^c minimisant la fonction suivante

$$\mathcal{J}_1 := \frac{1}{2} \|\mathbf{u}(\mathbf{f}^c) - \mathbf{u}_{ref}\|_{L^2}^2 + \frac{\alpha}{2} \|\mathbf{f}^c\|_{L^2}^2$$

où le second terme de l'expression vise à pénaliser la norme de \mathbf{f}^c et α est un paramètre choisi arbitrairement de manière à avoir un bon équilibre entre un terme de correction qui ne dénature pas le modèle physique initial mais qui permet de se rapprocher de la solution de référence. On note que le premier terme fait intervenir le champ moyen de vitesse puisque c'est dans les équations de moments (1.2) que la correction intervient. Dans [20], le problème d'optimisation pour déterminer \mathbf{f}^c est traité par une approche bayésienne. En considérant $\mathbf{u}(\mathbf{f}^c)$ et \mathbf{u}_{ref} comme des variables aléatoires, on pourrait imaginer que l'on ait une idée de la correction à appliquer à notre modèle et on pourrait ainsi définir une correction *a priori* \mathbf{f}_{prior}^c . Dès lors, les auteurs estiment la probabilité de reproduire \mathbf{u}_{ref} avec \mathbf{f}^c ($\pi[\mathbf{u}_{ref}|\mathbf{f}^c]$) dans le but de maximiser la probabilité $\pi[\mathbf{f}^c|\mathbf{u}_{ref}]$ par la relation de Bayes

$$\pi[\mathbf{f}^c|\mathbf{u}_{ref}] \propto \pi[\mathbf{u}_{ref}|\mathbf{f}^c] \pi[\mathbf{f}^c].$$

D'après [1], chercher \mathbf{f}^c maximisant $\pi[\mathbf{f}^c|\mathbf{u}_{ref}]$ c'est chercher \mathbf{f}^c minimisant la relation

$$\mathcal{J}_2 := \frac{1}{2} \left[(\mathbf{u}_{ref} - \mathbf{u}(\mathbf{f}^c))^T \mathcal{C}^{-1} (\mathbf{u}_{ref} - \mathbf{u}(\mathbf{f}^c)) + (\mathbf{f}^c - \mathbf{f}_{prior}^c)^T \mathcal{C}_{\mathbf{f}_{prior}^c}^{-1} (\mathbf{f}^c - \mathbf{f}_{prior}^c) \right]$$

avec \mathcal{C} la matrice de covariance de \mathbf{u}_{ref} et $\mathcal{C}_{\mathbf{f}_{prior}^c}$ la matrice de covariance de \mathbf{f}_{prior}^c .

Chacune des méthodes d'assimilation de données mène à un problème d'optimisation. Cependant les fonctions à minimiser ont été écrites à partir du problème (1.2), alors qu'en pratique c'est le problème initial sur une grille de discrétisation spatiale qui nous intéresse. Donc la résolution de ces problèmes d'optimisation fait intervenir des systèmes de très grande taille. Le vecteur \mathbf{f}^c comporte donc autant de composantes qu'il y a de points de discrétisation au problème. En particulier pour l'approche bayésienne, en notant \mathbf{f}_{MAP}^c la paramétrisation minimisant \mathcal{J}_2 , la matrice de covariance du vecteur est définie, lorsque

\mathcal{J}_2 est linéaire, comme

$$\mathcal{C}_{MAP} := \frac{\partial \mathcal{J}}{\partial \mathbf{f}_i^c \partial \mathbf{f}_j^c}(\mathbf{f}_{MAP}^c), \quad \forall i, j \in \tilde{\Omega} \quad (1.3)$$

avec $\tilde{\Omega}$ l'espace discrétisé du problème considéré. Si \mathcal{J}_2 n'est pas linéaire, une approximation de \mathcal{C}_{MAP} est donnée par $\frac{\partial \mathcal{J}}{\partial \mathbf{f}_i^c \partial \mathbf{f}_j^c}(\mathbf{f}_{MAP}^c)$. Mais si l'hypothèse gaussienne peut être faite pour \mathbf{f}_i^c , et c'est le cas pour des problèmes de grande dimension, le vecteur de correction est complètement défini par le calcul (1.3).

Finalement, les deux approches explicitées pour l'assimilation de données peuvent se résumer à la différentiation des fonctions \mathcal{J}_1 et \mathcal{J}_2 par rapport aux composantes du terme vectoriel de correction sur un espace discrétisé, (potentiellement de grande dimension) et à une méthode de descente. Une méthode adjointe est présentée dans les articles cités pour conclure cette étape. À ce stade, le problème (1.2) peut être résolu. Cependant la résolution de ce problème, comme cela est mentionné dans [28], dépend d'un contexte numérique. Par ailleurs peut-on utiliser les résultats qui ont été obtenus pour essayer de les généraliser à un cas similaire mais avec des paramètres de simulation différents?

1.1.3 Généralisation par apprentissage automatique

En fait, on aimerait pouvoir, à ce stade, appliquer une correction à tout problème similaire, non pas en ajoutant un terme de correction avec une dépendance spatiale mais plutôt avec une dépendance liée à la simulation que l'on effectue. Par exemple, pour le cas test *Periodic Hill* présenté dans [28], plusieurs solutions de référence sont supposées connues pour différentes géométries et conditions d'écoulement (la longueur de la section, la pente en entrée et sortie). On pourrait aussi essayer de tenir compte de l'information extraite grâce à l'assimilation de données pour corriger un modèle avec une géométrie ou des conditions d'écoulements différentes. On aimerait finalement construire, et c'est ce qui est présenté dans les articles cités, une fonction ϕ qui permet de faire le lien entre \mathbf{f}^c et un espace paramétré de variables communes à tous les cas que l'on souhaite corriger.

$$\phi : \eta \mapsto \mathbf{f}^c(\eta).$$

La nécessité de construire une telle fonction réside dans le fait que si l'on veut pouvoir corriger une solution pour laquelle on ne dispose pas de solution dite fine, il paraît optimiste d'utiliser le même terme de correction que pour une simulation de référence pour obtenir une solution plus précise. L'espace paramétré mentionné doit être défini en amont. En particulier, les auteurs de [28] définissent par exemple les variables suivantes

$$\begin{aligned} \eta_1 &:= \tanh\left(\frac{\|S\|}{\epsilon_1}\right), & \eta_2 &:= \tanh\left(\frac{\|\Omega\|}{\epsilon_2}\right), & \eta_3 &:= \frac{\sqrt{\frac{\partial p}{\partial x_k} \frac{\partial p}{\partial x_k}}}{\sqrt{\frac{\partial p}{\partial x_k} \frac{\partial p}{\partial x_k} + \epsilon_3}}, \\ \eta_4 &:= \frac{\frac{1}{2} \frac{\partial u_k^2}{\partial x_k}}{\left|\frac{1}{2} \frac{\partial u_k^2}{\partial x_k}\right| + \epsilon_4}, & \eta_5 &:= \frac{\left\| \frac{u_i u_j}{u_k u_k} \frac{\partial u_i}{\partial x_j} \right\|}{\left\| \frac{u_i u_j}{u_k u_k} \frac{\partial u_i}{\partial x_j} \right\| + \epsilon_5}, & \eta_6 &:= \frac{u_k \frac{\partial p}{\partial x_k}}{\left|u_k \frac{\partial p}{\partial x_k}\right| + \epsilon_6}, \\ \eta_7 &:= \frac{\nu_t}{\nu_t + \epsilon_7}, & \eta_8 &:= \frac{1}{1 + r}, \end{aligned}$$

avec S le tenseur de contraintes, Ω le tenseur de rotation. À ce stade, avoir effectué une étape d'assimilation de données nous permet d'avoir une solution de référence \mathbf{f}_{ref}^c pour pouvoir construire ϕ par apprentissage. L'article [28] fait état de plusieurs jeux de variables η pour construire une telle fonction. Dans [20],

une méthode de régression par processus gaussien est utilisée. Cette méthode, puisqu’au coeur de notre démarche pour ce stage, sera décrite succinctement par la suite (§3.3.2). Dans [25] et [28] un réseau de neurones est proposé pour construire la fonction ϕ . La partie §1.2 décrit l’utilisation d’un réseau de neurones pour la simulation numérique, donc on détaillera plus en détail ce point dans les parties §1.2.3 et §1.2.4. Finalement, en construisant une telle fonction ϕ , les articles montrent qu’on peut trouver toutes les composantes du vecteur \mathbf{f}^c pour une configuration donnée. Pour un cas test, le problème (1.2) est bien défini et peut être résolu numériquement sous réserve de disposer d’au moins une solution fine pour une configuration de simulation particulière.

1.2 L’apprentissage automatique pour la paramétrisation de modèles d’écoulements, une méthode spécifique pour les schémas Galerkin discontinus.

Pour obtenir des simulations numériques plus précises, avec l’aide de l’apprentissage automatique, on peut corriger le modèle additivement comme on vient de le voir ou directement agir sur les paramètres des équations à résoudre. Dans cette partie, il est question d’étudier un problème physique particulier et de trouver la paramétrisation optimale à l’aide d’une méthode d’apprentissage automatique. Les travaux présentés sont issus de [6].

1.2.1 Contexte

Un point important à considérer lors de la résolution numérique de problèmes espace-temps est la notion de stabilité, essentielle à la convergence de la méthode. Certaines méthodes numériques nécessitent d’être adaptées pour mener à des solutions acceptables après convergence. En particulier, le phénomène de Gibbs peut causer une instabilité, donc une perte de précision, de la solution obtenue par méthode de Galerkin au niveau des discontinuités. L’intérêt initial d’une telle méthode est justement d’obtenir une solution très précise dans les régions régulières. On présente ici une manière de compenser ce phénomène instable par l’ajout d’un terme dissipatif. A priori, pour atténuer ce phénomène instable à l’approche des zones discontinues, il est commun d’utiliser des limiteurs ou des schémas type WENO mais la nécessité d’identifier les cellules causant la perte de régularité a un impact conséquent sur le coût de calcul de la méthode. Dans une perspective non-adaptative et plus générale, il convient de s’intéresser plutôt à l’ajout d’un terme dissipatif. Cependant des modèles existent déjà pour régulariser des solutions par l’utilisation d’un terme artificiel de dissipation: un modèle utilisant la décroissance modale la plus élevée (*MDH*, voir [21]), un autre utilisant la décroissance modale en moyenne (*MDA*, voir [15]) et un dernier modèle basé sur l’erreur de l’entropie (*EV*, voir [9] et [29]). Mais la nécessité de paramétrer ces modèles en fonction du cadre applicatif complique leur généralisation. Dans les travaux présentés dans [6], les auteurs proposent la création d’un modèle par apprentissage supervisé pour remplacer ces modèles paramétrés.

1.2.2 Problème d'intérêt et discrétisation numérique

Les auteurs de [6] s'intéressent au problème suivant :

$$\partial_t \mathbf{u} + \nabla \cdot \mathbf{f} - \nabla \cdot \mathbf{g} = 0, \quad \forall (x, t) \in \Omega \times [0, T], \quad (1.4 a)$$

$$\mathbf{u}(x, t) = \mathbf{u}_a(x, t), \quad \forall (x, t) \in \partial\Omega \times [0, T], \quad (1.4 b)$$

$$\mathbf{u}(x, t = 0) = \mathbf{u}_0(x), \quad \forall x \in \Omega. \quad (1.4 c)$$

avec f le flux convectif, g le flux dissipatif tel que $\mathbf{g} = \mu \mathbf{q}$ et $\mathbf{q} = \nabla \cdot \mathbf{u}$, et \mathbf{u}_a une condition aux limites quelconque. En utilisant un schéma de Galerkin discontinu, l'étude est restreinte sur une discrétisation faisant converger l'algorithme vers une solution faible du problème (1.4). Dès lors, la démarche est la suivante, à la suite d'un premier calcul numérique, on souhaite faire intervenir un réseau de neurones pour calculer μ et stabiliser la solution numérique.

1.2.3 Réseau de neurones, définition et caractéristiques

La suite de l'article fait intervenir des problèmes unidimensionnels et un réseau de neurones, du type perceptron multi-couches, de taille 5×10 ; donc 5 couches avec 10 noeuds par couches. On souhaite donner, en entrée de ce réseau, qui a été préalablement entraîné, le champ $\mathbf{u} = (u_i)_{i \in \mathbb{N}_*^N}$ sur une grille discrète du domaine, pour qu'il puisse fournir en sortie, le coefficient de viscosité adapté à la solution. Le réseau de neurones est finalement un modèle

$$M_\Omega(\mathbf{u}) = \mathbf{y}$$

où Ω est l'ensemble des paramètres du réseau et \mathbf{y} le vecteur solution. Une schématisation du réseau est proposée (voir Figure 1). En pratique, on ne donne pas directement \mathbf{u} au modèle. Pour s'affranchir du contexte global de résolution du problème, il est nécessaire de donner, en entrée du réseau de neurones, un champ solution sans dimension

$$x_i = \frac{u_i}{\max_{i \in \mathbb{N}_*^N} (|u_i|) + \epsilon}, \quad \forall i \in \mathbb{N}_*^N$$

où les $(x_i)_{i \in \mathbb{N}_*^N}$ sont les entrées et $\epsilon > 0$. De la même façon, il faut pouvoir redimensionner les sorties pour qu'elles soient adaptées au problème :

$$y_i = \frac{\mu_i}{\mathcal{H} \|\partial_{\mathbf{u}} \mathbf{f}\|_\infty}, \quad \forall i \in \mathbb{N}_*^N$$

où les $(y_i)_{i \in \mathbb{N}_*^N}$ sont les sorties et \mathcal{H} est un coefficient d'adimensionnement défini tel que la solution numérique reste à la précision obtenue par le schéma de Galerkin discontinu sur les zones de régularité et pour qu'elle soit au moins précise à l'ordre un sinon.

Il est mentionné dans [6] que la même fonction d'activation est utilisée pour chaque neurone d'une même couche. En particulier la fonction F_1 (LReLU) pour les premières couches et la fonction F_2 (softplus) pour la couche de sortie.

$$F_1 : x \mapsto F_1(x) = \max(x, 0) - \gamma \max(-x, 0), \quad F_2 : x \mapsto F_2(x) = \ln(1 + \exp(x))$$

où γ est un coefficient positif arbitrairement fixé à 10^{-3} .

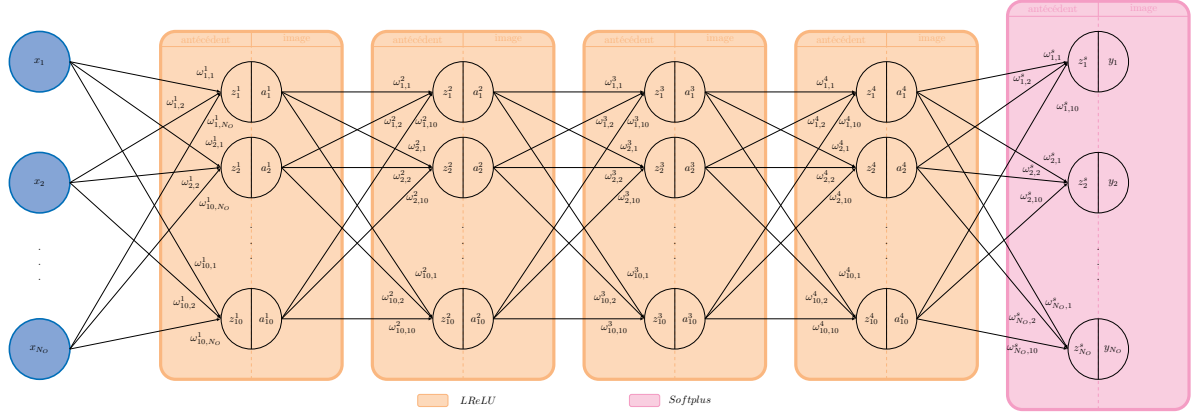


Figure 1. Schématisation du perceptron multi-couches utilisé dans [6].

1.2.4 Réseau de neurones, apprentissage

Comme la notion de réseau de neurones est récurrente dans cette première partie, on profite des travaux de [6] pour faire un point sur la manière dont est réalisé l'apprentissage de ce type de modèle. De nombreuses sources expliquent en détail cet apprentissage ([8], [18]). De part le contexte et le recours ici à l'utilisation d'un perceptron multi-couches, on s'intéresse uniquement, par le biais de [14] et [16], à une phase d'apprentissage utilisant un algorithme de rétropropagation. Initialement, on dispose de données d'entraînement $\hat{\mathbf{y}}$, typiquement des données issues des modèles ajoutant une dissipation artificielle (MDH, MDA et EV) sur des maillages fins, et donc d'autant de réalisations \mathbf{y} de notre réseau de neurones. Pour limiter au maximum le sur-apprentissage, les poids sont pondérés par un coefficient de pénalisation β arbitrairement choisi. Ils seront par la suite, déterminés par un algorithme de rétropropagation de manière à minimiser la fonction coût suivante

$$\mathcal{J} := \frac{1}{2} \left[\frac{1}{N_T} \sum_{i=1}^{N_T} \|\mathbf{y}_i - \hat{\mathbf{y}}_i\|_2^2 + \beta \|\omega_s\|_2^2 \right]$$

où ω_s est la matrice des poids entre la dernière sous-couche et la couche de sortie. En particulier, pour un réseau de taille 5×10 avec autant d'entrées que de sorties, N_O , on a $\omega_s \in \mathcal{M}_{N_O,10}(\mathbb{R})$. On note aussi que la mise à jour des poids s'effectue, non pas à chaque échantillon d'entrée fourni au réseau mais à chaque groupe d'échantillons d'entrée de taille N_T . La convergence de l'apprentissage de notre réseau est ainsi accélérée. Idéalement, N_T n'est pas exactement de la taille de notre base de données réservée à l'apprentissage mais de taille bien plus petite de manière à accélérer le temps de calcul lors de la mise à jour des poids. Cette technique est référencée en tant que *mini-batch gradient descent* dans la littérature anglophone.

L'algorithme de rétropropagation permet de mettre à jour les paramètres Ω une fois que le réseau, en phase d'apprentissage, a pu fournir des résultats à partir d'entrées. Cette mise à jour doit permettre de minimiser la fonction \mathcal{J} précédemment explicitée. Dans la suite, on note $e, 1, 2, 3, 4, s$ les différentes couches de notre modèle et on introduit

- ω_{jk}^l : poids de la relation entre le k -ième neurone de la couche $l-1$ et le j -ième neurone de la couche l ;

- a_j^l : valeur du j -ième neurone de la couche l ;
- $z_j^l := \sum_{k=1}^{N_{l-1}} \omega_{jk}^l a_k^{l-1}$ avec N_{l-1} le nombre de neurones sur la couche $l-1$.

On considère alors $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_s\}$ où donc $\omega_1 \in \mathcal{M}_{10, N_O}(\mathbb{R})$, $\omega_s \in \mathcal{M}_{N_O, 10}(\mathbb{R})$, et $\forall i \in \{2, 3, 4\}, \omega_i \in \mathcal{M}_{10, 10}(\mathbb{R})$. Pour faciliter l'écriture vectorielle du problème de propagation, on fait le choix de voir l'erreur de propagation entre une couche $l-1$ et l comme une conséquence d'une perturbation sur la somme pondérée $z_j^l \quad \forall j \in \mathbb{N}_*^{N_l}$. On définit alors $\epsilon_j^l := \frac{\partial \mathcal{J}}{\partial z_j^l}$. Il vient donc par dérivation composée que

$$\epsilon_j^s = \frac{\partial \mathcal{J}}{\partial a_j^s} \frac{\partial a_j^s}{\partial z_j^s} = \frac{\partial \mathcal{J}}{\partial a_j^s} F_2'(z_j^s) \quad \forall j \in \mathbb{N}_*^{N_O}.$$

Et finalement, en notant \star l'opérateur de produit terme à terme, on a la forme vectorielle suivante

$$\epsilon^s = \nabla_{a^s} \mathcal{J} \star F_2'(z^s).$$

Cependant on cherche à trouver les paramètres du modèle permettant de minimiser \mathcal{J} , donc il faut pouvoir identifier $(\nabla_{\omega_s} \mathcal{J})_{jk} := \frac{\partial \mathcal{J}}{\partial \omega_{jk}^s}$. En fait, on a la relation de fermeture

$$\frac{\partial \mathcal{J}}{\partial \omega_{jk}^s} = \frac{\partial \mathcal{J}}{\partial z_j^s} \frac{\partial z_j^s}{\partial \omega_{jk}^s} = \epsilon_j^s a_k^4, \quad \forall j \in \mathbb{N}_*^{N_O}, \forall k \in \mathbb{N}_*^{10}. \quad (1.5)$$

On note qu'on est autorisé à chercher une telle matrice ω^s puisque \mathcal{J} est quadratique. La mise à jour des poids requiert donc un algorithme d'optimisation. On rappelle la mise à jour des poids par la descente de gradient

$$\omega_s = \omega_s - \eta_{LR} \nabla_{\omega_s} \mathcal{J}$$

où $\nabla_{\omega_s} \mathcal{J}$ est le gradient de la fonction coût à minimiser. En pratique, les résultats présentés font mention d'une autre méthode pour la mise à jour des poids: l'algorithme d'Adam [14]. Dans ce cas, la mise à jour des poids se fait de la façon suivante

$$\omega_s = \omega_s - \eta_{LR} \frac{\hat{M}}{\sqrt{\hat{V} + \alpha}}$$

où \hat{M} et \hat{V} sont respectivement un estimateur de la moyenne et un estimateur de la variance du gradient de la fonction coût à minimiser et α est un coefficient arbitraire positif permettant de se prémunir des problèmes éventuels de la division. La facilité d'implémentation et l'efficacité de la méthode sont des points essentiels mis en avant par les auteurs de [6] pour justifier leur choix. Dans les exemples ci-dessus, η_{LR} fait référence au pas lors de la descente de gradient. Il faut noter qu'un grand pas permet de converger rapidement mais avec une paramétrisation potentiellement peu satisfaisante du modèle neuronal, tandis qu'un pas faible permet d'avoir une paramétrisation satisfaisante tout en ayant un temps d'apprentissage plus conséquent. On note par ailleurs que ces algorithmes permettent de mettre à jour les poids reliant la dernière sous-couche à la sortie. Il faut donc propager cette erreur pour mettre à jour les autres paramètres du modèle. Mais comme les neurones sont liés, d'une couche à l'autre par les poids et les fonctions d'activation il est possible de propager l'erreur obtenue, entre la dernière sous-couche et la couche de sortie, aux autres couches. Autrement dit, on peut facilement obtenir $\nabla_{\omega_4} \mathcal{J}$ à partir de $\nabla_{\omega_s} \mathcal{J}$,

puis $\nabla_{\omega_3} \mathcal{J}$ à partir de $\nabla_{\omega_4} \mathcal{J}$ et ainsi de suite par récurrence. En effet, soit $j \in \mathbb{N}_*^{10}$ fixé, on a

$$\begin{aligned}
\epsilon_j^4 &:= \frac{\partial \mathcal{J}}{\partial z_j^4}, \\
&= \frac{\partial \mathcal{J}}{\partial a_j^4} F_1'(z_j^4), \\
&= \sum_{i=1}^{N_o} \frac{\partial \mathcal{J}}{\partial a_i^s} \frac{\partial a_i^s}{\partial a_j^4} F_1'(z_j^4), \\
&= \sum_{i=1}^{N_o} \frac{\partial \mathcal{J}}{\partial a_i^s} \omega_{ij}^s F_2'(z_i^s) F_1'(z_j^4), \\
&= \sum_{i=1}^{N_o} \omega_{ij}^s \epsilon_i^s F_1'(z_j^4).
\end{aligned}$$

Ce qui donne sous forme vectorielle et en notant \times l'opérateur de produit matriciel

$$\epsilon^4 = ((\omega^s)^T \times \epsilon^s) \star F_1'(z^4).$$

De la même façon, la relation (1.5) peut se réécrire $\frac{\partial \mathcal{J}}{\omega_{jk}^4} = \epsilon_j^4 a_k^3, \forall j, k \in \mathbb{N}_*^{10}$. En itérant le processus, on propage l'erreur à travers le réseau. On note, pour les méthodes d'optimisation explicitées ci-dessus, que $\nabla \mathcal{J}$ est le seul paramètre calculable nécessaire à la mise à jour. L'algorithme de rétropropagation est appelé autant de fois que nécessaire. Le plus souvent et c'est d'ailleurs le cas dans l'article étudié, un nombre d'itérations est préalablement défini. Ce nombre d'itérations est référencé en tant que *epochs* dans la littérature anglophone.

1.3 L'apprentissage automatique pour la résolution numérique de problèmes physiques.

Dans cette partie, nous nous éloignons de ce qui a été présenté ci-dessus pour aborder une toute autre manière d'utiliser l'automatisation des tâches. Nous nous intéressons à la méthode permettant de résoudre des problèmes physiques, et en particulier des problèmes du type Navier-Stokes, présentée dans [23] et appelée *PINN* pour *Physics Informed Neural Networks*. Le point fort de cette méthode est qu'elle n'a recours à aucun code numérique type volumes finis mais utilise plutôt les propriétés des modèles à apprentissage profond.

1.3.1 Contexte

Pour chercher une solution d'un certain modèle f dépendant du temps et de l'espace, il est de plus en plus commun d'avoir recours à un métamodèle neuronal. En dynamique des fluides par exemple, un tel raisonnement est possible. Un cadre défini par l'article est la résolution d'un problème tel que

$$\partial_t \mathbf{u} + \mathcal{N}[\mathbf{u}, \lambda] = 0 \quad \forall (x, t) \in \Omega \times [0, T]. \quad (1.6)$$

\mathcal{N} étant un opérateur non-linéaire, l'utilisation d'un réseau neuronal pour modéliser ce problème semble adaptée. Néanmoins si on cherche \mathbf{u} par l'intermédiaire d'un modèle à apprentissage profond comment



Figure 2. Schématisation du procédé en temps continu.

tient-on compte du problème précédent, traduisant le phénomène physique pour une paramétrisation donnée?

1.3.2 Recherche de solution pour des problèmes en temps continu

On sait que beaucoup de problèmes physiques peuvent être modélisés par (1.6). Il faut aussi pouvoir prendre en compte les conditions aux limites et la condition initiale pour intégrer complètement le phénomène d'étude dans notre recherche de solution. Pour ce faire l'article, propose d'estimer \mathbf{u} par un modèle issu d'un apprentissage profond de données et d'utiliser cette estimation pour créer un deuxième modèle, lui aussi issu d'un apprentissage, approchant cette fois

$$f := \partial_t \mathbf{u} + \mathcal{N}[\mathbf{u}]$$

par différenciation automatique. L'avantage réside ici dans le fait que deux modèles ayant la même paramétrisation sont construits. Donc lors de la phase d'apprentissage, optimiser les paramètres de f revient à optimiser les paramètres de \mathbf{u} et ainsi intégrer parfaitement les contraintes physiques du phénomène modélisé. Une figure récapitulative est proposée (voir Figure 2). Cette recherche de paramétrisation optimale se fait en minimisant la fonction suivante

$$MSE = \frac{1}{N_u} \sum_{i=1}^{N_u} |u(t_u^i, x_u^i) - u^i|^2 + \frac{1}{N_f} \sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2$$

N_u représente ici le nombre de données d'apprentissage, en fait $\{t_u^i, x_u^i, u^i\}_{i=1}^{N_u}$ regroupe l'ensemble des points sur les bords du domaine spatial et temporel pour rendre compte des conditions aux limites et de la condition initiale du problème. Par ailleurs les points N_f sont des points aléatoires du domaine spatial et temporel et puisqu'une approximation de \mathbf{u} est d'abord faite pour ensuite obtenir f , les points $f(t_f^i, x_f^i)$ sont connus $\forall i \in \mathbb{N}_*^{N_f}$. En fait, ce deuxième terme de notre fonction MSE vient pénaliser la valeur moyenne de f sur le domaine puisque l'on souhaite obtenir une solution satisfaisant la condition $f = 0$.

Si on avait un terme source dans (1.6), on pourrait se ramener au cas d'étude de l'article en intégrant ce terme source dans l'opérateur \mathcal{N} . Un point important souligné par les auteurs est que l'on ne possède pas de certitude sur la bonne généralisation du modèle ainsi créé, puisque la généralisation est liée notamment à la convergence du gradient de notre fonction à minimiser. Cependant sous réserve d'un problème bien posé, les résultats observés permettent de dire que la modélisation donne de bonnes solutions. Cette bonne solution ne peut être obtenue qu'en se donnant N_f suffisamment grand. En effet, puisque l'on cherche à minimiser MSE plus N_f est important plus les points (t_f^i, x_f^i) recouvrent une partie importante du domaine, et donc plus la valeur moyenne de f sera représentative.

1.3.3 Application

Les auteurs de [23] cherchent une solution de l'équation de Shrödinger suivante

$$ih_t + 0.5h_{xx} + |h|^2h = 0, \quad \forall x \in [-5, 5], \forall t \in [0, \frac{\pi}{2}], \quad (1.7 a)$$

$$h(0, x) = 2\text{sech}(x), \quad \forall x \in [-5, 5], \quad (1.7 b)$$

$$h(t, -5) = h(t, 5), \quad \forall t \in [0, \frac{\pi}{2}], \quad (1.7 c)$$

$$h_x(t, -5) = h_x(t, 5), \quad \forall t \in [0, \frac{\pi}{2}] \quad (1.7 d)$$

En posant $f := ih_t + 0.5h_{xx} + |h|^2h$, une définition de la fonction à minimiser peut être

$$MSE = MSE_0 + MSE_b + MSE_f,$$

$$MSE_0 := \frac{1}{N_0} \sum_{i=1}^{N_0} |h(0, x_0^i) - h_0^i|^2, \quad MSE_f := \frac{1}{N_f} \sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2,$$

$$MSE_b := \frac{1}{N_b} \sum_{i=1}^{N_b} (|h(t_b^i, -5) - h(t_b^i, 5)|^2 + |h_x(t_b^i, -5) - h_x(t_b^i, 5)|^2).$$

1.3.4 Recherche de solution pour des problèmes en temps discret

La démarche proposée en temps discret est sensiblement la même que celle en temps continu à quelques nuances près que l'on va pouvoir détailler. Comme on cherche une solution d'un problème à temps discret, on peut utiliser un schéma de Runge-Kutta pour exprimer la solution discrète pour un temps t_n fixé

$$\mathbf{u}^{n+c_i} = \mathbf{u}^n - \Delta_t \sum_{j=1}^q a_{ij} \mathcal{N}[\mathbf{u}^{n+c_j}], \forall i \in \mathbb{N}_*^q, \quad (1.8 a)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \Delta_t \sum_{j=1}^q b_j \mathcal{N}[\mathbf{u}^{n+c_j}]. \quad (1.8 b)$$

On note $(c_i)_{1 \leq i \leq q}$, $(a_{ij})_{1 \leq i, j \leq q}$ et $(b_i)_{1 \leq i \leq q}$, les paramètres du schéma de Runge-Kutta d'ordre q . On remarque que ce n'est pas tout à fait la forme usuelle d'un schéma de Runge-Kutta généralisé, en particulier les signes sont opposés. Cela vient du fait qu'initialement (1.6) est considérée donc le schéma est écrit à partir de $u_t = -\mathcal{N}[u]$. En fait, l'équation (1.6) est remplacée

$$\mathbf{u}^n = \mathbf{u}^{n+c_i} + \Delta_t \sum_{j=1}^q a_{ij} \mathcal{N}[\mathbf{u}^{n+c_j}], \forall i \in \mathbb{N}_*^q (:= \mathbf{u}_i^n), \quad (1.9 a)$$

$$\mathbf{u}^n = \mathbf{u}^{n+1} - \Delta_t \sum_{j=1}^q b_j \mathcal{N}[\mathbf{u}^{n+c_j}], (:= \mathbf{u}_{q+1}^n). \quad (1.9 b)$$

Dès lors, avec (1.8) et (1.9), la solution au temps t_{n+1} peut être approchée à partir de la solution au temps t_n et en utilisant un modèle neuronal. Le raisonnement est le suivant: un premier réseau de neurones est construit et approche \mathbf{u}^{n+1} sur la base de (1.8) puis, de la même façon que dans §1.3.2, le deuxième réseau de neurones est construit sur la base de (1.9) et possède les mêmes paramètres que le premier réseau. Les deux réseaux donnent donc en sortie respectivement les $q + 1$ équations du problème (1.8) et les $q + 1$ équations du problème (1.9). On propose une figure résumant la méthodologie (voir Figure 3).

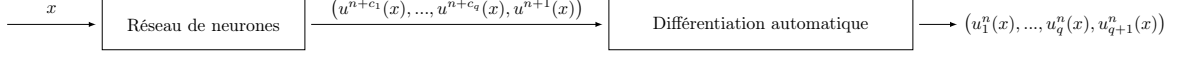


Figure 3. Schématisation du procédé pour passer d'une solution à temps n fixé à une solution à $n + 1$.

1.3.5 Application

Les auteurs de [23] mettent en oeuvre ce procédé pour l'équation d'Allen-Cahn,

$$u_t - 0.0001u_{xx} + 5u^3 - 5u = 0, \quad \forall x \in [-1, 1], \forall t \in [0, 1], \quad (1.10 a)$$

$$u(0, x) = x^2 \cos(\pi x), \quad \forall x \in [-1, 1], \quad (1.10 b)$$

$$u(t, -1) = u(t, 1), \quad \forall t \in [0, 1], \quad (1.10 c)$$

$$u_x(t, -1) = u_x(t, 1), \quad \forall t \in [0, 1]. \quad (1.10 d)$$

L'identification de $\mathcal{N}[u] := -0.0001u_{xx} + 5u^3 - 5u$ permet, lors de la phase d'apprentissage, la minimisation la fonction suivante

$$MSE = SSE_b + \sum_{j=1}^{q+1} \sum_{i=1}^{N_n} |u_j^n(x^{n,i}) - u^{n,i}|^2$$

où SSE_b est le terme d'erreur quadratique aux bords du domaine spatial. Le second terme n'est finalement qu'une pénalisation de la sortie du second réseau de neurone, puisque l'objectif est d'avoir $\partial_t \mathbf{u} + \mathcal{N}[\mathbf{u}] = 0$. De la même manière que pour un modèle en temps continu, les N_n points sur le domaine $[-1, 1]$ sont choisis aléatoirement.

Pour conclure, nous avons pu voir dans cette partie que l'apprentissage automatique, dans un contexte de dynamique des fluides numérique, est utilisé de manière diverse mais essentiellement dans un but de compréhension des phénomènes. En effet, abstraction faite de l'article [23] proposant une manière de résoudre les problèmes forts directement par l'utilisation d'outils à apprentissage profond, on fait appel à l'apprentissage pour gagner en précision sur les simulations numériques en agissant sur le schéma numérique ou sur le modèle de turbulence. La suite de notre démarche s'oriente dans cette voie là. Il nous faut cependant définir un contexte physique, c'est l'objet de la seconde partie.

2 Une approche volumes finis pour la résolution d'un problème Euler compressible stationnaire.

Après avoir introduit et appuyé, par une étude bibliographique, notre intérêt pour l'apprentissage automatique, il est question dans la suite de définir un cadre d'étude particulier pour notre travail. Dans cette partie, nous présentons le contexte physique qui nous intéresse. On spécifiera la nature des écoulements, la résolution théorique des équations et les problèmes associés. Il est aussi question dans cette partie de présenter la méthode de résolution numérique utilisée ainsi qu'une étude de la convergence du code numérique. Toutes les simulations seront effectuées, dans la suite, par le code *elsA* [5], développé à l'ONERA.

2.1 Le problème Euler compressible stationnaire.

Comme beaucoup de projets de recherche à l'ONERA se font en collaboration avec l'industrie, il paraît naturel d'étudier l'écoulement d'un fluide à des nombres de Mach aéronautiques usuels. On présente ici des travaux autour des régimes compressibles. Parmi ces phénomènes, on distingue trois régimes identifiés par leur nombre de Mach, notée M par la suite. Les écoulements dits subsoniques, $M \leq 0.8$, les écoulements transsoniques, $0.8 < M < 1$, et les écoulements supersoniques, $M > 1$. Il est important de noter que ces deux derniers régimes font apparaître des ondes de chocs, qui se traduisent par des discontinuités des variables d'état, sur le domaine d'étude. On aimerait, à ce stade, pouvoir identifier des paramètres optimaux liés à la réécriture discrète des équations dans l'espace pour la mise en place d'une méthode volumes finis sur laquelle s'appuie le code *elsA* [5].

2.1.1 Présentation générale du problème

On choisit de limiter notre étude aux fluides parfaits, en négligeant tout phénomène physique de dissipation, c'est pourquoi on s'intéresse à la résolution numérique du système des équations d'Euler

$$\frac{\partial \mathbf{W}}{\partial t} + \text{div}(\mathcal{F}) = \mathbf{0} \quad (2.1)$$

où \mathbf{W} est le vecteur de variables d'état. On note que \mathbf{W} peut s'écrire comme une fonction de t et \bar{x} . Et de la même façon, le flux physique \mathcal{F} peut s'écrire comme une fonction de \mathbf{W} ([10])

$$\begin{aligned} \mathbf{W} : \mathbb{R}_+ \times \mathbb{R}^d &\rightarrow \mathbb{R}^{d+2} & \mathcal{F} : \mathbb{R}^{d+2} &\rightarrow \mathbb{R}^{d(d+2)} \\ (t, \bar{x}) &\mapsto [\rho(t, \bar{x}), \rho(t, \mathbf{x})\bar{V}(t, \bar{x}), \rho(t, \bar{x})E(t, \bar{x})] & \mathbf{W} &\mapsto [f, g, h] \end{aligned}$$

$$f := \begin{pmatrix} \rho u \\ \rho u^2 + p_s \\ \rho uv \\ \rho uw \\ \rho uH \end{pmatrix}, \quad g := \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p_s \\ \rho vw \\ \rho vH \end{pmatrix}, \quad h := \begin{pmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p_s \\ \rho wH \end{pmatrix}.$$

2.1.2 Complexité théorique du problème

Avant de spécifier une méthode pour la résolution numérique du système (2.1), il convient de s'intéresser, comme tout problème faisant intervenir des équations différentielles ou aux dérivées partielles, à la

notion de problème bien posé, au sens d'Hadamard. Pour se faire, il faut encore spécifier complètement le problème associé à (2.1). Dans un contexte aéronautique externe, on souhaite simuler l'écoulement d'un fluide autour d'un objet solide de paroi Γ_b et sur un domaine spatial d'écoulement Ω suffisamment grand pour que la simulation soit représentative (voir Figure 4). Ce domaine est délimité par une frontière Γ_∞ généralement très éloignée¹ de Γ_b . Il faut donc spécifier des conditions aux limites en plus d'une condition initiale.

$$\frac{\partial \mathbf{W}}{\partial t} + \text{div}(\mathcal{F}) = \mathbf{0}, \quad \forall (t, \bar{x}) \in \mathbb{R}_*^+ \times \Omega, \quad (2.2 a)$$

$$\bar{V}_b \cdot \bar{n}_b = 0, \quad \forall (t, \bar{x}) \in \mathbb{R}_*^+ \times \Gamma_b, \quad (2.2 b)$$

$$C_{in}(\mathbf{W}) = C_{in}(\mathbf{W}_\infty), \quad \forall (t, \bar{x}) \in \mathbb{R}_*^+ \times \Gamma_\infty, \quad (2.2 c)$$

$$\mathbf{W}(t = 0, \cdot) = \mathbf{W}_0, \quad \forall \bar{x} \in \Omega. \quad (2.2 d)$$

où C_{in} désigne les grandeurs convectées par les caractéristiques entrantes dans le domaine fluide (dans le cadre de la théorie des caractéristiques pour les systèmes hyperboliques), \bar{V}_b est la norme de la vitesse au profil et \bar{n}_b est la normale au profil. Un problème bien posé au sens d'Hadamard est un problème pour lequel il existe une unique solution stable. Cependant pour le problème (2.2), il n'est pas possible de justifier l'existence d'une unique solution. Certains travaux ont été effectués sur des problèmes similaires. On peut par exemple citer Bardos pour sa démonstration de l'existence d'une unique solution aux équations de moments du système Euler en dimension deux ([2]), mais rien de tel n'a été effectué sur le système complet. Néanmoins, comme on ferait pour la résolution de problèmes elliptiques, on peut introduire la notion de solution faible.

2.1.3 Notion de solution faible

On se contente dans cette partie de faire un rappel sur la notion de solution faible pour les équations hyperboliques scalaire en dimension un, à partir de [19]. On considère alors le problème suivant

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(f(u)) = 0, \quad \forall (t, x) \in \mathbb{R}_*^+ \times \mathbb{R}, \quad (2.3 a)$$

$$u(t = 0, x) = u_0(x), \quad \forall x \in \mathbb{R}. \quad (2.3 b)$$

où g est supposée convexe dérivable et on note $a := f'$.

Définition 2.1 On appelle solution faible du problème (2.3) toute solution $u \in L_{loc}^\infty$ de $\mathbb{R} \times [0; T]$ telle que $\forall \phi \in \mathcal{C}^1(\mathbb{R} \times [0; T])$ à support compact, on ait

$$\int_{[0; T]} \int_{\mathbb{R}} \left(u \frac{\partial \phi}{\partial t} + f(u) \frac{\partial \phi}{\partial x} \right) dx dt = - \int_{\mathbb{R}} u_0(x) \phi(0, x) dx$$

On introduit cette notion de manière à définir une solution vérifiant notre problème (2.3) en tout point, c'est-à-dire à la fois dans les zones où elle est régulière et dans les zones en présence de chocs. Pour définir ce dernier point, on introduit la proposition suivante.

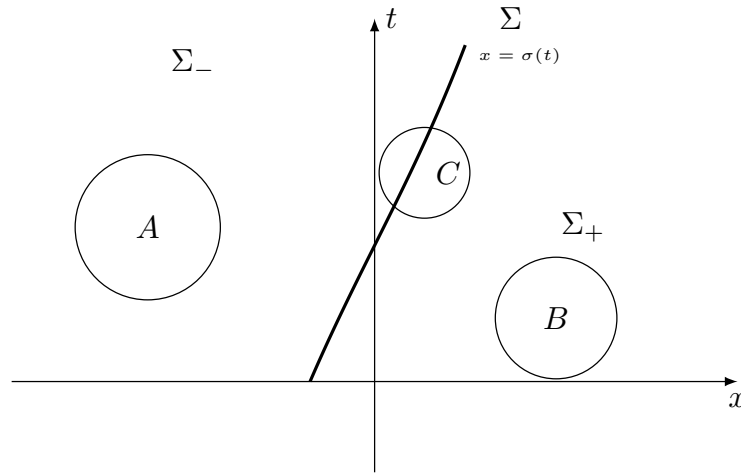
Proposition 1 (*Relation de Rankine-Hugoniot*) Soit Σ une courbe de discontinuité du plan (x, t)

¹ Dans le contexte aéronautique de profils, la frontière délimitant le domaine est située à une distance de 150 à 300 cordes du profil. La corde étant la longueur du bord de fuite au bord d'attaque du profil.

d'équation $x = \sigma(t)$. Pour toute solution faible du problème (2.3), en notant $u(t, x) := u^-(t, x)$ pour $x < \sigma(t)$ et $u(t, x) := u^+(t, x)$ pour $x > \sigma(t)$, on a

$$(u^+ - u^-)\sigma'(t) = f(u^+) - f(u^-)$$

Démonstration :



On suppose que l'on dispose d'une solution faible de (2.3) telle qu'il existe une courbe Σ du plan (x, t) d'équation $x = \sigma(t)$ avec $u(t, x) := u^-(t, x)$ pour $x < \sigma(t)$ et $u(t, x) := u^+(t, x)$ pour $x > \sigma(t)$. On attend de u qu'elle soit suffisamment régulière, \mathcal{C}^1 à minima. Ce raisonnement étant local, il s'étend à un nombre quelconque de discontinuités. Tout d'abord, plaçons nous dans une zone quelconque A de Σ^- et considérons $\phi \in \mathcal{C}^1(\mathbb{R} \times [0; T])$ à support compact dans A . On a

$$\iint_A \left(u^- \frac{\partial \phi}{\partial t} + f(u^-) \frac{\partial \phi}{\partial x} \right) = 0.$$

Les régularités supposées pour ϕ et u nous permettent d'effectuer des intégrations par parties pour obtenir

$$\iint_A \left[\frac{\partial u^-}{\partial t} + \frac{\partial}{\partial x} f(u^-) \right] \phi = 0.$$

Ceci étant vrai pour quelque soit ϕ à support compact dans A et pour tout compact $A \subset \Sigma^-$, on en déduit que

$$\frac{\partial u^-}{\partial t} + \frac{\partial}{\partial x} f(u^-) = 0.$$

Donc u^- est solution au sens classique sur Σ^- . Un raisonnement similaire peut être effectué pour montrer que u^+ est solution au sens classique sur Σ^+ . On considère maintenant une fonction ϕ à support dans C . Notons $\mathbf{n} := (n_x, n_t)$ le vecteur unitaire normal à la courbe de discontinuité orienté vers les x croissants. On a $\mathbf{t} := (\sigma'(t), 1)$ le vecteur unitaire tangent à la courbe et donc $\mathbf{n} := \frac{1}{\sqrt{1+(\sigma'(t))^2}}(1, -\sigma'(t))$ puisque \mathbf{t}

et \mathbf{n} sont orthogonaux. Par définition de la solution faible, on a

$$\iint_{\mathbb{R} \times [0; T]} u \frac{\partial \phi}{\partial t} + f(u) \frac{\partial \phi}{\partial x} = 0$$

$$\iint_{C \cap \Sigma^-} u^- \frac{\partial \phi}{\partial t} + f(u^-) \frac{\partial \phi}{\partial x} + \iint_{C \cap \Sigma^+} u^+ \frac{\partial \phi}{\partial t} + f(u^+) \frac{\partial \phi}{\partial x} = 0$$

En parallèle, la formule de Green-Ostrogradski nous permet d'écrire

$$\begin{aligned} \iint_{C \cap \Sigma^-} u^- \frac{\partial \phi}{\partial t} &= - \iint_{C \cap \Sigma^-} \frac{\partial u^-}{\partial t} \phi + \iint_{C \cap \Sigma} u^- \phi n_t, \\ \iint_{C \cap \Sigma^-} f(u^-) \frac{\partial \phi}{\partial x} &= - \iint_{C \cap \Sigma^-} \frac{\partial}{\partial x} (f(u^-)) \phi + \iint_{C \cap \Sigma} f(u^-) \phi n_x, \\ \iint_{C \cap \Sigma^+} u^+ \frac{\partial \phi}{\partial t} &= - \iint_{C \cap \Sigma^+} \frac{\partial u^+}{\partial t} \phi - \iint_{C \cap \Sigma} u^+ \phi n_t, \\ \iint_{C \cap \Sigma^+} f(u^+) \frac{\partial \phi}{\partial x} &= - \iint_{C \cap \Sigma^+} \frac{\partial}{\partial x} (f(u^+)) \phi - \iint_{C \cap \Sigma} f(u^+) \phi n_x. \end{aligned}$$

En utilisant ces équations dans la formule précédente, la réécriture nous donne

$$\begin{aligned} \iint_{C \cap \Sigma^-} \left[\frac{\partial u^-}{\partial t} + \frac{\partial}{\partial x} f(u^-) \right] \phi + \iint_{C \cap \Sigma^+} \left[\frac{\partial u^+}{\partial t} + \frac{\partial}{\partial x} f(u^+) \right] \phi \\ + \iint_{C \cap \Sigma} [(u^- - u^+) n_t + (f(u^-) - f(u^+)) n_x] \phi = 0 \end{aligned}$$

Comme on a montré que u^+ et u^- sont solutions de (2.3) sur leur zone respective de régularité, on peut dire que les deux premiers termes sont nuls. De plus, comme ϕ est quelconque à support compact dans C et que le compact C est lui aussi quelconque le long de Σ , on a la relation de Rankine-Hugoniot:

$$(u^+ - u^-) \sigma'(t) = f(u^+) - f(u^-)$$

□

Ainsi on peut réécrire les équations d'Euler propres au problème (2.2) en présence d'ondes de choc, et lignes de glissement. Notons $\Sigma(t)$ la surface de discontinuité se déplaçant dans le temps à la vitesse \bar{V}_Σ et n la normale à $\Sigma(t)$ orientée dans le sens $-$ vers $+$. Dans ce cas, on écrit avec $\|\cdot\|$ l'opérateur de saut défini tel que $\|\cdot\| = \cdot_+ - \cdot_-$

$$\begin{cases} \frac{d\rho}{dt} + \rho \nabla \cdot \bar{V} = 0, \\ \|\rho(\bar{V} - \bar{V}_\Sigma)\| \cdot n = 0. \end{cases}$$

$$\begin{cases} \rho \frac{d\bar{V}}{dt} = -\nabla p_s, \\ \|\bar{V}\| \rho(\bar{V} - \bar{V}_\Sigma) \cdot n = -\|p\| n. \end{cases}$$

$$\begin{cases} \rho \frac{d}{dt} \left(e + \frac{|V|^2}{2} \right) = -\nabla \cdot \rho \bar{V}, \\ \rho \|e + \frac{|V|^2}{2}\| (\bar{V} - \bar{V}_\Sigma) \cdot n = -\|p \bar{V}\| n. \end{cases}$$

De cette façon on définit complètement une solution pour les écoulements transsoniques et supersoniques pour lesquels la solution introduit des surfaces de discontinuités sur le domaine.

2.1.4 Le problème Euler compressible stationnaire

Dans la suite, pour vraiment s'intéresser aux équations spatiales et tenter d'identifier des paramètres, on choisit d'étudier une version stationnaire du problème (2.2)

$$\operatorname{div}(\mathcal{F}) = \mathbf{0}, \quad \forall \bar{x} \in \Omega, \quad (2.4 a)$$

$$\bar{\mathbf{V}}_b \cdot \bar{\mathbf{n}}_b = 0, \quad \forall \bar{x} \in \Gamma_b, \quad (2.4 b)$$

$$C_{in}(\mathbf{W}) = C_{in}(\mathbf{W}_\infty), \quad \forall \bar{x} \in \Gamma_\infty. \quad (2.4 c)$$

Il est question, à présent, d'écrire le problème sous sa forme discrète de manière à le résoudre numériquement par une méthode volumes finis.

2.2 La méthode volumes finis du code elsA.

Après avoir défini notre cadre théorique, on se propose dans cette partie de présenter les éléments nécessaires à la résolution numérique du problème (2.4). Cette résolution numérique se fait par l'intermédiaire du solveur volumes finis *elsA* ([26] et [5]). Comme nous fixons notre étude sur les écoulements externes, nous allons utiliser un maillage de Ω autour du profil de contour Γ_b .

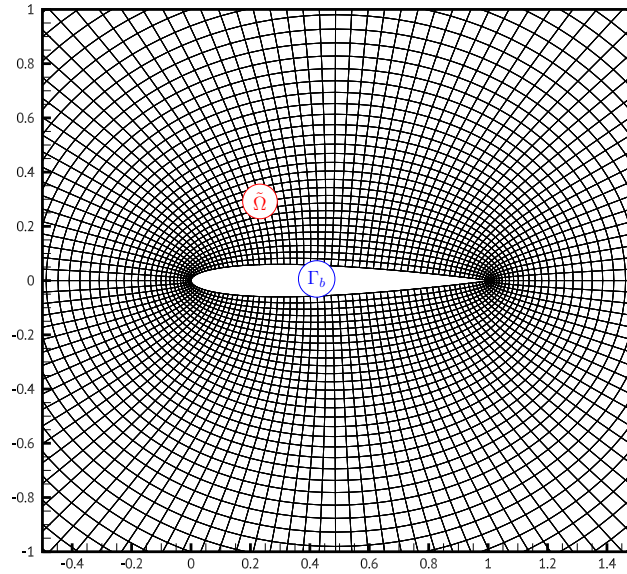


Figure 4. Représentation d'un maillage $\tilde{\Omega}$ autour de la paroi Γ_b .

2.2.1 Présentation de la méthode volumes finis

On souhaite ici écrire les équations faisant intervenir les variables locales de \mathbf{W} par l'intermédiaire des échanges de flux entre les différentes cellules du maillage. Dans la suite, les inconnues discrètes, donc les variables locales de \mathbf{W} sont situées au centre des cellules et seront notées $\psi_{\frac{1}{2},c}$. En intégrant sur Ω

l'équation $\operatorname{div}(\mathcal{F}) = \mathbf{0}$ et grâce à formule de Green-Ostrogradski, on peut écrire

$$\begin{aligned} \iiint_{\Omega} \operatorname{div}(\mathcal{F}) &:= \iiint_{\Omega} \nabla \cdot \mathcal{F} d\Omega = \mathbf{0} \\ \iiint_{\Omega} \nabla \cdot \mathcal{F} d\Omega &= \iint_{\partial\Omega} \mathcal{F} \cdot n_S dS = \mathbf{0} \end{aligned} \quad (2.5)$$

avec n_S la normale sortante de la surface $\partial\Omega$. Dans la suite on se donne un maillage de taille $\mathcal{N} \times \mathcal{N}$ de l'espace Ω . La taille \mathcal{N} est le nombre de points que l'on se donne sur la paroi, pour définir la forme solide Γ_b . On dispose alors d'un maillage en "O" (qui sera traité pour le calcul de manière déstructurée) de $(\mathcal{N} - 1) \times (\mathcal{N} - 1)$ cellules notées $\{c_i\}_{i=1}^N$, avec $N = (\mathcal{N} - 1)^2$. On note $\tilde{\Omega}$, l'espace Ω ainsi discrétisé. Soit c_i une cellule quelconque de $\tilde{\Omega}$. On rappelle que $\bar{\Omega} := \bigcup_{c_i \in \tilde{\Omega}} \bar{c}_i$ et $\forall i, j \in \mathbb{N}_*^N, i \neq j, c_i \cap c_j = \emptyset$.

On a alors, à la cellule c_i et en notant F_{c_i} le nombre de faces composant la cellule c_i , l'équation (2.5) qui peut s'écrire

$$\sum_{f \in F_{c_i}} \iint_{\sigma_f} \mathcal{F}(\psi_{\frac{1}{2}}) \cdot n_f d\sigma_f = 0$$

avec n_f la normale sortante de la face f . On approche ensuite la double intégrale sur la surface σ_f avec une formule de quadrature. Ce choix de formule, comme mentionné dans [26], dépend d'une stratégie de reconstruction de variables discrètes nécessaire pour la montée en ordre de certains schémas. Pour les besoins du stage et puisque l'on fixe notre schéma d'étude par la suite, l'utilisation d'une formule du point milieu paraît adaptée puisqu'elle est compatible avec une précision d'ordre 2 en espace (précision usuelle des schémas volumes finis).

$$\sum_{f \in F_{c_i}} \mathcal{H}_c^f(\cdot) \|\sigma_f\| = 0$$

où $\|\sigma_f\|$ est la surface sur laquelle on étudie l'échange de flux et \mathcal{H}_c^f l'approximation de $\mathcal{F}(\psi_{\frac{1}{2}}) \cdot n_f$, appelée flux numérique. À ce stade, le flux numérique n'est pas explicité puisque l'objet du travail à réaliser est justement d'étudier les différentes possibilités de calcul de ce terme. C'est ce que nous allons voir par la suite.

2.2.2 Théorème de Lax-Wendroff et schémas numériques

Il est important, en tenant compte de la partie §2.1, de pouvoir réaliser une bonne approximation de $\mathcal{F}(\psi_{\frac{1}{2}}) \cdot n_f$ menant à la définition de \mathcal{H}_c^f . Pour ce faire, on doit définir complètement la notation \mathcal{H}_c^f dans l'espace.

Définition 2.2 Soit c_i une cellule de $\tilde{\Omega}$ et $f \in F_{c_i}$. Notons $\psi_{\frac{1}{2}}$ les inconnues à la cellule c_i . On appelle flux numérique selon la direction n_f , l'approximation de $\mathcal{F}(\psi_{\frac{1}{2}}) \cdot n_f$, noté $\mathcal{H}_c^f(\cdot, n_f)$. La notation « \cdot » indique que les cellules voisines choisies pour réaliser l'approximation dépendent entièrement du choix du schéma numérique.

La suite du travail consiste à choisir un schéma numérique permettant d'approcher une solution faible. On peut identifier des schémas pouvant être considérés comme de bons candidats pour réaliser l'approximation. C'est l'objet du théorème de Lax-Wendroff.

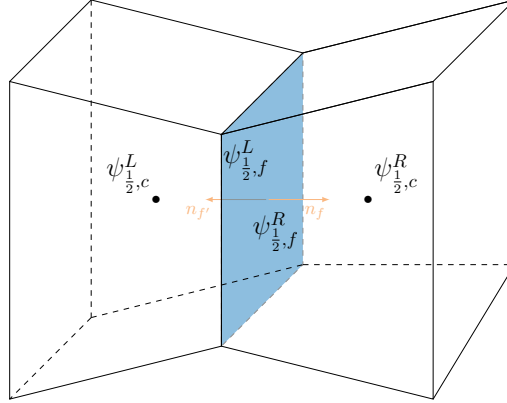


Figure 5. Représentation et notation à l'interface entre une maille L et une maille R de $\tilde{\Omega}$.

Théorème 2.3 (Théorème de Lax-Wendroff) Soit \mathcal{H}_c^f un schéma sous forme conservative qui soit consistant avec l'équation principale du problème (2.4). Alors, si pour toute suite $\Delta \bar{x}_k \xrightarrow[k \rightarrow +\infty]{} 0$, la suite de fonctions $(\psi_\Delta)_k$ constantes par morceaux et définies par $(\psi_\Delta)_k(\bar{x}) = \psi_{\frac{1}{2}}$ converge vers une fonction $\tilde{\mathbf{W}}$ telle que si, pour K un compact quelconque de \mathbb{R}^d , on a

$$\|(\psi_\Delta)_k - \tilde{\mathbf{W}}\|_{L^\infty(K)} \xrightarrow[k \rightarrow +\infty]{} 0$$

alors $\tilde{\mathbf{W}}$ est solution faible de (2.4).

On déduit de ce théorème que les schémas conservatifs consistants sont de bons candidats pour approcher les solutions faibles. On décide à ce stade d'écrire le schéma centré de Jameson-Schmidt-Turkel (JST [12]) pour vérifier s'il peut être identifié comme un bon candidat pour mener à l'obtention de bonnes solutions.

$$\begin{aligned} \mathcal{H}_c^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f) &= \frac{1}{2} \left[\mathcal{F}(\psi_{\frac{1}{2},c}^L) \cdot n_f + \mathcal{F}(\psi_{\frac{1}{2},c}^R) \cdot n_f \right] - \epsilon_f^{(2)} \lambda_f \left(\psi_{\frac{1}{2},c}^L - \psi_{\frac{1}{2},c}^R \right) + \epsilon_f^{(4)} \lambda_f \left(\delta^2 \psi_{\frac{1}{2},c}^L - \delta^2 \psi_{\frac{1}{2},c}^R \right) \\ \delta^2 \psi_c^L &:= \sum_{f \in F_{c_i}} (\psi_c^{R,f} - \psi_c^L), \quad \epsilon_f^{(2)} := k_2 \max(\nu_{c_i}, \nu_{c_i}^f), \quad \epsilon_f^{(4)} := \max(0, k_4 - \epsilon_f^{(2)}). \end{aligned}$$

Proposition 2 Le schéma JST est conservatif et consistant avec l'équation principale du problème (2.4).

Démonstration : La conservativité du schéma est directement issue de son écriture sous forme de flux numérique.

La consistance du schéma est visible en évaluant le flux numérique sur un état \mathbf{W} constant uniforme. Ceci étant valable quelque soit l'état courant, le schéma nous permet de retrouver l'état global. Comme la partie dissipative, lors d'une évaluation en un même état, revient à faire une somme de termes nuls, il

nous reste la partie convective suivante

$$\mathcal{H}_c^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f) = \frac{1}{2} \left[2\mathcal{F}(\psi_{\frac{1}{2},c}) \cdot n_f \right] = \mathcal{F}(\psi_{\frac{1}{2},c}) \cdot n_f.$$

□

Une autre écriture du schéma JST est possible, appelée *skew – symmetric*, par opposition à l'écriture initiale appelée *divergence* et présentée ci-dessus, que l'on définit de la façon suivante

$$\begin{aligned} \mathcal{H}_c^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f) &= \mathcal{F} \left(\frac{1}{2} \left[\psi_{\frac{1}{2},c}^L + \psi_{\frac{1}{2},c}^R \right] \right) \cdot n_f - \epsilon_f^{(2)} \lambda_f \left(\psi_{\frac{1}{2},c}^L - \psi_{\frac{1}{2},c}^R \right) + \epsilon_f^{(4)} \lambda_f \left(\delta^2 \psi_{\frac{1}{2},c}^L - \delta^2 \psi_{\frac{1}{2},c}^R \right) \\ \delta^2 \psi_c^L &:= \sum_{f \in F_{c_i}} (\psi_c^{R,f} - \psi_c^L) \quad \epsilon_f^{(2)} := k_2 \max(\nu_{c_i}, \nu_{c_i}^f) \quad \epsilon_f^{(4)} := \max(0, k_4 - \epsilon_f^{(2)}) \end{aligned}$$

Proposition 3 *Le schéma JST skew – symmetric est conservatif et consistant avec l'équation principale du problème (2.4).*

Démonstration : Il est possible d'effectuer le même raisonnement que pour le schéma JST *divergence*.

□

On peut donc dire que quelque soit son écriture, le schéma JST est un bon candidat pour l'obtention de solution faible à notre problème. Néanmoins le théorème de Lax-Wendroff repose sur une hypothèse de convergence en maillage dont il faut pouvoir s'assurer. C'est l'objet de notre étude sur la convergence du code *elsA*.

2.3 Étude de la convergence du code *elsA*.

Il est question dans cette partie de s'assurer que le code *elsA*, sous réserve d'utilisation d'un schéma satisfaisant du point de vue du théorème de Lax-Wendroff, puisse nous permettre d'atteindre une solution faible. On commence par s'intéresser à certains maillages et au raffinement pour fixer le contexte de simulation.

2.3.1 Introduction de différents maillages

Pour la simulation, les maillages acceptés par le code peuvent être structurés ou non-structurés. Pour un maillage structuré, chaque cellule est identifiable dans l'espace grâce à des indices qui lui sont associées. Pour un maillage non-structuré, il est nécessaire d'avoir des tableaux de connectivité entre les cellules et leurs faces, les faces et leurs deux cellules voisines. C'est cette architecture que nous gardons et nous présentons ici quatre maillages (voir Figure 6 et Figure 7) qui nous seront utiles pour la suite. Le maillage avec rapport d'aspect constant et le maillage régulier sont des maillages souvent utilisés en simulation pour la dynamique des fluides. Pour preuve, le maillage régulier est directement issu d'un article de Jameson et Vassberg (voir [27]). Les maillages de la figure 7 ont pour base le même maillage ([27]) mais ont été déformés de telle sorte que le maillage non-adapté ait des mailles resserrées au milieu du profil ainsi qu'à une longueur d'environ trois cordes du profil (non visible sur l'image). Le maillage dit aléatoire a été obtenu en ajoutant un aléa sur la position des points du maillage régulier. Bien que ces maillages

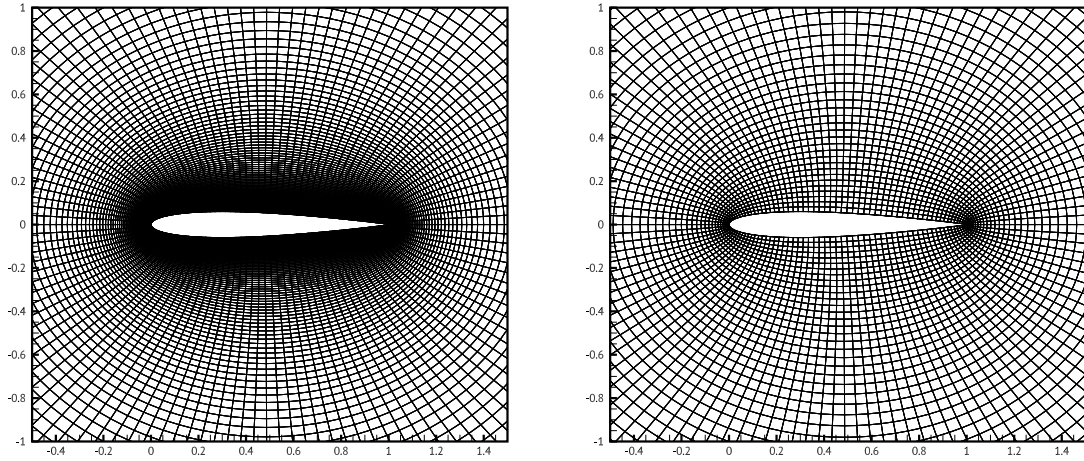


Figure 6. Maillage avec rapport d'aspect constant (à gauche) et maillage régulier (à droite).

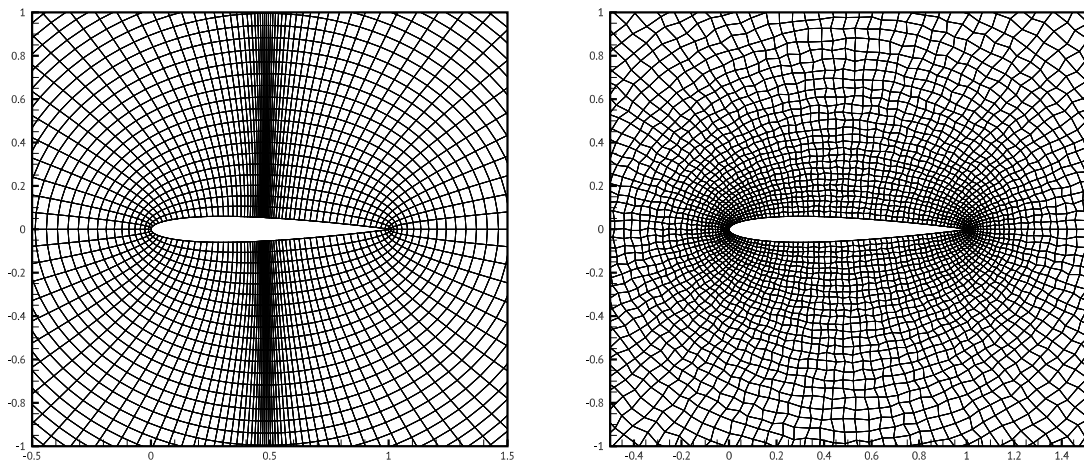


Figure 7. Maillage avec raffinement non-adapté (à gauche) et maillage aléatoire (à droite).

puissent être représentés par un indicage structuré, le code numérique ne les considère que par une vision non-structurée comme explicité précédemment. C'est un choix qui a été fait et qui pourra être justifié par la suite.

2.3.2 Convergence en maillage

On cherche à s'assurer de la convergence du champ solution par raffinement de maillage dans le code *elsA*. Pour ce faire nous choisissons de garder le maillage avec rapport d'aspect constant et resserrement à la paroi qui historiquement est un des premiers types de maillages utilisé en dynamique des fluides

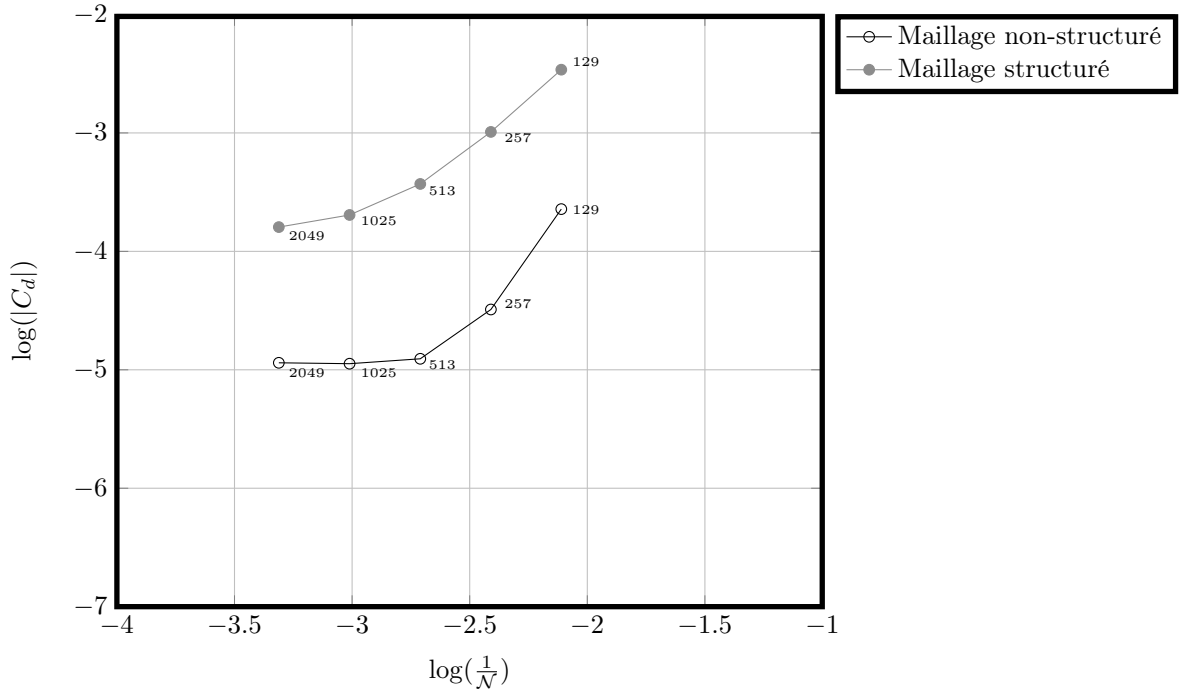


Figure 8. Courbe de convergence du code *elsA* à partir de C_d .

et qui paraît naturellement adapté dans le sens où on souhaite être d'autant plus précis dans le calcul que l'on est près de notre surface d'intérêt. Et de fait, avec un tel maillage, plus on se rapproche de la paroi du profil, plus les mailles sont petites et plus on s'en éloigne plus elle sont grandes. Nous reprenons la démarche proposée [27] et nous traçons pour chaque taille de maillage \mathcal{N} la courbe de convergence à partir des données sur la traînée C_d (voir Figure 8). On rappelle que pour un écoulement subsonique de fluide parfait la traînée C_d est nulle. On note que l'on a tracé la courbe en maillage structuré et en maillage non-structuré et cela nous permet de justifier le fait que l'on garde la vision non-structurée pour la suite. En effet, la courbe non-structurée est bien en dessous de celle associée au calcul structuré. Le calcul est donc plus précis à convergence, quelque soit la taille du maillage. Ceci est notamment lié aux travaux réalisés dans [26] autour de l'amélioration de la précision des schémas numériques en maillage non-structuré. Enfin cette figure nous permet d'explicitier la convergence effective du code même si on constate l'apparition d'un pallier sur la précision du coefficient C_d pour des grandes tailles de maillage, que l'on soit en structuré ou non-structuré.

On a pu, dans cette partie, expliciter le contexte physique qui nous intéresse et justifier l'utilisation de maillages non-structurés. Le schéma de Jameson, en le considérant comme un candidat pouvant mener à l'obtention d'une solution faible, constitue un vrai point d'ancrage dans le déroulé du stage. On souhaite donc chercher par la suite des paramètres pouvant influencer sur la précision obtenue sur la solution stationnaire à convergence. De manière à mesurer concrètement cette précision, il est nécessaire, dans une démarche non-supervisée, d'introduire un critère de sélection (puisque la traînée C_d est non nulle pour des écoulements transsoniques ou supersoniques). Tous ces points constituent des étapes importantes de la partie suivante.

3 Une méthodologie pour l'identification de bons paramètres.

On aimerait dans cette partie mettre en place un processus automatique pour identifier des paramètres du schéma JST permettant d'obtenir une bonne approximation numérique d'une solution faible. Comme évoqué précédemment, pour identifier une bonne solution et puisque la variable C_d ne nous permet pas de travailler avec des écoulements transsoniques ou supersoniques, il est nécessaire de définir un nouveau critère. C'est l'objet de la première partie. La suite concerne l'identification des paramètres potentiellement influents et intéressants du point de vue de ce critère. Enfin dans une dernière partie, on explicite concrètement la méthodologie que l'on applique pour réaliser un apprentissage automatique et identifier de bonnes valeurs des paramètres retenus.

3.1 Définition d'un critère d'intérêt.

Dans ce qui a été fait précédemment, on a pu voir la nécessité de définir un critère, que l'on notera ε , indépendant de l'écoulement, permettant de dire si un choix de paramètres numériques est plus intéressant qu'un autre. En l'occurrence, des variables d'état telle que l'enthalpie totale ont des propriétés particulières pouvant servir de base à notre raisonnement en considérant un écoulement stationnaire. On rappelle dans cette partie certaines propriétés, leurs utilités pour définir un critère et on cherche ensuite à justifier la cohérence de ce critère du point de vue de la dynamique des fluides. Pour effectuer des comparaisons et justifier de l'écriture de notre critère des calculs numériques ont été effectués à partir d'un cas en deux dimensions, l'aile NACA0012 ([17]). Le calcul des flux numériques est effectué par un schéma de Roe avec reconstruction MUSCL ([7] et [22]) afin d'obtenir une précision d'ordre deux en espace.

3.1.1 Propriété de l'enthalpie totale pour un écoulement Euler compressible

On rappelle la relation $H = h + \frac{|\bar{V}|^2}{2} = C_p T + \frac{|\bar{V}|^2}{2}$ pour un gaz parfait. Or dans le cadre d'un écoulement de fluide parfait sans force de volume, on a en particulier que

$$\rho \frac{d}{dt} \left(C_p T + \frac{|\bar{V}|^2}{2} \right) = \frac{\partial p_s}{\partial t}$$

Cependant dans un contexte stationnaire, on a $\frac{\partial}{\partial t} = 0$. Donc finalement $\frac{dH}{dt} = 0$, c'est-à-dire que H est constante sur toutes les lignes de courant de l'écoulement. Mais puisque l'on fixe un état uniforme à l'infini amont, H est constante sur tout le domaine. On note qu'une démarche similaire peut être effectuée pour justifier le fait que l'entropie S est constante pour les écoulements subsoniques.

3.1.2 Définition d'un critère d'intérêt

Puisque le champ d'enthalpie totale doit être constant pour un écoulement de fluide parfait stationnaire, on peut s'intéresser, à convergence, à l'écart entre le champ d'enthalpie totale de la solution numérique obtenue et sa valeur théorique. Comme rappelé ci-dessus, un écoulement externe de fluide parfait est caractérisé par son nombre de Mach et l'orientation de sa vitesse. Pour la simulation, on spécifie un nombre de Mach référence noté M_∞ et on adimensionne les variables d'état par $\rho_{ad} = 1$ et $\bar{V}_{ad} = 1$ qui sont la masse volumique et le module de la vitesse à l'infini. On peut, par définition de la pression statique et du nombre de Mach définir

$$p_\infty = \frac{1}{\gamma M_\infty^2}, \quad E_\infty = \frac{1}{\gamma(\gamma - 1)M_\infty^2} + \frac{1}{2}$$

On peut alors définir une enthalpie totale de référence H_∞ par la définition $H := E + \frac{p_s}{\rho}$, ce qui donne

$$H_\infty = \frac{1}{(\gamma - 1)M_\infty^2} + \frac{1}{2}$$

On peut alors penser à calculer une erreur absolue à partir du champ solution obtenu à convergence en pondérant par le volume de la cellule. En effet, par les mêmes relations explicitées ci-dessus, on peut recalculer l'enthalpie totale propre à une cellule du maillage à partir des valeurs de ρ , ρu , ρv , ρw et ρE issues de la simulation. On peut aussi rappeler que les variations d'enthalpie totale d'une cellule à l'autre vont être d'autant plus importantes que les cellules seront proches de la paroi autour de laquelle on simule un écoulement. Ceci s'expliquant directement par la variation de la pression d'une cellule à l'autre en présence de discontinuités. Et inversement, si on s'éloigne du profil pour se rapprocher du bord infini, la variation d'enthalpie totale d'une cellule à l'autre doit être très petite numériquement. On aimerait ainsi avoir un calcul pour une région restreinte autour de la paroi. Ces informations sont importantes pour définir un critère, d'autant plus que pour des maillages non-réguliers et cohérent d'un point de vue de la dynamique des fluides, plus on s'éloigne du profil ou de la paroi, plus les cellules sont importantes en taille. On propose alors de définir ε tel que

$$\varepsilon := \frac{\sum_{c_i \in \mathcal{L}} |H_{c_i} - H_\infty| \mathcal{V}_{c_i}}{H_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}}. \quad (3.1)$$

On désigne par \mathcal{L} l'ensemble des cellules proches du profil, c'est-à-dire les cellules dont le barycentre est situé à une distance inférieure à l de la paroi. On spécifiera dans §4 une valeur numérique pour l . On pondère la différence entre l'enthalpie totale calculée et l'enthalpie totale de référence par le volume de la cellule pour garder une cohérence physique: plus la cellule est grande plus elle doit avoir de l'influence sur notre critère. La somme sur les volumes des cellules au dénominateur permet d'avoir un calcul de ε qui soit comparable quelque soit le maillage sur lequel on l'effectue. En effet, d'un maillage à l'autre, l'ensemble \mathcal{L} peut ne pas contenir les mêmes cellules et ne pas correspondre exactement au même volume total. De la même façon, on divise par H_∞ dans l'idée de comparer les valeurs numériques pour différents écoulements. Pour illustrer notre démarche, on a tracé sur les figures 9 et 10, les variables M , p_s et $\frac{|H - H_\infty|}{H_\infty}$.

3.1.3 Comparaison de maillages d'un point de vue dynamique des fluides

On aimerait pouvoir justifier la définition de notre critère. En particulier, on aimerait s'assurer que plus le maillage est adapté à notre simulation de fluide parfait, plus le critère précédemment calculé est faible. On aimerait aussi s'assurer que l'on retrouve une tendance liée à la définition de l'enthalpie, puisqu'on a noté que les variations doivent être d'autant plus fortes en présence de discontinuité. Pour ce faire, on a repris les maillages détaillés et introduits dans la partie §2.3.1 pour deux tailles différentes $\mathcal{N} = 129$ et $\mathcal{N} = 513$, et on a calculé ε pour trois écoulements: $M_\infty = 0.4$, $M_\infty = 0.85$ et $M_\infty = 1.5$ (voir Figure 11).

On remarque plusieurs choses sur la figure 11. Tout d'abord comme la pression intervient directement dans l'expression de l'enthalpie totale, on remarque que le critère ε prend des valeurs plus élevées pour les écoulements faisant apparaître des discontinuités ($M_\infty = 0.85$ et $M_\infty = 1.5$). On observe aussi, de manière générale et quelque soit l'écoulement, que les valeurs obtenues pour les maillages non-adaptés et aléatoires sont significativement plus grandes que pour les maillages resserrés et réguliers. On peut expliquer cela en faisant référence à la manière dont on a introduit les maillages dans la partie §2.3.1. En effet, les maillages non-adaptés et aléatoires, bien qu'issus d'un maillage régulier, ne sont pas adaptés à

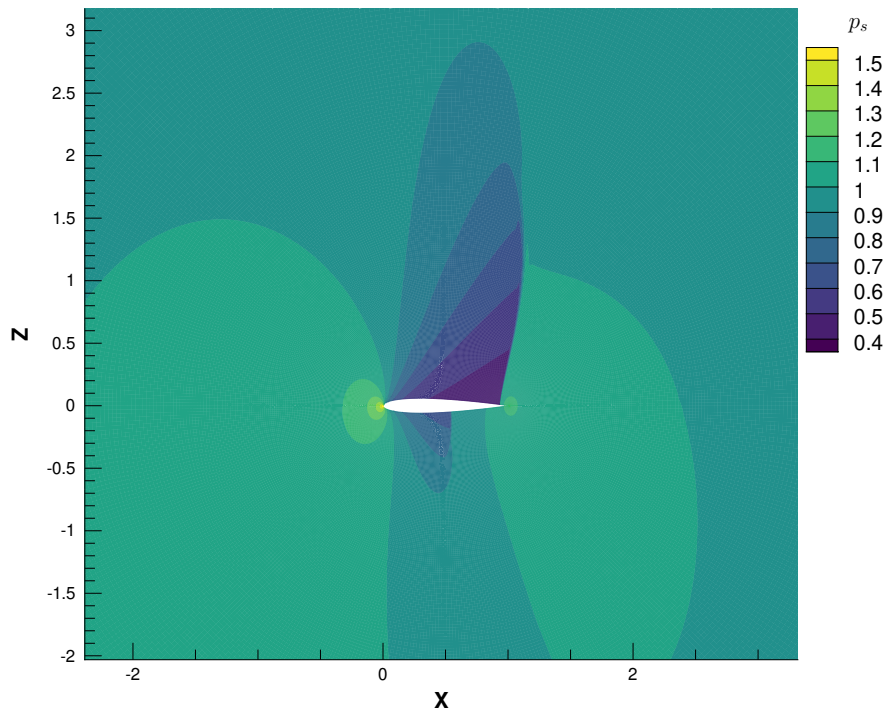
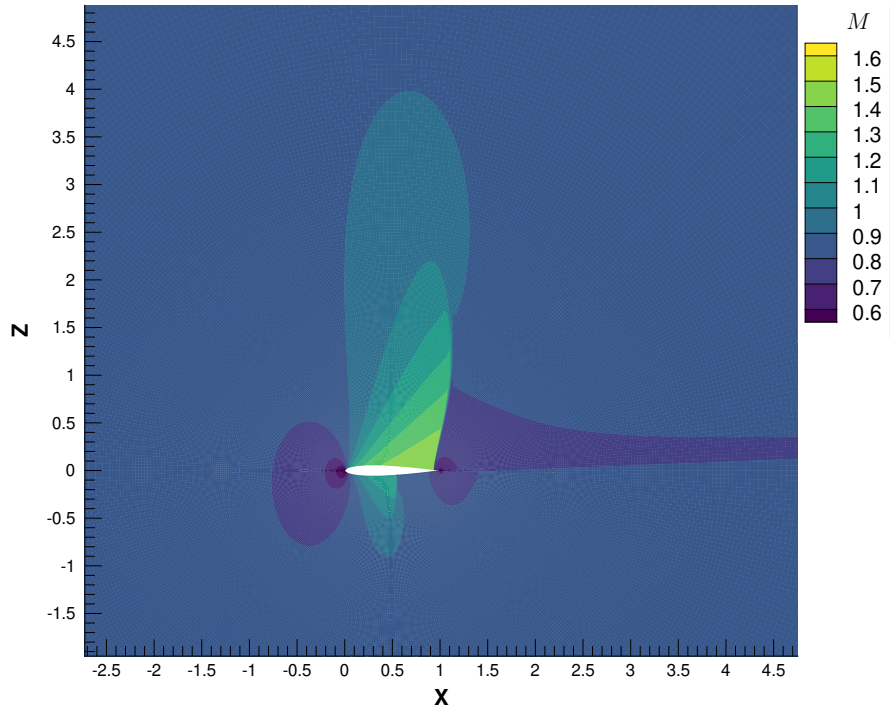


Figure 9. Champs des variables M et p_s autour du profil Γ_b pour un écoulement transsonique $M_\infty = 0.85$ sur un maillage régulier de taille $\mathcal{N} = 513$.

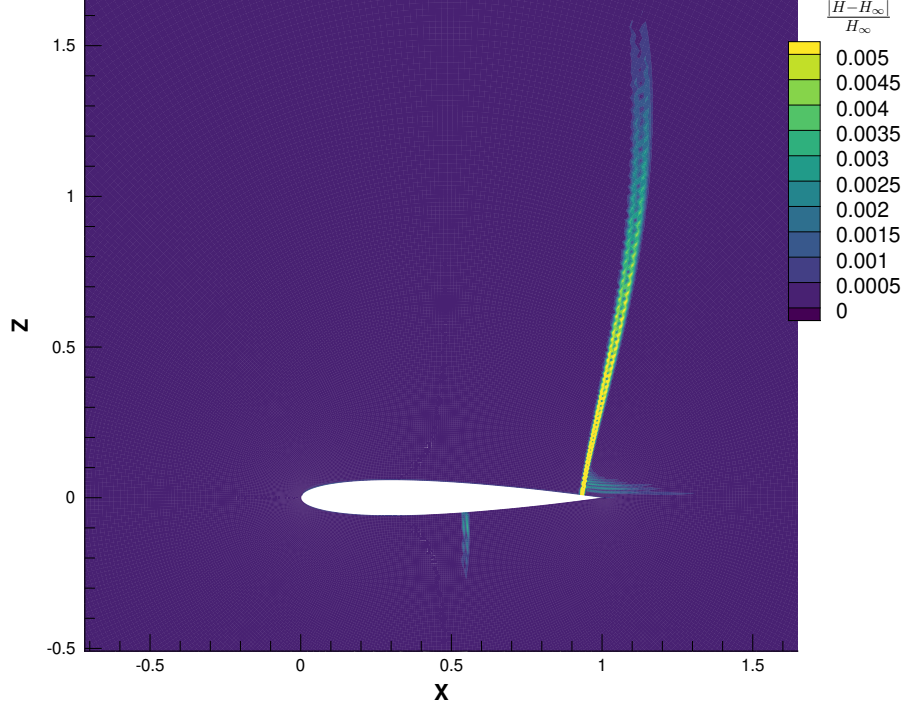


Figure 10. Champs de la variable $\frac{|H-H_\infty|}{H_\infty}$ autour du profil Γ_b pour un écoulement transsonique $M_\infty = 0.85$ sur un maillage régulier de taille $\mathcal{N} = 513$.

notre problème, contrairement aux deux autres. Il semble donc cohérent d'un point de vue de la dynamique des fluides numérique qu'on obtienne des valeurs plus élevées dans ces cas².

On aurait aimé justifier de manière plus flagrante, par le calcul et la figure 11, que les deux derniers maillages ne sont pas intéressants pour notre problème. En particulier, on aurait aimé un calcul de ε qui nous donne des valeurs plus élevées pour les maillages non-adaptés et aléatoires, quelque soit l'écoulement. Et pour palier l'anomalie, observée et discutée ci-dessus, pour l'écoulement supersonique, on a envisagé d'éliminer les cellules incluses dans les zones de choc du calcul. Pour ce faire, on a identifié les cellules concernées par le senseur de pression du schéma JST, de tel sorte qu'on puisse réécrire une nouvelle version de ε , notée $\tilde{\varepsilon}$

$$\tilde{\varepsilon} := \frac{\sum_{c_i \in \mathcal{L}} |H_{c_i} - H_\infty| \delta_{c_i} \mathcal{V}_{c_i}}{H_\infty \sum_{c_i \in \mathcal{L}} \delta_{c_i} \mathcal{V}_{c_i}} \quad (3.2)$$

avec $\delta_{c_i} = \mathbb{1}_{\{\nu_{c_i} < \alpha\}}$ avec ν_{c_i} le senseur de pression du schéma JST et α un seuil de tolérance fixé.

² On peut tout de même souligner le résultat obtenu pour l'écoulement supersonique pour les maillages de taille $\mathcal{N} = 129$, on observe que la valeur de ε est plus faible pour le maillage aléatoire que pour le maillage régulier. On peut tenter d'expliquer cette fine différence en supposant que la géométrie irrégulière du maillage aléatoire puisse l'avantager en présence de discontinuités par rapport au maillage régulier.

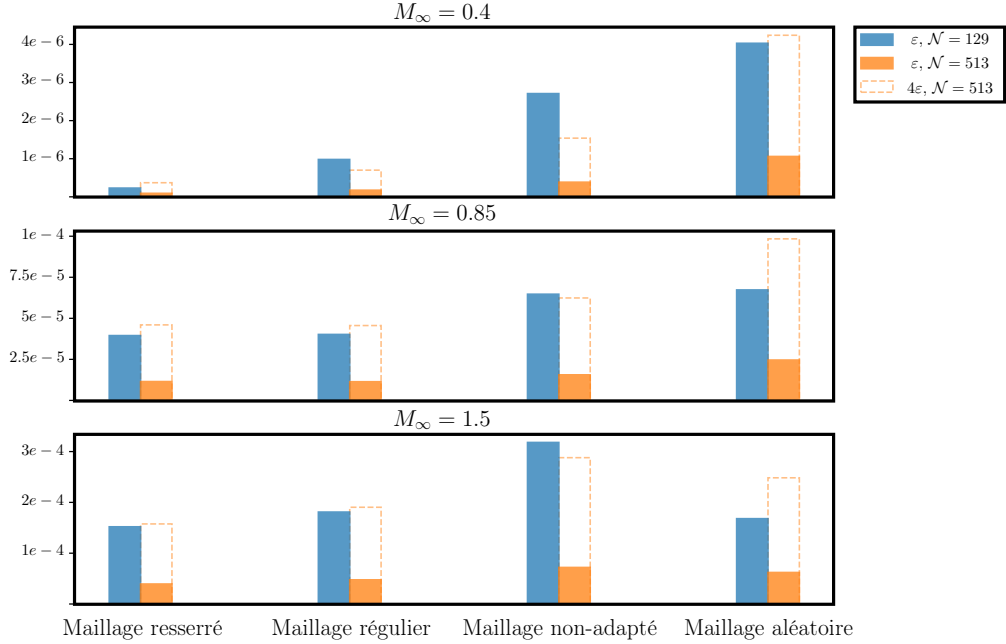


Figure 11. Calcul du critère ε pour différents maillages et différents écoulements.

Naturellement plus α est grand, moins le nombre de cellules éliminées va être important et si α est trop grand, on risque de ne pas éliminer de cellule. Pour autant, en choisissant α tel qu'il intègre parfaitement les zones de chocs, les valeurs obtenues pour $\tilde{\varepsilon}$ ne sont toujours pas strictement ordonnées dans l'ordre escompté. En fait, pour cela, il faut une valeur de α bien moindre mais dans ce cas, on supprime du calcul bien plus de cellules que celles strictement concernées par les zones de chocs, ce qui ne fait pas sens.

Cette idée a donc été abandonnée pour la suite. D'autant que si on reprend la figure 11, on constate tout de même qu'aucune anomalie n'est observée pour les maillages de taille $\mathcal{N} = 513$. Et dans tous les cas, on retrouve que, visuellement, notre critère est bien sensible à la convergence du code *elsA* puisque les valeurs pour les maillages de taille $\mathcal{N} = 513$ sont presque quatre fois plus petites que les valeurs pour les maillages $\mathcal{N} = 129$ donc notre critère est d'ordre un en espace. Pour toutes les raisons explicitées dans cette partie, on peut considérer que le critère ε peut nous permettre de hiérarchiser les solutions: plus la valeur obtenue est faible plus la solution peut être considérée comme bonne. De manière à être totalement certain de cette information et de façon à appuyer la définition de notre critère, on aimerait pouvoir effectuer notre série de calculs sur chacun des maillages présentés en utilisant une erreur dite supervisée qui elle permet réellement d'identifier si la solution obtenue peut-être considérée comme proche d'une solution plus précise obtenue sur un maillage très fin. Ainsi en comparant les variations de ε et d'une erreur supervisée ϵ^s en fonction du maillage sur lequel on a effectué le calcul, on doit pouvoir confirmer le fait que notre critère ε permet bien d'identifier une bonne solution. On introduit, pour la définition de notre erreur supervisée une solution fine notée, avec un indice f , qu'on considère comme solution de référence pour un calcul. On peut reconstituer un champ solution sur les maillages courants,

| $\mathcal{N} = 129$ | | | | |
|---------------------|-------------------|-------------------|--------------------|---------------------|
| | Maillage resserré | Maillage régulier | Maillage aléatoire | Maillage non-adapté |
| ε | $2.30e - 7$ | $9.81e - 7$ | $4.03e - 6$ | $2.71e - 6$ |
| ϵ_e | $1.60e - 6$ | $4.72e - 6$ | $4.98e - 6$ | $1.67e - 5$ |
| ϵ^s | $8.30e - 4$ | $2.82e - 3$ | $3.07e - 3$ | $7.81e - 3$ |
| $\mathcal{N} = 513$ | | | | |
| | Maillage resserré | Maillage régulier | Maillage aléatoire | Maillage non-adapté |
| ε | $9.26e - 8$ | $1.75e - 7$ | $1.06e - 6$ | $3.85e - 7$ |
| ϵ_e | $5.70e - 9$ | $2.94e - 7$ | $4.82e - 7$ | $1.19e - 6$ |
| ϵ^s | $2.53e - 5$ | $3.28e - 4$ | $3.73e - 4$ | $1.00e - 3$ |

Table 1. Calculs de ϵ , ϵ_S et $\tilde{\epsilon}$ sur différents types de maillages de taille $\mathcal{N} = 129$ et $\mathcal{N} = 513$ à $M_\infty = 0.4$.

par interpolation et définir notre erreur comme

$$\begin{aligned} \epsilon^s := & \frac{\sum_{c_i \in \mathcal{L}} |\rho_{c_i} - \rho_{c_i,f}| \mathcal{V}_{c_i}}{\rho_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}} + \frac{\sum_{c_i \in \mathcal{L}} |\rho u_{c_i} - \rho u_{c_i,f}| \mathcal{V}_{c_i}}{\rho_\infty u_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}} \\ & + \frac{\sum_{c_i \in \mathcal{L}} |\rho w_{c_i} - \rho w_{c_i,f}| \mathcal{V}_{c_i}}{\rho_\infty w_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}} + \frac{\sum_{c_i \in \mathcal{L}} |\rho E_{c_i} - \rho E_{c_i,f}| \mathcal{V}_{c_i}}{\rho_\infty E_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}}. \end{aligned}$$

Par ailleurs, on a rappelé dans la partie §3.1.1, que l'entropie est constante pour les écoulements subsoniques. On peut donc aussi introduire le critère non-supervisé suivant

$$\epsilon_e := \frac{\sum_{c_i \in \mathcal{L}} |S_{c_i} - S_\infty| \mathcal{V}_{c_i}}{S_\infty \sum_{c_i \in \mathcal{L}} \mathcal{V}_{c_i}}.$$

Afin de comparer les maillages candidats, on effectue les calculs subsoniques pour ε et ces critères. Dans notre cas, on a choisi un calcul à partir du schéma de Roe avec reconstruction MUSCL sur un maillage non-structuré avec rapport d'aspect constant et resserrement autour du profil de taille $\mathcal{N} = 2049$. On comprend bien que pour un problème Euler, le critère ϵ^s basé sur les variables d'état du système, nous permet de dire que la simulation est d'autant plus bonne que la valeur du critère obtenue est faible à convergence. On peut voir sur la table 1 que les valeurs numériques des différents critères suivent la même tendance et s'ordonnent selon les maillages tel qu'on l'espérait à travers la partie §2.3.1. Dans la table 1 les valeurs sont croissantes vers la droite, quelque soit la taille du maillage³. Le critère ε est bien corrélé au critère classique ϵ^s . On conclut donc que notre critère ε est un critère capable d'identifier une bonne solution. Afin de comprendre le véritable enjeu derrière la modification de paramètres pour le schéma en espace dans le code numérique *elsA*, on choisit, dans la suite, d'effectuer nos calculs avec le maillage classique resserré, visible sur la figure 6 à gauche. Le critère et le maillage étant définis pour les applications numériques, il faut maintenant identifier des paramètres qui pourrait être intéressant sur le schéma JST.

³ On peut faire une exception pour la valeur de ε pour les maillages de taille $\mathcal{N} = 513$. En effet, le maillage non-adapté donne une valeur plus faible que le maillage aléatoire pour une raison que l'on ne sait expliquer.

3.2 Discussion autour des paramètres influents du schéma Jameson-Schmidt-Turkel (JST).

On a pu voir deux formes du schéma JST dans la sous-section §2.2.2 et présentes dans *elsA*. C'est en fait le premier paramètre que l'on explicitera. Les autres paramètres sont directement issues de la lecture des articles [12] et [11]. On détaille dans cette partie leur mise en place dans le code. Il y a aussi des paramètres classiques du schéma JST qui sont propres à la forme de sa partie dissipative. On pense aux réels k_2 et k_4 sur lesquels on ne s'attardera pas dans cette section mais qu'on pourra intégrer dans la partie suivante.

3.2.1 Une paramétrisation du calcul de la partie convective

L'intérêt du schéma JST est qu'il est naturellement paramétrable comme mentionné ci-dessus, k_2 et k_4 sont des réels initialement choisis arbitrairement. À cela s'ajoute le fait que le schéma puisse s'écrire sous deux formes différentes, ce qui peut constituer un vrai premier point d'étude. On aimerait alors explicité un paramètre qui nous permette de trancher en faveur d'une méthode ou d'une autre et ce concrètement d'après le critère que l'on a défini ci-dessus. Dans les faits, et on pourra revenir dessus dans la section §3.3, on souhaite définir un paramètre continu et pas seulement un paramètre discret qui suivant une valeur ou une autre change de méthode de calcul. On a donc pensé à mettre en place une combinaison linéaire des flux numériques discrets de telle sorte qu'on ait

$$\mathcal{H}_c^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f) = \eta_{SkDiv} \mathcal{H}_{c,skew-symmetric}^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f) + (1 - \eta_{SkDiv}) \mathcal{H}_{c,divergence}^f(\psi_{\frac{1}{2},c}^L, \psi_{\frac{1}{2},c}^R, n_f)$$

avec η_{SkDiv} un réel compris dans $[0; 1]$. De cette façon il est naturel de définir η_{SkDiv} comme notre paramètre, puisque que l'on remarque que $\eta_{SkDiv} = 0$ donne exactement un calcul par la méthode *divergence* et $\eta_{SkDiv} = 1$ donne exactement un calcul par la méthode *skew-symmetric*. Une possibilité est donc de déterminer la valeur numérique de η_{SkDiv} telle que ε soit minimal.

3.2.2 Une paramétrisation sur la dissipation scalaire

Le deuxième paramètre, comme écrit précédemment, vient directement des auteurs de [12]. Ils spécifient que pour la partie dissipative, on peut faire le choix, pour l'équation d'énergie, d'effectuer un calcul de la dissipation numérique avec la variable ρE , soit le cas usuel, ou bien avec la variable ρH . On note tout de même que bien que l'on se soit appuyé sur le fait que H est théoriquement constante sur le domaine spatial pour définir un critère, la quantité ρH n'a, a priori, aucune raison d'être constante. Pour mettre en place une dissipation sur ρH , on reprend la définition de ρE et on y ajoute un coefficient sur p_s . De telle sorte que le terme τ_5 sur lequel on effectue la dissipation artificielle du schéma JST pour l'équation d'énergie s'écrive

$$\tau_5 = \rho E + \eta_{EH} p_s$$

avec η_{EH} un réel compris dans $[0; 1]$, de telle sorte que lorsque η_{EH} vaut 0, on ait une dissipation sur ρE et lorsque η_{EH} vaut 1, on ait une dissipation sur ρH .

3.2.3 Une paramétrisation sur le senseur de chocs

Un autre paramètre issu de [11] est lié au senseur de pression du schéma JST. Initialement le senseur pour une cellule est défini tel que

$$\nu_{c_i} := \frac{|\sum_{f \in F_{c_i}} (p_s)_{\frac{1}{2},c}^f - (p_s)_{\frac{1}{2},c}|}{\sum_{f \in F_{c_i}} (p_s)_{\frac{1}{2},c}^f + (p_s)_{\frac{1}{2},c}}$$

Comme rappelé dans [26], l'intérêt d'un tel senseur est d'éviter les oscillations d'une maille à l'autre en identifiant les cellules de choc. Ainsi, par l'écriture du schéma JST (voir §2.2.2), plus la zone est irrégulière et un choc est détecté, plus la dissipation est élevée et inversement. Cette méthode d'identification est efficace car elle réplique le calcul du gradient de pression au niveau d'une cellule. Une manière de s'en rendre compte et d'écrire l'équivalent en une dimension et de s'apercevoir que le terme ν_{c_i} est une approximation de la dérivée première à l'ordre deux par différences finies. Ainsi plus ν_{c_i} est élevé, plus la cellule c_i peut être considérée comme une cellule traversée par une discontinuité. Cependant l'article [11] propose une nouvelle méthode d'identification des chocs.

$$\tilde{\nu}_{c_i} := \frac{|\sum_{f \in F_{c_i}} (p_s)_{\frac{1}{2},c}^f - (p_s)_{\frac{1}{2},c}|}{\max\left(\sum_{f \in F_{c_i}} |(p_s)_{\frac{1}{2},c}^f - (p_s)_{\frac{1}{2},c}|, p_{lim}\right)}$$

En réalité le numérateur est commun aux deux expressions, la nuance se situe au dénominateur. En effet, si l'on considère que p_{lim} n'est autre qu'une tolérance permettant de se prémunir de la division par 0, on peut comparer les deux dénominateurs

$$\sum_{f \in F_{c_i}} |(p_s)_{\frac{1}{2},c}^f - (p_s)_{\frac{1}{2},c}| \leq \sum_{f \in F_{c_i}} (p_s)_{\frac{1}{2},c}^f + (p_s)_{\frac{1}{2},c}.$$

Ainsi $\nu_{c_i} \leq \tilde{\nu}_{c_i}$. Le second senseur va donc augmenter le nombre de cellules identifiées où appliquer une dissipation d'ordre 2. Là encore, on aimerait avoir un paramètre réel continu qui nous permette de définir le meilleur senseur de pression. On a donc défini un nouveau senseur issu de la combinaison linéaire des précédents

$$\sigma_{c_i} := \eta_\nu \tilde{\nu}_{c_i} + (1 - \eta_\nu) \nu_{c_i}$$

η_ν est un réel à valeurs dans $[0; 1]$ tel qu'en 0, $\sigma_{c_i} = \nu_{c_i}$ et en 1, $\sigma_{c_i} = \tilde{\nu}_{c_i}$. Une fois ces paramètres explicités, on propose dans la section suivante une méthodologie pour définir le choix optimal de η_{SkDiv} , η_{EH} et η_ν

3.3 Modélisation et recherche optimale des paramètres.

On souhaite dorénavant pouvoir tirer de l'information du code numérique *elsA* afin qu'à travers une minimisation de notre critère ε , on puisse trouver des valeurs cohérente du point de vue de ce dernier, pour nos paramètres η . On commence par présenter l'idée générale, puis on passera plus de temps sur la manière dont on va apprendre les données recueillies. Enfin puisqu'on cherche à minimiser la valeur prise par ε , on introduira dans une dernière sous-section une méthode d'optimisation globale.

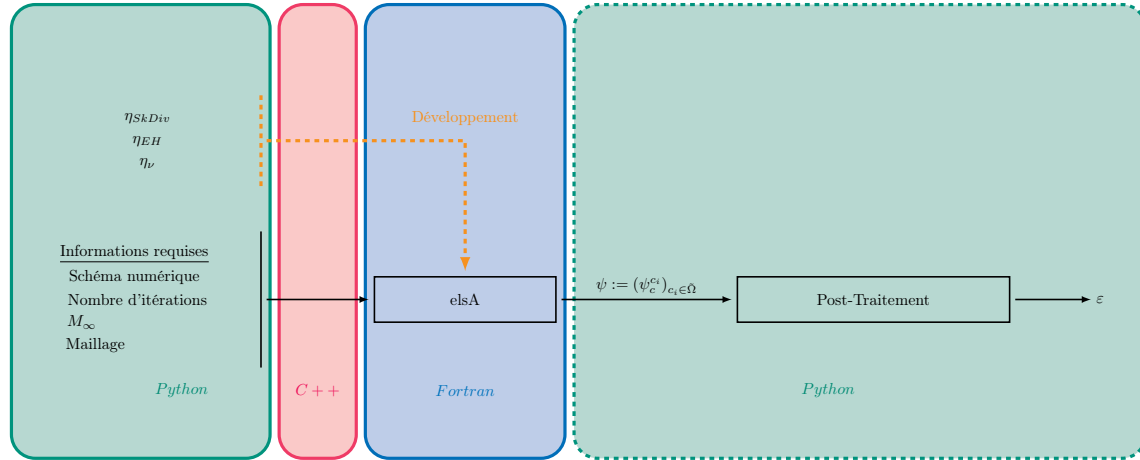


Figure 12. Graphique des ajouts réalisés dans le code *elsA*.

3.3.1 Présentation d'une démarche de recherche automatique de bons paramètres

On a fixé ci-dessus le maillage resserré pour effectuer nos simulations. La figure 12 montre les développements qui ont été ajoutés ainsi que toute la chaîne de calcul, du code *elsA* au script python permettant de calculer la valeur de ε associée à une simulation. Le script de post-traitement utilise essentiellement les modules du logiciel Cassiopée [4] pour extraire les champs nécessaires au calcul de H directement des fichiers de sorties du code numérique. On peut donc à ce stade, pour des valeurs fixées de paramètres, obtenir une valeur associée de ε . L'idée étant dans la suite de se donner un ensemble fini de couples (η_i, ε_i) . Une fois ces données acquises, il faut lancer un processus d'apprentissage de tel sorte que l'on puisse, à l'issue de cette étape, avoir une estimation de ε quelque soit les valeurs des paramètres en entrée. On crée finalement un métamodèle du processus chaîné présenté sur la figure 12 et on va ainsi estimer les paramètres qui minimisent notre critère à partir de l'information obtenue par la construction de notre métamodèle. Pour construire une estimation du processus présenté sur la figure 12, on a choisi d'utiliser une régression par processus gaussien. Ce choix est fortement lié à notre méthode d'optimisation itérative, qu'on présentera dans la sous-section §3.3.3.

On rappelle aussi qu'on cherche à améliorer notre compréhension de la simulation par l'utilisation du schéma JST et comme le montre la figure 12, une simulation numérique nécessite un nombre de Mach M_∞ en entrée. Donc la valeur de ε est lié à la nature de l'écoulement comme on l'a montré dans §3.1. Ainsi pour plusieurs écoulements, on fera le choix de faire une moyenne sur les réalisations de ε avant d'estimer notre métamodèle.

3.3.2 Régression par processus gaussien

On détaille dans cette sous-section la manière dont est réalisé la régression par processus gaussien. Pour ce faire, on s'appuie sur les travaux de *Jones et al.* [13] et de *Rasmussen et Williams* [24]. Pour la suite, on note f la fonction définie sur un espace de paramètres \mathbb{R}^D et à valeurs dans \mathbb{R} telle que $f(x) = \varepsilon$, c'est exactement la fonction décrite par la figure 12. La nécessité de créer une fonction \hat{f} , estimant f , réside dans le fait que le coût de calcul de $f(x)$ peut être relativement important, sans mentionner les difficultés numériques que peut rencontrer *elsA* pour certains points entraînant ainsi la non-convergence du code.

Pour mettre en place la régression par processus gaussien, on considère que l'on dispose initialement de

M_p points p_i et autant de valeurs de f , f_i associées. Dans notre cas, les points sont des points de notre espace paramétré inclus dans $\mathbb{R}^D: \{p_i \in \mathbb{R}^D, f_i \in \mathbb{R}\}_{1 \leq i \leq M_p}$. L'hypothèse centrale de notre démarche est de considérer que le vecteur (f_1, f_2, \dots, f_M) est une réalisation d'un processus gaussien. Cette hypothèse est forte mais peut être évacuée facilement en pratique grâce à la nature du processus gaussien. Un premier point naturel dont il faut s'assurer est que les données $(f_i)_{1 \leq i \leq M_p}$ vivent bien dans un espace image, en d'autres termes il faut s'assurer que f est bien une fonction. Par ailleurs, comme la nature gaussienne de notre processus a un effet régularisant, il faut encore s'assurer que la fonction que l'on cherche à estimer est suffisamment régulière, c'est-à-dire qu'elle est continue et n'autorise pas de pallier. Ces points seront attentivement pris en compte dans notre partie §4. Ceci étant dit la suite consiste donc à estimer un processus gaussien dont les réalisations s'ajustent parfaitement aux données dont on dispose, notre processus doit donc avoir un caractère interpolant par rapport à ces points. On résume notre hypothèse à

$$\hat{f} \sim \mathcal{GP}(m, k)$$

où m est la moyenne, et k est la fonction de variance covariance. En théorie, on peut supposer que nos points $(p_i)_{1 \leq i \leq M_p}$ sont projetés dans un espace de plus grandes dimensions dont ϕ est une base. Comme notre hypothèse initiale suggère qu'il existe une distribution gaussienne jointe entre les différents $(f_i)_{1 \leq i \leq M_p}$, on peut à priori penser qu'il existe $\beta \sim \mathcal{N}(\mathbf{0}, \Sigma_\beta)$, avec Σ_β la matrice de variance covariance de notre à priori. Finalement, on peut écrire, en utilisant notre jeu de données que $\mathbf{m} \in \mathcal{M}_{M_p, 1}(\mathbb{R})$ est le vecteur moyen des réalisations et $\mathbf{k} \in \mathcal{M}_{M_p, M_p}(\mathbb{R})$ est la matrice de covariance. Ils peuvent être définis tels que

$$\begin{aligned} m_i &:= \mathbb{E}[f_i] = \phi(p_i)^T \mathbb{E}[\beta] = 0, \quad \forall i \in \mathbb{N}_*^{M_p} \\ k_{i,j} &:= \mathbb{E}[f_i, f_j] = \phi(p_i)^T \mathbb{E}[\beta \beta^T] \phi(p_j) = \phi(p_i)^T \Sigma_\beta \phi(p_j), \quad \forall i \in \mathbb{N}_*^{M_p} \end{aligned}$$

La tendance nulle de notre processus est directement issu de notre a priori. Comme cette hypothèse pourrait être réductrice dans notre cas, on a choisi pour la suite de considérer une tendance centrale μ constante, telle que $\hat{f} \sim \mu + \mathcal{GP}(0, k)$. Cette méthode est souvent référencée en tant que *krigeage ordinaire* dans la littérature. On pourrait croire que pour définir totalement notre processus gaussien, on ait besoin d'explicitier la base ϕ ainsi que la matrice Σ_β de notre a priori. En réalité, on a aussi, par définition, que $k_{i,j} := k(p_i, p_j)$. Donc si on fixe la fonction k en amont, il n'y a plus besoin de définir précisément ϕ et Σ_β . En fait, tout notre à priori est évacué par le choix de notre fonction de variance covariance. Des choix couramment utilisés dans la littérature pour ces fonctions sont les fonctions exponentielle carrée et Matérn. Le point commun entre ces différentes fonctions et qu'elles tiennent toutes deux compte de la distance entre les deux points pour évaluer leur corrélation. C'est cohérent avec ce que l'on pourrait penser en pratique: deux points sont d'autant plus corrélés qu'ils sont proches. D'un point de vue prédictif c'est aussi cohérent. En effet, on aimerait que notre prédiction en un point dont on ne connaît pas l'image par f , soit d'autant plus proche de f_i que notre point est proche de x_i .

$$k_{SE}(r) = \exp\left(-\frac{r^2}{2\theta^2}\right) \quad (3.3)$$

La fonction de covariance exponentielle carrée est définie selon 3.3. L'utilisation de l'exponentielle assure le fait que notre fonction est différentiable à l'infini. Par conséquent, un processus gaussien qui utilise

cette fonction est très régulier et donc la fonction que l'on cherche à métamodéliser doit l'être aussi.

$$k_{\text{Matern}}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\theta} \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}r}{\theta} \right) \quad (3.4)$$

Ainsi, pour des cas réels comme ceux auxquels on risque d'être confronté dans la suite, une telle hypothèse n'est pas nécessairement adaptée. On préfère alors introduire la fonction Matérn définie selon 3.4. Parmi les informations qui les composent, on retrouve r la distance entre deux points et θ dont on parlera plus en détails dans la suite. Enfin pour la fonction Matérn, on introduit un nouveau paramètre scalaire ν dont on peut discuter de la valeur. Le livre [24] précise que les valeurs $\nu = \frac{3}{2}$ et $\nu = \frac{5}{2}$ peuvent être intéressante pour l'apprentissage. En pratique ce paramètre sera aussi estimé. On considèrera dans la suite que la fonction de variance covariance est définie par

$$k(r) = \sigma^2 k_{\text{Matern}}(r)$$

D'une façon générale, que l'on choisisse une fonction plutôt qu'une autre il y a des paramètres propres à chacune qu'il faut trouver pour définir le processus qui s'ajuste le mieux à notre échantillon initial. On distingue des paramètres communs comme l'amplitude moyenne σ et la tendance centrale μ du processus, ainsi qu'un vecteur $\theta \in \mathbb{R}^D$ de taille identique à la dimension de l'espace de paramètres que l'on étudie. Ce vecteur contient l'information sur la corrélation entre les différents paramètres. Ceci est visible dans l'expression des fonctions exponentielle carrée et Matérn. L'écart entre les valeurs de θ_i est important lorsque les paramètres sont faiblement corrélés. En fait, plus les variations du i -ème paramètre ont un faible impact sur les valeurs f_i , plus la valeur de θ_i sera grande par rapport aux autres composantes du vecteur.

Le but de la régression est finalement d'estimer ces différents paramètres à partir des données initiales de façon à garantir le caractère interpolant de notre processus. Une méthode d'optimisation est alors nécessaire pour trouver les estimations maximisant la vraisemblance. Pour établir notre procédure de définition du processus gaussien nous utilisons la librairie *python OpenTURNS* [3]. Pour l'estimation des paramètres par maximum de vraisemblance la librairie fait elle même appel à la boîte outil *C++ NLOpt*. Nous utilisons dans la suite l'algorithme d'optimisation *GN_DIRECT*. Une fois que les paramètres de la fonction de variance covariance sont établis, la valeur en un point quelconque peut être connue. En effet, notons p_* un point dont on veut connaître l'estimation donnée par \hat{f} . Notons aussi $K_* \in \mathcal{M}_{1, M_p}(\mathbb{R})$ le vecteur défini tel que $(K_*)_i := k(p_*, p_i)$, $\forall i \in \mathbb{N}_*^{M_p}$. On rappelle que $K \in \mathcal{M}_{M_p, M_p}(\mathbb{R})$ est la matrice définie à partir des points initiaux, $K_{i,j} := k(p_i, p_j)$, $\forall i, j \in \mathbb{N}_*^{M_p}$. On a alors

$$\hat{f}_* | p_*, \mathbf{p}, \mathbf{f} \sim \mathcal{N}(\mu + K_* K^{-1}(\mathbf{f} - \mu), k(p_*, p_*) - K_* K^{-1}(K_*)^T).$$

On s'autorise par la suite, les notations $\hat{y}(p_*) := \mu + K_* K^{-1}(\mathbf{f} - \mu)$ et $s^2(p_*) := k(p_*, p_*) - K_* K^{-1}(K_*)^T$. Le réel avantage de cette construction de notre estimation est que l'on dispose d'une double information. On a une estimation de f au point p par l'intermédiaire de $\hat{y}(p)$ ainsi que la variance associée à notre estimation $s^2(p)$. On note que si p est un point de notre échantillon initial donc il existe $i \in \mathbb{N}_*^{M_p}$ tel que $p = p_i$, on a alors $\hat{y}(p) = f_i$ et $s^2(p) = 0$. La bonne utilisation de ces informations constitue le base de notre méthode d'optimisation à venir.

3.3.3 Optimisation globale

À ce stade, on sait estimer la valeur de f en un point que l'on ne connaît pas, sans avoir à lancer un calcul numérique juste en évaluant la fonction \hat{f} , construite par régression. On suppose alors que l'approximation

\hat{f} est suffisamment précise et proche de f pour pouvoir considérer qu'un point de minimum pour \hat{f} est un point de minimum pour f . On peut donc chercher à minimiser \hat{f} en fonction des paramètres en entrée de façon à définir des valeurs optimales de ces derniers, ce qui serait notre objectif final. Une méthode couramment utilisée pour optimiser une réponse à partir d'une régression par processus gaussien est la méthode d'optimisation globale de Jones ([13], *Efficient Global Optimization (EGO)*).

Un point important à avoir en tête est que la méthode décrite est itérative comme beaucoup de méthode d'optimisation mais s'appuie sur un critère d'amélioration attendu de telle sorte qu'on privilégie autant l'hypothèse qu'un minimum puisse se trouver dans une zone pour laquelle on ne dispose que de peu d'informations que l'hypothèse qu'il se trouve dans une zone déjà bien identifiée. Ce critère d'amélioration attendu est défini de la façon suivante

$$EI(p) := (f_{min} - \hat{y})\Phi\left(\frac{f_{min} - \hat{y}}{s}\right) + s\phi\left(\frac{f_{min} - \hat{y}}{s}\right). \quad (3.5)$$

Par abus de notation, on note \hat{y} la valeur moyenne et s^2 la variance estimées par notre processus, défini dans la partie précédente, au point p . On note f_{min} l'estimation du minimum connu à cet instant, c'est à dire le minimum parmi les $(f_i)_{1 \leq i \leq M_p}$ de notre échantillon initial. De plus, on note Φ et ϕ respectivement la fonction de répartition et la fonction densité de probabilité d'une variable aléatoire gaussienne centrée réduite. En effet, puisque $\hat{f} \sim \mathcal{N}(\hat{y}, s^2)$ et qu'on cherche à estimer le minimum de f , on a que

$$\frac{f_{min} - \hat{y}}{s} \sim \mathcal{N}(0, 1).$$

On note que par définition de \hat{f} , la valeur de EI au point p_i telle que $f_i = f_{min}$ est nulle. De façon plus générale, il peut être nécessaire de bien distinguer les deux termes de 3.5 pour la suite. Plus notre estimation en un point p est faible plus le terme $(f_{min} - \hat{y})\Phi\left(\frac{f_{min} - \hat{y}}{s}\right)$ est grand et plus notre variance en un point p est grande plus le terme $s\phi\left(\frac{f_{min} - \hat{y}}{s}\right)$ est grand. De manière tout à fait naturelle, les zones de plus grandes variances sont les zones pour lesquelles on ne dispose que de peu d'information initialement, c'est-à-dire les zones où il y a une faible densité des $(f_i)_{1 \leq i \leq M_p}$. Ainsi on retrouve bien ce que l'on soulignait en introduction à savoir qu'on s'autorise autant à « creuser » les puits déjà identifiés, à travers le terme $(f_{min} - \hat{y})\Phi\left(\frac{f_{min} - \hat{y}}{s}\right)$, qu'à explorer les zones peu connues, à travers le terme $s\phi\left(\frac{f_{min} - \hat{y}}{s}\right)$, pour tenter d'identifier un point de minimum.

La manière dont on procède est la suivante, on cherche p maximisant l'équation 3.5. D'après ce que l'on a montré précédemment, en maximisant EI , p ne fait pas partie de notre échantillon initial presque sûrement. On lance un calcul de f au point p de manière à obtenir une valeur exacte à ce point et on injecte l'information exacte dans notre échantillon initial. Une nouvelle estimation des paramètres du processus \hat{f} par maximum de vraisemblance, comme vu dans §3.3.2, est alors nécessaire. Et on répète le processus jusqu'à ce que la valeur estimée du maximum de EI soit négligeable par rapport à la valeur estimée à l'itération précédente. Le point optimal p_{opt} alors identifié est celui de la dernière itération concluante de l'algorithme.

Cette partie nous a permis de développer théoriquement la manière dont on va procéder afin d'identifier des valeurs optimales pour nos paramètres. La partie suivante concerne la mise en place concrète de cette procédure à travers deux cas particuliers.

4 Résultats numériques sur des simulations 2D autour du profil NACA0012.

L'objectif de cette partie est de présenter les résultats obtenus pour différents espaces paramétrés définis préalablement, l'un de dimension deux et l'autre de dimension trois. Le cas d'étude choisi pour réaliser notre démarche est le même que celui évoqué dans la partie §3: il s'agit du cas profil, en deux dimensions, NACA0012. Pour chacun des espaces, on a mis en place le processus de façon à obtenir des valeurs optimales pour les paramètres η concernés. On a mentionné dans la sous-partie §3.1.2 qu'une valeur numérique pour l serait proposée au moment d'effectuer les calculs numériques. On a choisi pour ce faire $l = 4$ cordes. On rappelle que l'on désigne par corde la longueur caractéristique du profil. Donc au-delà de 4 cordes, on ne considère plus les cellules pour le calcul du critère ε .

4.1 Espace paramétré à deux dimensions.

Dans cette partie nous considérons uniquement les paramètres η_{SkDiv} et η_{EH} pour étudier leur influence sur le critère ε . Comme évoqué précédemment et pour apprendre de la simulation, on choisit de faire du multi-points à partir de deux écoulements subsoniques $M_\infty = 0.4$ et $M_\infty = 0.6$ et pour un angle d'attaque $\alpha = 2^\circ$. On revient plus en détail sur ce point dans une première sous-partie. On propose, dans la partie suivante, une validation de notre plan d'expérience et de notre métamodèle à partir de la démarche proposée dans [13]. Enfin dans une dernière partie, on rend compte des résultats obtenus. Dans la suite, pour les calculs du code *elsA*, nous avons fixé, comme critère d'arrêt, le nombre maximal d'itérations à 10000 et la tolérance sur le résidu de la variable d'état ρ à $1e - 11$.

4.1.1 Plan d'expérience et contexte

Pour construire le plan d'expérience, on a raisonné de la façon suivante. Il a tout d'abord été question de fixer le nombre de points initiaux, que l'on a choisi arbitrairement. Puis on a généré, par la méthode d'échantillonnage de l'hypercube latin d'*OpenTURNS*, 40 points dans $[0; 1]^2$. À chaque point, on va donc pouvoir associer une valeur de ε pour les écoulements $M_\infty = 0.4$ et $M_\infty = 0.6$. La valeur finale f_i que l'on va associer à ce point est une moyenne de ces deux valeurs. C'est à partir de ce plan d'expérience qu'est réalisé l'apprentissage. Une représentation des points dans l'espace paramétré est proposée sur la figure 13 et une représentation avec les valeurs de ε associées est proposée sur la figure 14. On remarque notamment que cette figure semble déjà dessiner l'allure de la surface que l'on va estimer grâce au fait que la valeur de η_{SkDiv} n'ait que peu d'influence sur les variations de ε aux points initiaux. Finalement la surface de réponse estimée par processus gaussien est présentée sur la figure 15. Une étape de validation de cette surface est proposée dans la sous-partie suivante avant d'effectuer notre recherche de points de minimum. L'optimisation des paramètres a permis de trouver les valeurs de la table 2. On remarque notamment que le paramètre θ_0 associé à η_{SkDiv} est bien plus grand que θ_1 associé à η_{EH} comme on s'y attendait d'après la remarque sur l'influence du premier paramètre sur les variations de ε . On note de plus que $\nu = 1.5$ d'après l'estimation, et cela est cohérent avec ce que l'on a pu voir dans §3.3.2.

4.1.2 Validation du métamodèle

Afin de valider notre métamodèle, on reprend la méthodologie de validation proposée dans [13] à partir des points initiaux que l'on connaît. Cette nécessité de réutiliser l'information dont on dispose provient du fait

| μ | θ_0 | θ_1 | σ | ν |
|---------------|------------|------------|---------------|-------|
| $3.5561e - 7$ | 99.97 | 5.650 | $9.9649e - 8$ | 1.5 |

Table 2. Valeurs des paramètres du processus gaussien obtenues par maximisation de la vraisemblance.

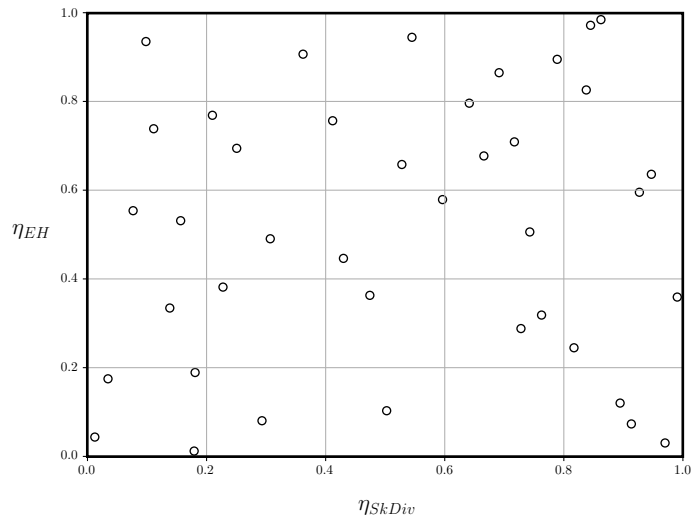


Figure 13. Plan d'expérience de 40 points dans l'espace $[0; 1]^2$.

que le coût de calcul en un nouveau point de notre fonction est conséquent. Initialement on peut penser que, pour un échantillon suffisamment grand, prédire la valeur en un point p_i uniquement à partir des 39 autres semble intéressant pour effectuer une comparaison avec la valeur réelle. Enfin comme la prédiction par l'intermédiaire d'un processus gaussien nous permet d'avoir aussi une estimation de la variance, on pourra comparer la distribution de la prédiction à partir des points initiaux complémentaires. En effet, en notant $\hat{y}_{-i}(p_i)$ et $s_{-i}(p_i)$ respectivement la prédiction moyenne et l'écart-type, au point p_i , sans la prise en compte du point p_i dans l'estimation des paramètres, alors, en théorie,

$$\xi_i = \frac{f(p_i) - \hat{y}_{-i}(p_i)}{s_{-i}(p_i)} \sim \mathcal{N}(0, 1).$$

On peut alors vérifier cela en visualisant bien le fait que les réalisations ξ_i sont bien toutes comprises dans l'intervalle $[-3, 3]$. En effet si une valeur est présente au-delà de cette intervalle, on peut considérer que la condition n'est pas satisfaite puisque l'on est dans la queue de la distribution gaussienne centrée réduite (voir Figure 17, gauche). En ordonnant les valeurs de ξ_i , on peut comparer la liste aux quantiles théoriques de la distribution gaussienne centrée réduite (voir Figure 17, droite). Enfin on peut aussi comparer la valeur estimée et la valeur attendue pour véritablement attester de la validité du métamodèle construit (voir Figure 16). Dans la pratique, on commence par cette étape. En effet, on dispose de 40 points donc

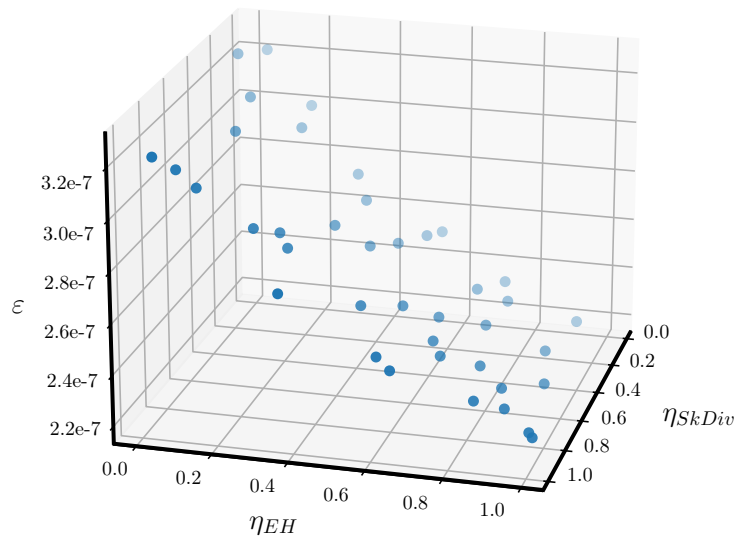


Figure 14. Plan d'expérience de 40 points dans l'espace $[0; 1]^2$ avec valeurs de ε associées.

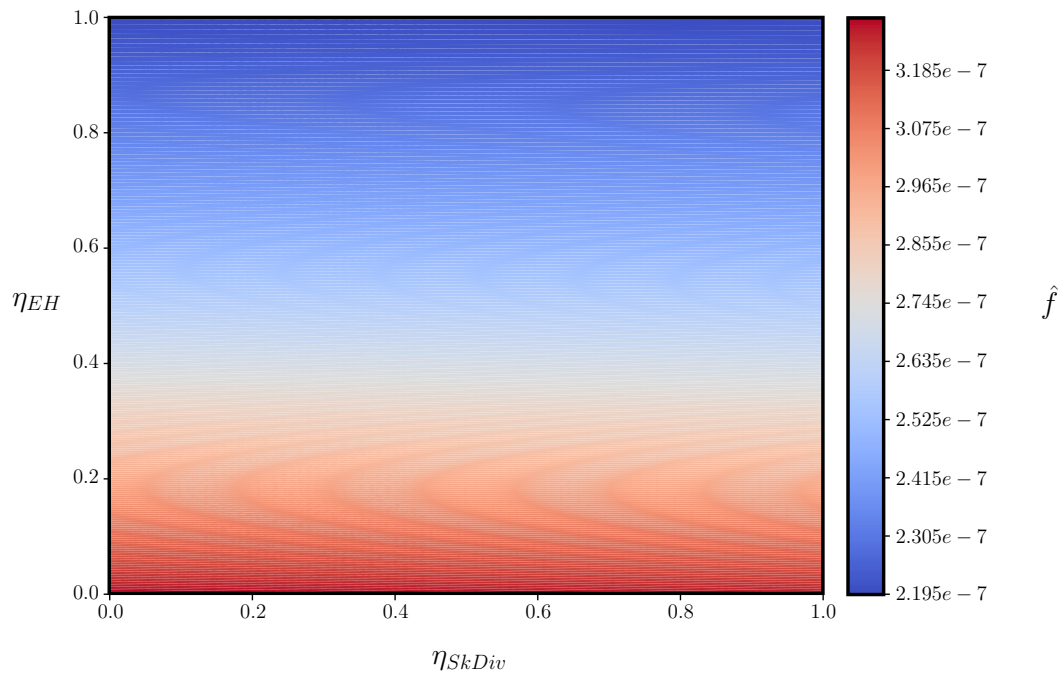


Figure 15. Surface du métamodèle estimé par régression dans l'espace $[0; 1]^2$.

| η_{SkDiv} | η_{EH} |
|----------------|-------------|
| 1.0 | 1.0 |

Table 3. Valeurs numériques des paramètres obtenues par EGO.

on peut estimer la valeur en un point à partir des 39 autres, et concrètement dans la formule de \hat{y} cela change la taille de la matrice K que l'on inverse. On retire la ligne i et la colonne i pour estimer $\hat{y}_{-i}(p_i)$ et comparer la valeur à $\hat{y}(p_i)$. Les figures 17 et 16, nous permettent de justifier, d'après l'article de Jones, l'utilisation du métamodèle ainsi construit pour chercher des valeurs numériques optimales de nos paramètres.

4.1.3 Résultats et interprétation

L'algorithme *EGO* propose une itération supplémentaire au point de minimum $(1.0, 1.0)$ et un calcul *elsA* est lancé de manière à ajouter ce point dans l'échantillon initial. Néanmoins comme la valeur d'amélioration attendue avant le calcul numérique est de l'ordre de $1e - 10$, il est normal que l'algorithme ne propose pas d'itération supplémentaire et le point de minimum identifié est bien $(1.0, 1.0)$. On place le point de minimum sur la surface estimée par le métamodèle sur la figure 18.

Les résultats obtenus confirment bien la tendance observable sur les figures 14 et 15, puisque le minimum est bien atteint lorsque $\eta_{EH} = 1.0$. Il était cependant plus difficile de visuellement estimer une valeur optimale de η_{SkDiv} du fait que ses variations influent faiblement sur f et donc aussi sur \hat{f} .

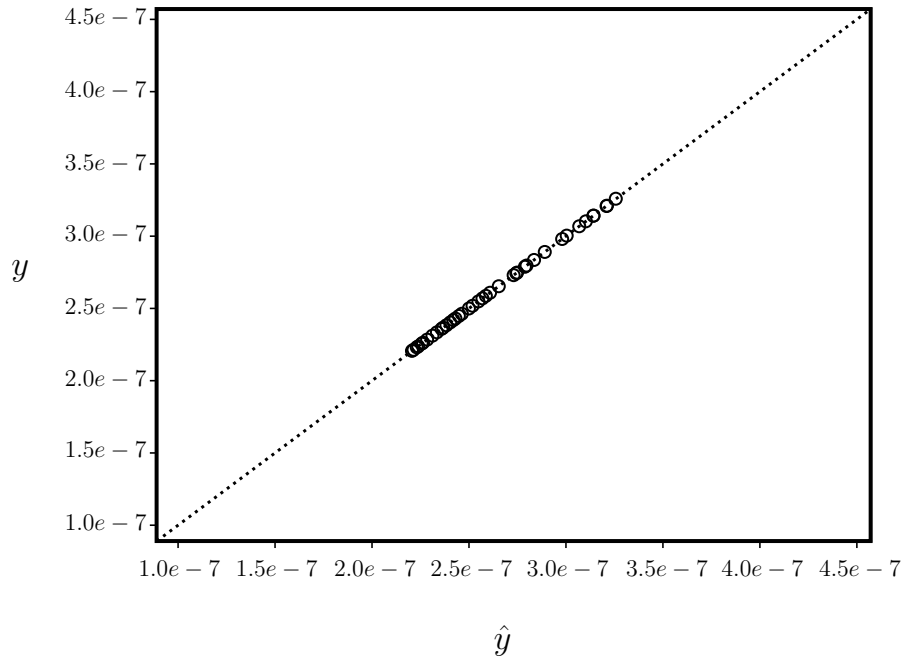


Figure 16. Distribution des valeurs prédites \hat{y} en fonction des valeurs théoriques y .

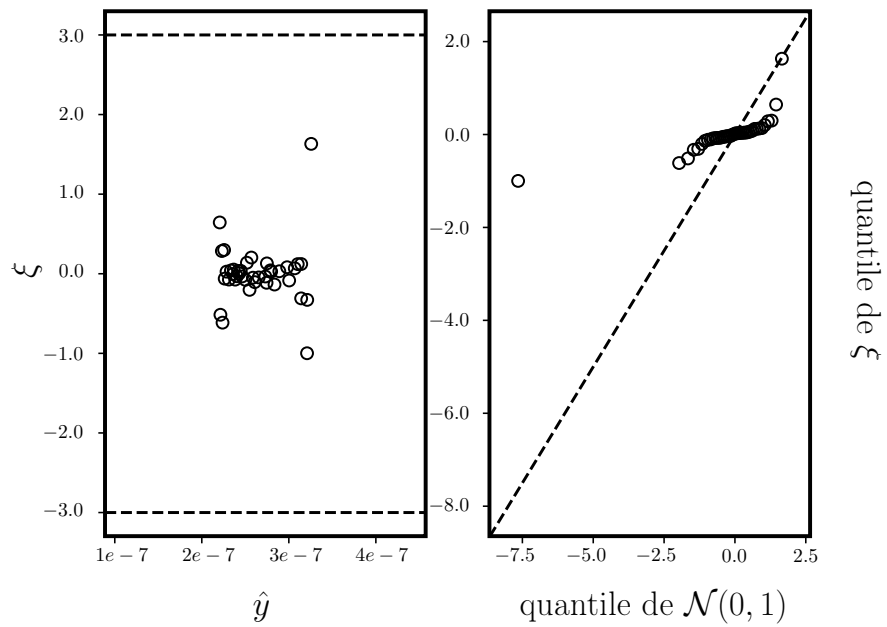


Figure 17. Distribution des valeurs prédites \hat{y} en fonction de ξ et graphe quantile-quantile pour la distribution de ξ .

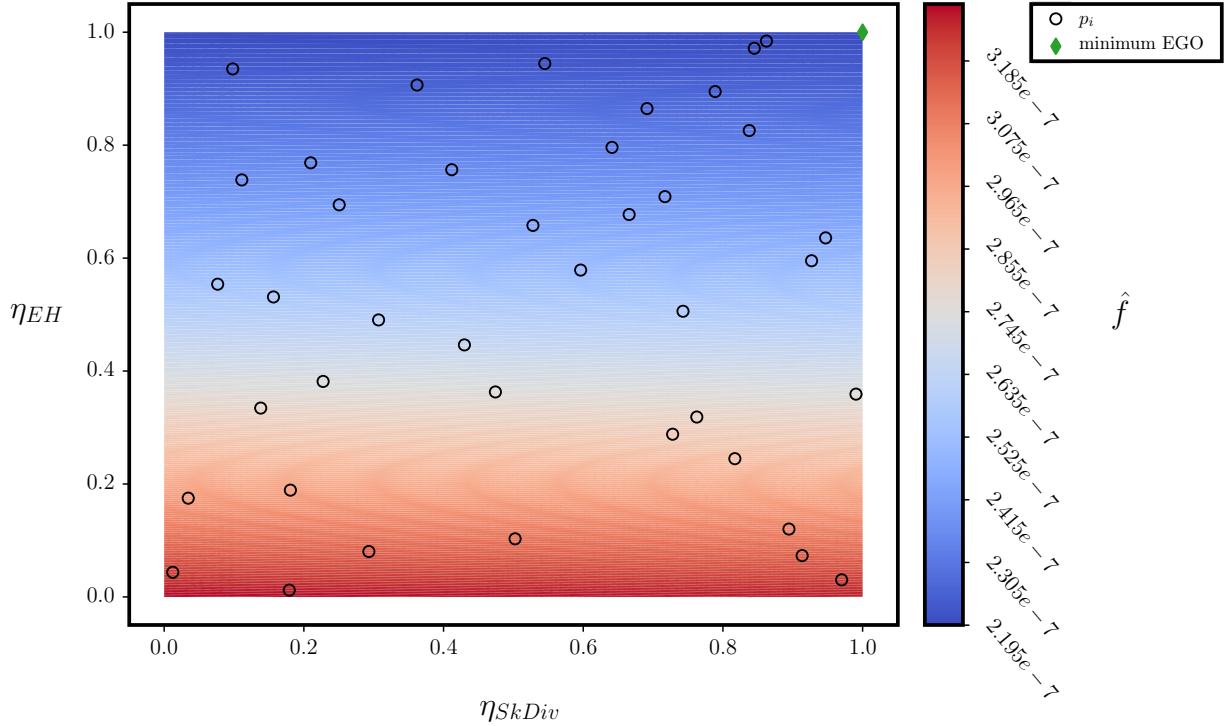


Figure 18. Surface estimée à partir de \hat{f} sur $[0; 1]^2$ et identification du point de minimum par EGO.

4.2 Espace paramétré à trois dimensions.

Dans cette partie nous considérons les paramètres η_{EH} , η_ν et k_4 pour étudier leur influence sur le critère ε . De la même façon que ce qui a été fait précédemment, on réalise notre recherche numérique à partir de plusieurs écoulements. Un écoulement subsonique $M_\infty = 0.4$ et un écoulement transsonique $M_\infty = 0.85$ et pour un angle d'attaque $\alpha = 2^\circ$. On propose toujours dans la partie suivante, comme dans la partie précédente, une validation de notre plan d'expérience et de notre métamodèle à partir de la démarche proposée dans [13]. Enfin on s'intéresse aux résultats obtenus dans la dernière partie. Dans la suite, pour les calculs du code *elsA*, nous avons fixé, comme critère d'arrêt, le nombre maximal d'itérations à 12000 et la tolérance sur le résidu de la variable d'état ρ à $1e - 4$.

4.2.1 Plan d'expérience et contexte

Préalablement et puisque l'on réalise une recherche de k_4 optimale pour nos différents écoulements, on a identifié la plage de valeurs $[0.012; 0.032]$ pour ce paramètre. On réalise la même démarche que pour l'espace de deux paramètres à savoir un échantillonnage par la méthode de l'hypercube latin d'*OpenTURNS* cette fois-ci sur l'espace $[0; 1] \times [0; 1] \times [0.012; 0.032]$. Comme initialement dans le cas subsonique, il n'y a pas de discontinuité, on fixe $k_2 = 0.0$ et pour le cas transsonique, on pose $k_2 = 0.5$. On a pu estimer notre fonction sur l'espace de définition de nos paramètres à l'aide de valeurs locales connues (voir 19) et on

| μ | θ_0 | θ_1 | θ_2 | σ | ν |
|-------------|------------|------------|------------|-------------|-------|
| $2.2754e-4$ | 55.806 | 89.106 | 3.183 | $3.5742e-3$ | 1.5 |

Table 4. Valeurs des paramètres du processus gaussien obtenues par maximisation de la vraisemblance.

a obtenu les valeurs de la table 4 pour les paramètres de notre processus gaussien. On remarque, sur la figure 19, que comme précédemment, ce sont les variations de η_{EH} qui influent le plus sur les variations de \hat{f} , puisqu'on peut distinguer que les points les plus bleus donc correspondant à des valeurs plus faibles de \hat{f} sont plus proches de $\eta_{EH} = 1.0$.

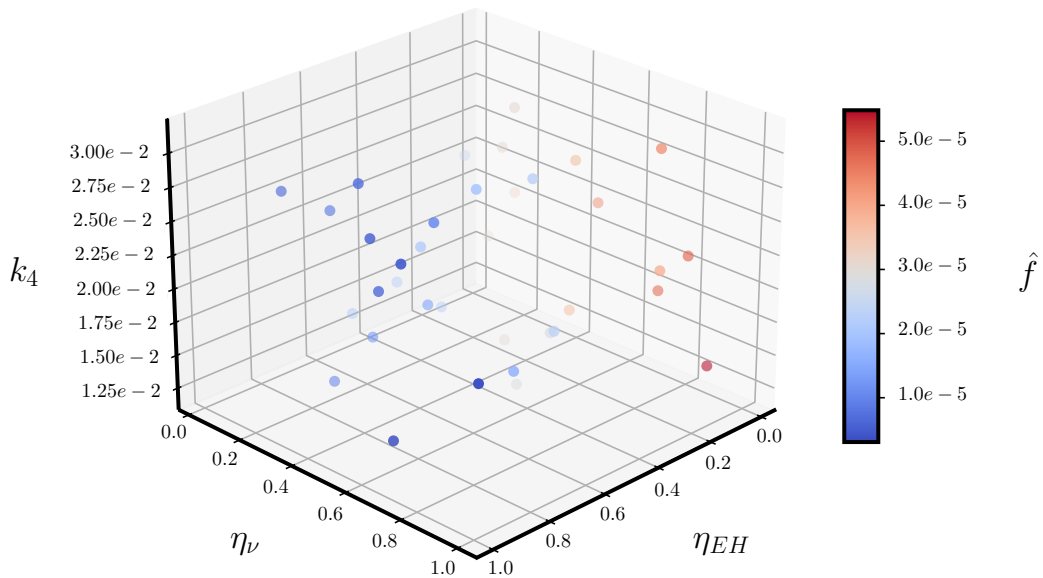


Figure 19. Plan d'expérience de 36 points dans l'espace $[0; 1] \times [0; 1] \times [0.012, 0.032]$.

4.2.2 Validation du métamodèle

Comme cela a été fait dans le cas 2D, on reprend la méthodologie proposée par Jones pour valider l'optimisation des paramètres du processus gaussien par maximisation de la vraisemblance. On obtient alors les figures 20 et 21. Par rapport au métamodèle de notre cas précédent, le métamodèle ainsi construit

| η_{EH} | η_ν | k_4 |
|-------------|------------|--------|
| 0.9859 | 0.7033 | 0.0149 |

Table 5. Valeurs numériques des paramètres obtenues par EGO.

paraît moins bon du point de vue de la méthode de validation, on a deux valeurs de ξ qui sont dans la queue de distribution de la gaussienne centrée réduite (figure 21). Néanmoins l’ajustement des valeurs prédites par rapport aux valeurs attendues est toutefois satisfaisant (figure 20) donc on décide de garder ce métamodèle pour notre optimisation avec *EGO*.

4.2.3 Résultats et interprétation

L’algorithme *EGO* a identifié le point (1.0, 0.0, 0.032) pour relancer le calcul *elsA* durant le processus d’optimisation. On note, grâce à la figure 22, que ce point se situe au bord du domaine donc, mais aussi dans une partie de l’espace des paramètres qui n’est pas bien identifiée: il y a une faible densité de points dans cette zone. Néanmoins l’algorithme n’effectue qu’une itération, ce qui est logique du fait que l’amélioration attendue (EI) en calculant la valeur de f au point identifié est de l’ordre de $1e - 6$.

On peut voir que les résultats sur la figure 22 et dans la table 5, confirment ce que nous avons observé à la création du plan d’expérience. En effet, l’algorithme *EGO* nous indique une valeur optimale de 0.9859 donc très proche de 1.0 pour η_{EH} . L’influence des autres paramètres sur \hat{f} étant moins visible, il est difficile de discuter des valeurs obtenues par notre optimisation, même si le fait d’obtenir une valeur de k_4 égale à 0.0149, donc se situant dans une tranche basse de l’intervalle sur lequel on a défini le paramètre, est rassurant. En effet, on sait que plus k_4 est grand plus le schéma JST est dissipatif donc en théorie on peut penser que notre critère ε a tendance à croître dans ce cas.

Afin de conclure sur cette partie, on peut noter que les deux recherches de valeurs optimales nous indiquent notamment qu’une dissipation sur ρH plutôt que sur ρE est plus intéressante du point de vue de notre critère ε . On pourra discuter de pistes d’améliorations pour notre démarche dans la conclusion.

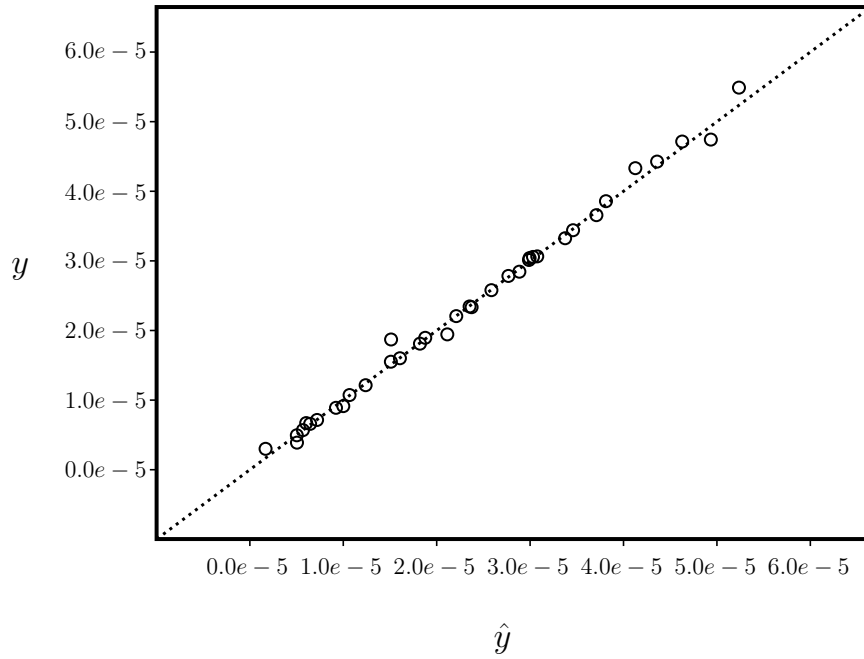


Figure 20. Distribution des valeurs prédites \hat{y} en fonction des valeurs théoriques y .

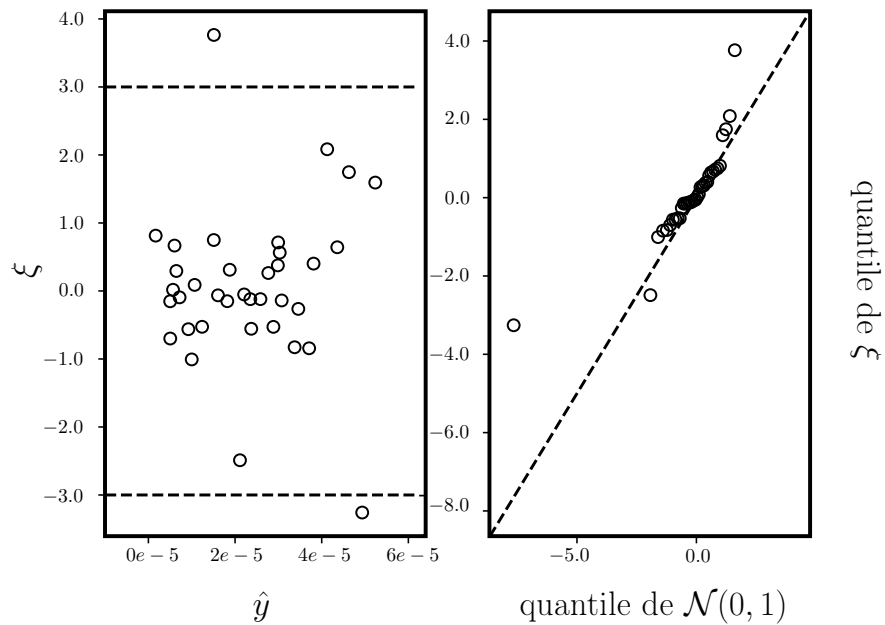


Figure 21. Distribution des valeurs prédites \hat{y} en fonction de ξ et graphe quantile-quantile pour la distribution de ξ .

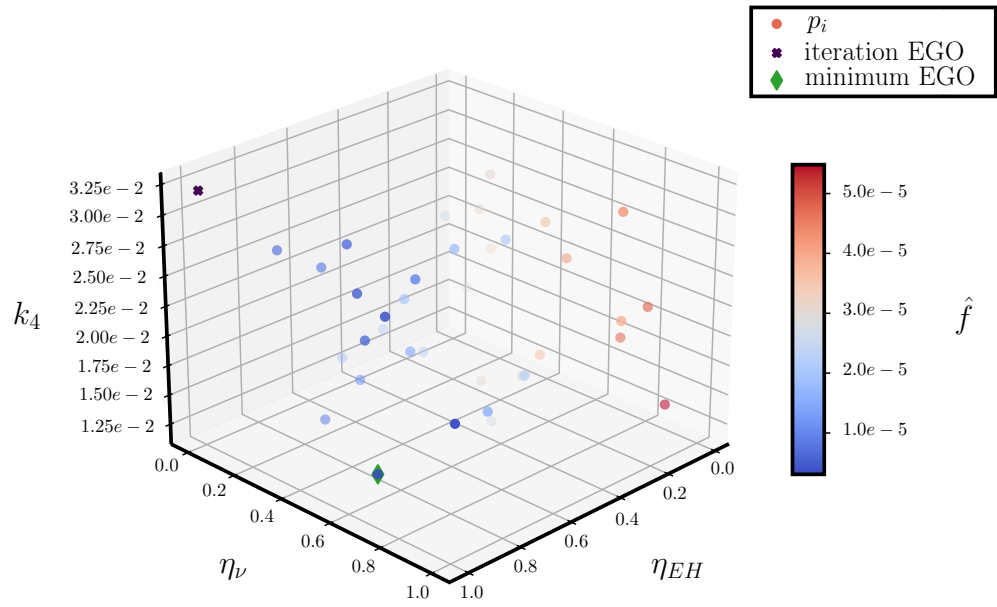


Figure 22. Identification du point de minimum par EGO sur l'espace $[0; 1] \times [0; 1] \times [0.012, 0.032]$.

Conclusion

Pour conclure, on a tout d'abord recensé des contributions récentes au domaine de l'IA pour la simulation numérique en mécanique des fluides. On a pu notamment observer et comprendre que dans un tel contexte, l'apprentissage automatique peut apporter des éléments nécessaires à une meilleure compréhension de nos simulations, et c'est dans cette perspective que l'on a initié notre démarche. Après être revenu dans un second temps sur le contexte physique et numérique du stage, la troisième partie rend compte des champs d'intérêts potentiels nécessitant le recours à un apprentissage de données pour étudier différentes configurations des équations spatiales de notre problème discret. Enfin en mettant en place concrètement notre démarche, on a pu observer des résultats intéressants.

D'un point de vue scientifique, l'intérêt du travail a pu être appuyé par l'obtention de valeurs numériques optimales pour deux espaces de paramètres différents tout en mettant en avant l'influence particulière de la variable utilisée pour la dissipation artificielle dans le schéma JST. En effet, dans les deux cas mis en place, en utilisant la variable ρH plutôt que la variable ρE pour l'équation d'énergie dans la partie dissipative, on minimise clairement l'erreur sur l'enthalpie totale. On pourrait alors imaginer une utilisation de la démarche présentée sur des plans d'expériences plus fournis, prenant en compte plus d'écoulements ou effectuer nos simulations sur des cas tests différents, de façon à essayer de généraliser les résultats obtenus.

D'un point de vue personnel, ce stage a été l'occasion pour moi de développer mes compétences en vue d'une application à la dynamiques des fluides, domaine que je connaissais peu avant cette expérience. Durant ces six mois j'ai apprécié évoluer autour de personnes dont les mathématiques appliquées, l'analyse et la simulation numérique constituent le coeur de leurs travaux de recherche. J'ai ainsi confirmé mon engouement pour les cours que j'ai pu suivre durant ma scolarité et notamment à l'école Sup Galilée. Les difficultés rencontrées durant ce stage ont été liées à mon manque de connaissances sur le domaine d'application d'une part, mais aussi est surtout à l'adaptation à l'environnement de travail. Avec quelques efforts de compréhension et l'aide reçue des personnes avec qui j'ai pu échanger, ces difficultés n'ont jamais été pénalisantes pour le déroulement du stage. J'ai donc la certitude que cette expérience me permettra d'évoluer plus sereinement à l'avenir dans le milieu professionnel.

Références

- [1] R. C. ASTER, B. BORCHERS, AND C. H. THURBER, *Parameter estimation and inverse problems*, Elsevier, 2018.
- [2] C. BARDOS, *Existence et unicéité de la solution de l'équation d'euler en dimension deux*, Journal of Mathematical Analysis and Applications, (1972).
- [3] M. BAUDIN, A. DUTFOY, B. IOOSS, AND A.-L. POPELIN, *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*, Springer International Publishing, Handbook of Uncertainty Quantification, Cham, 2016, pp. 1–38.
- [4] C. BENOIT, S. PERON, AND S. LANDIER, *Cassiopee: a CFD pre- and post-processing tool*, vol. vol. 45, Aerospace Science and Technology, 2015, pp. pp. 272–283.
- [5] L. CAMBIER, M. GAZAIX, S. HEIB, S. PLOT, M. POINOT, J. VEUILLOT, J. BOUSSUGE, AND M. MONTAGNAC, *An Overview of the Multi-Purpose elsA Flow Solver*, Aerospace Lab, 2, 2011.
- [6] N. DISCACCIATI, J. HESTHAVEN, AND D. RAY, *Controlling in high order discontinuous galerkin schemes using artificial viscosity tuned by neural networks*, Journal of Computational Physics, (2020).
- [7] E. GONCALVES DA SILVA, *Résolution numérique des équations d'Euler 1D*, Laboratoire des Écoulements Géophysiques et Industriels, Grenoble.
- [8] I. GOODFELLOW, Y. BENGIO, AND A. COURVILLE, *Deep Learning*, MIT Press, 2016.
- [9] J.-L. GUERMOND, R. PASQUETTI, AND B. POPOV, *Entropy viscosity method for nonlinear conservation laws.*, Journal of Computational Physics, (2011).
- [10] C. HIRSCH, *Numerical Computation of Internal and External Flows*, vol. Volume 2, Computational Methods for Inviscid and Viscous Flows, John Wiley and Sons Ltd., 1990.
- [11] A. JAMESON, *Origins and further development of the jameson-schmidt-turkel scheme.*, AIAA Journal, 55 (2017), pp. 1487–1510.
- [12] A. JAMESON, W. SCHMIDT, AND E. TURKEL, *Numerical solutions of the euler equations by finite volume methods using runge-kutta time-stepping schemes*, in AIAA 14th Fluid and Plasma Dynamic Conference, 1981.
- [13] D. R. JONES, M. SCHONLAU, AND W. J. WELCH, *Efficient global optimization of expensive black-box functions*, Journal of Global Optimization, (1998).
- [14] D. P. KINGMA AND J. L. BA, *Adam, a method for stochastic optimization*, in ICLR (Poster), 2015.
- [15] A. KLÖCKNER, T. WARBURTON, AND J. HESTHAVEN, *Viscous shock capturing in a time-explicit discontinuous galerkin method*, Mathematical Modelling of Natural Phenomena, (2011).
- [16] J. LACAILLE, *Statistiques descriptives*. Notes de cours, MACS3.
- [17] L. R. C. NASA, *Turbulence modeling resource, 2d naca 0012 airfoil validation case*.
- [18] M. NIELSEN, *Neural Networks and Deep Learning (online)*, Determination Press, 2015.
- [19] P. OMNES, *Volumes finis*. Notes de cours, MACS3, 2021.
- [20] E. J. PARISH AND K. DURASAMY, *A paradigm for data-driven predictive modeling using field inversion and machine learning*, Journal of Computational Physics, (2015).
- [21] P.-O. PERSSON AND J. PERAIRE, *Sub-cell shock capturing for discontinuous galerkin methods*, in In 44th AIAA Aerospace Sciences Meeting and Exhibit, 2006.
- [22] J. PETER, *Étude de schémas numériques pour la résolution des équations de Navier-Stokes instationnaires. Application à des écoulements en turbomachines.*, PhD thesis, ENSAM, 1996.
- [23] M. RAISSI, P. PERDIKARIS, AND G. KARNIADAKIS, *Pinns: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, Journal of Computational Physics, (2019).
- [24] C. E. RASMUSSEN AND C. WILLIAMS, *Gaussian Processes for Machine Learning*, MIT Press, 2006.
- [25] A. P. SINGH, S. MEDIDA, AND K. DURASAMY, *Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils*, AIAA Journal, (2017).
- [26] M. SOISMIER, *Stratégie de résolution hybride structurée et non-structurée pour la simulation d'effets technologiques en turbomachines*, PhD thesis, Communauté Université Grenoble Alpes, 2016.
- [27] J. C. VASSBERG AND A. JAMESON, *In pursuit of grid convergence for two-dimensional euler solution*, Journal of Aircraft, (2010).
- [28] P. S. VOLPIANI, M. MEYER, L. FRANCESCHINI, J. DANDOIS, F. RENAC, E. MARTIN, O. MARQUET, AND

-
- D. SIPP, *Machine learning-augmented turbulence modeling for RANS simulations of massively separated flows*, *Physical Review Fluids*, (2021).
- [29] V. ZINGAN, J.-L. GUERMOND, J. MOREL, AND B. POPOV, *Implementation of the entropy viscosity method with the discontinuous galerkin method*, *Computer Methods in Applied Mechanics and Engineering*, (2013).