Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut Supérieur de l'Aéronautique et de l'Espace

Présentée et soutenue par :
Raphaël PERRET

le mardi 7 janvier 2020

Titre :

Modélisation du champ des vitesses induit par un rotor et de ses interactions pour simuler la dynamique du vol des aéronefs à voilures tournantes

Model of the Induced Velocity of a Rotor and its Interactions to Simulate Flight Dynamics of Rotary Wings Aircraft

---

École doctorale et discipline ou spécialité :

ED AA : Dynamique des fluides et Génie mécanique, mécanique des matériaux

Unité de recherche :

Équipe d'accueil ISAE-ONERA EDyF

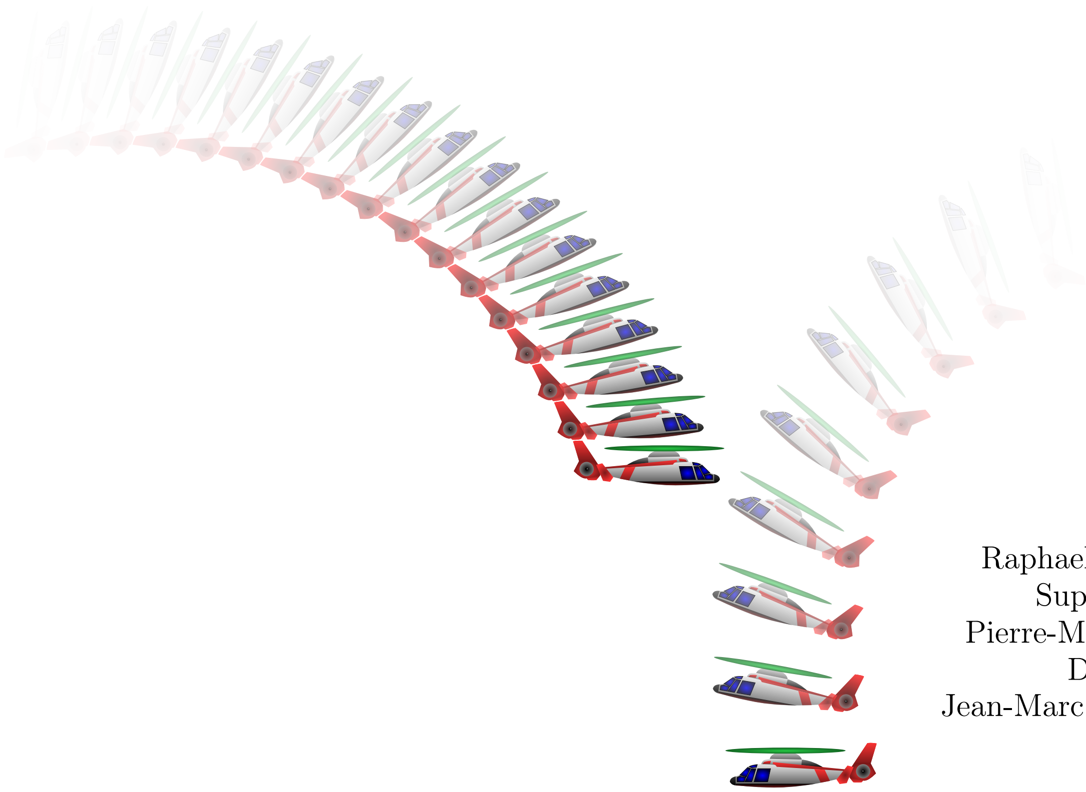
Directeur(s) de Thèse :

M. Jean-Marc MOSCHETTA (directeur de thèse)
M. Pierre-Marie BASSET (co-directeur de thèse)

Jury :

M. Thomas LEWEKE Directeur de Recherche IRPHE - Président, Rapporteur
M. Jean-Marc MOSCHETTA Professeur ISAE-SUPAERO - Directeur de thèse
M. Eric GONCALVES DA SILVA Professeur ENSMA - Rapporteur
M. Pierre-Marie BASSET Ingénieur de Recherche ONERA - Co-directeur de thèse
Mme Marilena PAVEL Professeure Assistante Delft University of Technology
M. Arnaud LE PAPE Ingénieur de Recherche ONERA

Model of the Induced Velocity of a Rotor and its Interactions to Simulate the Flight Dynamics of Rotary Wings Aircraft



Raphael PERRET
Supervised by:
Pierre-Marie Basset
Directed by:
Jean-Marc Moschetta

Contents

Remerciement

Tout d'abord il me faut remercier Pierre-Marie Basset et Jean-Marc Moschetta, sans qui cette thèse n'aurait pas eu lieu. En particulier, merci Pierre-Marie pour ta relecture assidue de ce document et tes nombreux commentaires.

Je souhaite aussi remercié Mathieu et Gemma, qui ont partagés mon bureau, ainsi que Alexandre et Adrien qui eux aussi n'ont pas manqués d'être de bonne compagnie.

Plus généralement, je tiens à remercier certaines personnes du laboratoire. Christian pour son humour et sa discussion, mais surtout sa promptitude à me *kill* lorsque ma session plante. Greg pour son humour, et particulièrement pour sa galette des rois rhum-chocolat. Nawfel pour son entrain pour les apéros et pour son bon esprit au foot. Bruno pour sa bonne humeur et pour ses conseils aiguisés de tacticien en foot. Fabrice pour ses discussions et son humour douteux. Simon pour sa formation python et malgré son humour.

Nicolas pour ses discussions à la pause café. Patrick pour sa gentillesse. Flo pour avoir été le *el fuego* le plus chaud du monde. Jonas pour sa bonne humeur sans faille. Quang Huy pour la fanfare ! Alex pour être tranquillou. Jérome pour ses discussion autour d'une bière, et sa franchise. Olivier pour son décalage par rapport au monde qui nous entoure. Kelly pour les stop tou ? Anne-Laure pour les stop tou aussi ?. Adrien parcequ'il a fait Henry IV et l'ENSC. Antonin, car sans toi il n'y aurais pas eu d'UML dans cette thèse. Floriane, tu sais pourquoi ! Quentin pour son service au bad. Mick pour sa gentillesse infinie. François et Alexis pour l'humour, les bières, la musique, et les bons moments passés ensemble.

J'aimerai aussi remercier ma famille pour leur soutient au long de ces trois années. Mon père, surtout, qui répond toujours présent lorsqu'il y a un pépin.

Je tiens aussi à remercier les membres de Mudanza, pour toutes ces répéts et weekends en musique.

Thesis overview

The first chapter is dedicated to the literature review. Because of the innovating direction taken by this thesis, the domains covered by the literature review are not limited to the dynamic inflow methods.

The second chapter presents in details the model developed by Huang and Peters in [1, 2]. From this thorough analysis results a number of remarks and possible improvements that could be added to the model. Most notably, the fact that the model is developed in the frequential domain pushed the author to develop it in the temporal domain, in order to be able to couple it with a blade element model.

Chapter 4 is composed of three main parts, all concerned with the improvements of some aspects of Huang and Peters model. The first part presents the development of the Huang and Peters model in the temporal domain, once again revealing possible improvements. Then, section 4.2 presents two improvements of the Huang and Peters model, that were implemented and tested. They show satisfying results when compared with the existing method, but several potential improvements still remain. Finally section 4.3 presents the exploration of some of the failed or abandoned paths that were tried. This section is mostly optional but is still deemed valuable as providing examples of dead ends which were explored during this research.

Chapter 5 is the heart of this thesis. A new model, inspired both by Huang and Peters model and by Lopez, Marques and Shen treatment of the incompressible Naviers-Stokes equations in [3, 4], is developed and implemented. Comparisons with other models, such as free wake, are made in order to evaluate the new model.

Nomenclature

δ_t	Non dimensional time step. ($\Delta t\Omega$, or $\Delta t\frac{R}{V_\infty}$)
β	Parameter of the remapping
χ	Wake skew angle
Γ	Streamline curves
$\hat{L}_q$	Non dimensional lift generated by blade q
Λ	Scale factor of the cylindrical domain ($\Lambda = \frac{H}{R_c}$)
λ	Total inflow velocity
λ_f	Axial free stream velocity
Λ_j^r	Test functions
λ_m	Axial induced velocity
μ	Advance ratio
ν_∞	Non dimensional free stream velocity ($\frac{V_\infty}{R\Omega}$)
Ω	Rotor rotational speed (rad/s)
$\overrightarrow{\delta\nu}$	Small variation of induced velocity (m/s)
$\overrightarrow{\gamma}$	Curvilinear direction of the streamline
$\overrightarrow{\nu}$	Non dimensional velocity. ($\frac{\overrightarrow{V}}{V_\infty}$ or $\frac{\overrightarrow{V}}{\Omega R}$)
$\overrightarrow{\xi}$	Direction of the straight streamline
$\overrightarrow{G}$	Vector potential
$\overrightarrow{n}$	Normal direction to the boundaries
$\overrightarrow{u}$	Real part of the induced velocity in the exact solution formulation

$\overrightarrow{V}$	Induced velocity (m/s)
$\overrightarrow{w}$	Imaginary part of the induced velocity in the exact solution formulation
ϕ	Velocity potential
Φ_n^m	Potential function solution of the Laplace equation in ellipsoidal coordinates
Ψ_n^m	Potential function describing the velocity
ρ	Air density
σ	Filter function
τ	Non dimensional time. ($t\Omega$, or $t\frac{R}{V_\infty}$)
τ_n^m	Coefficients describing the non dimensional pressure, also called pressure states
$\mathfrak{F}_N$	Filtered interpolation operator, up to order N
ξ	Curvilinear coordinate of the streamline
a_n^m	Coefficients describing the non dimensional induced velocity, also called velocity states
b_β	Remapping function
C_T	Thrust coefficient
D_c	Cylindrical domain
H	Height of the cylindrical domain
L_n	n^{th} Legendre polynomial
N_θ	Maximum order of the azimuthal approximation
N_r	Number of coefficients for the radial approximation
N_z	Number of coefficients for the axial approximation
P	Pressure (Pa)
p	Non dimensional pressure ($\frac{P}{\rho V_\infty^2}$ or $\frac{P}{\rho(\Omega R)^2}$)
p	Order of a filter
P_N	Interpolation operator, up to order N
P_n^m	Associated Legendre function of the first kind
Q	Number of blades of a rotor
q_β	Inverse of the remapping function
Q_N	Quadrature interpolation operator, up to order N
Q_n^m	Associated Legendre function of the second kind
R	Rotor radius

R_c	Radius of the cylindrical domain
t	Time
V_∞	Free stream velocity
V_T	$= \sqrt{\mu^2 + \lambda^2}$
V_{MFP}	Mass flow parameter
w	Weight function

1 Introduction

Résumé en français: Introduction Une courte introduction sur les vitesses induites et la mécanique du vol hélicoptère. Différentes méthodes de calcul des vitesses induites sont présentées.

1.1 Induced Velocity

The induced velocities are the flow generated by the rotor due to its action on the air. The need to model them emerges from their impact on the flight mechanics of rotary wing aircraft. Indeed the induced velocities have a strong impact on the blades. They modify the local angle of attack seen by the blades and are to be reckoned with in order to model their behaviour. In most flight conditions, they tend to add a downward wind component, which reduces the lift provided by the blades. They are therefore an essential part in various models of helicopters: for the computation of the performance of an aircraft, to the representation of its handling qualities.

Furthermore, they also have an impact on the other parts of the aircraft. An eye opening example of this impact can be seen in the pitch up effect that a helicopter experience with increasing forward velocity. This effect consists in an unintuitive behaviour of the helicopter pitch angle when the wake of the main rotor hits the rear elements of the aircraft. The wake implies a force on the horizontal tailplane which modifies the pitch of the aircraft. Without a good model of the wake of the main rotor, this behaviour is hard to represent physically.

A lot of other behaviours of rotary wing aircraft demand to represent the induced velocity. For example, modelling the off axis response was only captured recently through a better model of induced velocity in an offline simulation [5].

1.2 Various methods for modelling the induced velocites

Due to their importance, several models exist for the induced velocities generated by rotary wings. They all have specificities and various degree of fidelity. The aim here is not to be exhaustive but simply to give a better view of what exists.

The most precise models are surely obtained thanks to Computational Fluid Dynamics (CFD). This method is one of the most advanced way of modelling fluid flows, and can be applied to a a rotary wing. It can use methods such as finite element or finite volume, and attain a level of detail in the representation of the flow without equal. This underlines the fact that the space treated by the method has to be meshed, and that the performance of the method will highly depend on the mesh. It is however a time consuming, computational heavy method, which is used in specific applications that do not require any time efficiency. Examples of this method applied to rotor wake can be found in [6].

Another family of models is represented by the prescribed wake model and the free wake model. Both of those methods rely on the representation of fluid discontinuities with singularities of various strengths. The induced velocity is then computed at any point in space using the Biot-Savart induction law. The difference between prescribed and free wake lies in the fact that the free wake is auto induced, meaning that the vortices in the wake also influence each other, while the prescribed wake does not see the impact of the vortices on its wake. This makes the free wake model a good method for representing the wake of a rotary wing aircraft, while the prescribed wake model is much more efficient, but can lack of precision on some details of the wake. Both methods can be quite computational heavy since a lot of singularities are required to represent the complete wake of a helicopter. An example of a free wake model applied to the wake of a helicopter is found in [7].

The last family of models presented here is the one of interest in this thesis. They are based on a finite state formulation and answer the need for an efficient induced velocity model. The finite state formulation means that the model relies on a basis of functions to describe the flow rather than on a discretisation of space. A second peculiarity of those methods is their adaptability to the required fidelity of representation of the flow. Indeed, the basis of representation can be expanded or restrained depending on the application, and of its required fidelity. A great effort has been made during the years to improve the domain of validity of those models to a better representation of all the flight conditions of the helicopter. The main example of those methods is the He and Peters model, described in [8].

There are of course other models that have not been presented in this introduction (particle method, basic momentum theory,...) for the sake of brevity.

2

Literature Review

Résumé en français: état de l'art Dans ce chapitre, un résumé de l'état de l'art en matière de modèle de vitesses induites à nombre d'états finis est proposés. De plus, d'autres domaines sont explorés, notamment les méthodes spectrales, qui sont les fondations des méthodes à nombre d'états finis, ainsi que leurs applications aux équations hyperboliques.

2.1 Finite State Inflow model

Finite state inflow models have been around for quite a long time and are still actively developed nowadays. They aim for a real time solution of the induced flow field by a rotor. This need is due to the strong influence of the induced flow on the flight dynamics of helicopters, and was noticed, for example, by Amer [9] in 1950. This thesis paper focuses on the latest advance of the theory made by Huang, which is both the result of 50 years of work on the subject by Professor D. Peters and of recent progress on the subject.

The work of Huang [2] can be decomposed through various main events during the development of this theory.

One of the predecessor of the current model, and probably one of the most widely used around the world, is the Pitt-Peters model [10], [11]. It is a linear, unsteady model that link three inflow states (one collective and two cyclic) to the aerodynamic thrust, roll moment and pitch moment. It is developed thanks to the actuator disc theory of Loewy and Joglekar [12], and gives the following form of the perturbed inflow:

$$\lambda = \lambda_0 + r\lambda_s \sin(\psi) + r\lambda_c \cos(\psi) \tag{2.1.1}$$

Those states are linked through the following differential equation:

$$[M]\dot{\begin{pmatrix} \lambda_0 \\ \lambda_s \\ \lambda_c \end{pmatrix}} + [L]^{-1}\begin{pmatrix} \lambda_0 \\ \lambda_s \\ \lambda_c \end{pmatrix} = \begin{pmatrix} C_T \\ -C_L \\ -C_M \end{pmatrix} \tag{2.1.2}$$

where $[M]$ and $[L]$ are:

$$[M] = \begin{bmatrix} \frac{128}{75\pi} & 0 & 0 \\ 0 & \frac{16}{45\pi} & 0 \\ 0 & 0 & \frac{16}{45\pi} \end{bmatrix} \tag{2.1.3}$$

$$[L] = \frac{1}{V}\begin{bmatrix} \frac{1}{2} & 0 & -\frac{15\pi}{64}\sqrt{\frac{1-\sin(\alpha)}{1+\sin(\alpha)}} \\ 0 & \frac{4}{1+\sin(\alpha)} & 0 \\ \frac{15\pi}{64}\sqrt{\frac{1-\sin(\alpha)}{1+\sin(\alpha)}} & 0 & \frac{4\sin(\alpha)}{1+\sin(\alpha)} \end{bmatrix} \tag{2.1.4}$$

and where α is the incidence of the rotor at the disc and V is the mass flow parameter. V is defined as twice the mean axial induced velocity in hover and as $\mu = \frac{V_\infty}{\Omega R}$, the advance ratio, in forward flight. The various elements of the matrices are obtained either through derivation of the momentum or actuator disc theory, or thanks to experimental comparisons.

Although it is useful and widely used, this model is limited to a crude description of the induced flow because it only uses three states and because of its linear radial description.

In 1989, He and Peters [13], were able to include higher harmonics, and higher order radial shape functions, giving a better description of the axial inflow velocity on the disc. This model is known as the generalized dynamic wake model, or Peters-He model. The form of the pressure distribution was taken as follow:

$$P = \sum_{m=0}^{\infty} \sum_{\substack{n>m \\ n+m \text{ odd}}}^{\infty} P_n^m(\nu) Q_n^m(i\eta)(\tau_n^{m,c}\cos(\psi) + \tau_n^{m,s}\sin(\psi)) \tag{2.1.5}$$

And the axial component of the inflow was rewritten as:

$$v_z = \sum_{m=0}^{\infty} \sum_{\substack{n>m \\ n+m \text{ odd}}}^{\infty} \frac{P_n^m(\nu)}{\nu} Q_n^m(i\eta)(a_n^{m,c}\cos(\psi) + a_n^{m,s}\sin(\psi)) \tag{2.1.6}$$

where the P_n^m and the Q_n^m are the associated Legendre polynomials respectively of first and second kind, and ν, η and ψ are the ellipsoidal coordinates. In this coordinates system, the associated Legendre polynomials can form general solutions to the Laplace equation. The model can be put under a form similar to the Pitt-Peters model, that is to say as a first order differential equation coming from the momentum equation:

$$[M](\dot{a_n^{m,c}}) + V[L^c]^{-1}(a_n^{m,c}) = \frac{1}{2}(\tau_n^{m,c}) \tag{2.1.7}$$

$$[M](\dot{a_n^{m,s}}) + V[L^s]^{-1}(a_n^{m,s}) = \frac{1}{2}(\tau_n^{m,s}) \tag{2.1.8}$$

Where the apparent mass matrix $[M]$ is known in closed form, and $[L^c]$, $[L^s]$ are the influence coefficient matrices for the cosine or sine part, and only depends on the wake skew parameter $X = \tan\left(\frac{\chi}{2}\right)$, and $\chi = \frac{\pi}{2} - \alpha$.

Their model was validated by various studies [14], [15], [16] and is now used in numerous real-time simulation codes, such has FLIGHTLAB in the USA (Advanced Rotorcraft Technology), COPTER (Bell Helicopter), RCAS (Us-Army) and in Europe where it has been implemented by ONERA into the HOST code (Airbus Helicopters "Helicopter Overall Simulation Tool") [17] [18].

This model has however a few drawbacks. It only considers odd terms of the Legendre polynomials thus preventing the representation of the mass source terms (representatives of blade tip-jet rotors). Furthermore it only gives the axial inflow velocity on the disc.

The following researches thus focused on the extension of the Peters-He model out of the disc, and in 2001, Morillo [19], [20], developed a model based on a Galerkin approach of the linearized Euler's equations. He was able to derive rigorously an expression of the three components of the velocity above the rotor disc by assuming a velocity potential. He also included both odd and even terms in the representation of both the pressure and the velocity:

$$P = -\sum_{m=0}^{\infty}\sum_{n>m}^{\infty} P_n^m(\nu)Q_n^m(i\eta)(\tau_n^{m,c}\cos(\psi) + \tau_n^{m,s}\sin(\psi)) \tag{2.1.9}$$

And the induced velocity was rewritten as:

$$\vec{v} = \overrightarrow{\text{grad}}\left(\sum_{m=0}^{\infty}\sum_{n>m}^{\infty} a_n^{m,c}\Psi_n^{m,c} + a_n^{m,s}\Psi_n^{m,s}\right) \tag{2.1.10}$$

where the Ψ_n^m are the velocity potentials.

Although giving good agreement with an integral exact solution, this model had slow convergence and was not including the $m = n$ terms that are significant for the mass sources terms.

In 2006, Hsieh [21] was able to find an expression for the $m = n$ terms, and even for the $m = n = 0$ terms although it involves a singularity. This allowed Duffy and Peters [22] to later update the dynamic inflow model and give it a better convergence.

This lead to a model having good convergence for all three components of the velocity, but only above the rotor disc, the model still having some convergence issues on the disc. The induced flow below the disc is however of a crucial importance for numerous phenomena in helicopter flight dynamics (interaction, ground effect, etc.) but also for multi-rotor systems and wind turbines.

In 2012, Fei and Peters [23][24] managed to extend the model of Morillo, to compute the velocities below the disc. This was accomplished thanks to the adjoint method and derived from an integral form of the exact solution for the momentum conservation equation of the linearized Euler's equations. It mainly uses a number of spatial and temporal symmetries in the assumed form of the pressure, and the adjoint equation, which is the momentum conservation equation for which the time sign has been inverted. This method thus requires to compute the adjoint states variables but gives the velocities below the disc.

Finally in 2015, Huang and Peters [2], [1], solved the remaining problems of the theory. First the convergence on the disc was improved by using the odd terms used in the Peters-He model coupled with the even terms used by Morillo's model, giving the form of the velocity on the disc as follow, for example for the axial component of the velocity:

$$
\begin{aligned}
v_z = & \sum_{m=0}^{\infty} \sum_{\substack{n>m \\ n+m \text{ odd}}}^{\infty} \frac{P_n^m(\nu)}{\nu} Q_n^m(i\eta) \left(\alpha_{n,z}^{m,c}\cos(\psi) + \alpha_{n,z}^{m,s}\sin(\psi)\right) \\
& + \frac{\partial}{\partial z}\left(\sum_{m=0}^{\infty} \sum_{\substack{n>m \\ n+m \text{ even}}}^{\infty} a_n^{m,c}\Psi_n^{m,c} + a_n^{m,s}\Psi_n^{m,s} \right)
\end{aligned}
$$

where the $\alpha_{n,z}^m$ are called the Huang-He variables, and can be obtained with a projection of the basis of the a_n^m. Furthermore, the formula is viable for the other components of the velocity.

This velocity was then incorporated into Morillo's model through the use of a blending function to transition from the solution on the disc to the solution off the disc. Moreover, a solution for perfectly edgewise flow ($\alpha = 0°$) was obtained through the use of the adjoint method. This case corresponds to high speed flights for which the rotor wake spreads downstream in or closed to the rotor disc plane. It was then incorporated to the general solution with another blending function. The blending functions were obtained through curve fitting. This model will later be referred to as the Huang and Peters model.

2.2 Nonlinear extensions

The finite state inflow models mentioned above, that have been developed throughout 40-50 years, being by principles linear models, are not able to render some nonlinear effects. Those various effects are however of a vital importance for time simulation of manoeuvring flights. They have thus been adapted with various nonlinear extensions which will be described hereafter.

In the Peters-Huang model, a number of non linear extensions are considered. The most widely used extensions is probably the one accounting for mass flow nonlinearity, which is lost when a linear version of the Euler's equations are used. It was first developed in 1953 when Carpenter and Fridovich [25] characterised the induced velocity with a simple momentum theory. The Pitt-Peters model included a crude mass flow parameter having only two discrete values depending on the flight conditions, that had been introduced by

Ormiston and Peters in 1972 [26]. This was found to greatly improve the results of the theory when compared to experimental data.

The mass flow parameter expression was later expanded in order to cover all the range of flight conditions. It is usually incorporated to the model by multiplying the [L] matrix by it. Peters summarized the attempts to find a general mass flow parameter, and retained the most convincing in 1988 [27] which is:

$$V = \frac{\mu^2 + \lambda(\lambda + \lambda_m)}{\sqrt{\mu^2 + \lambda^2}} \tag{2.2.1}$$

where μ is the advance ratio, λ is the total inflow velocity, and λ_m is the mean axial induced velocity. All are nondimensionalized by the blade tip speed due to rotor rotation.

The Pitt-Peters model was then corrected in 1988 by Peters and HaQuang [28], by replacing the mass flow parameter by a mass flow parameter matrix, having a different value for the collective term:

$$[V] = \begin{bmatrix} V_T & 0 & 0 \\ 0 & V & 0 \\ 0 & 0 & V \end{bmatrix} \tag{2.2.2}$$

where $V_T = \sqrt{\mu^2 + \lambda^2}$. It was also added to the model by multiplying it to the $[L]$ matrix.

A similar extension was then used for all the induced flow models mentioned above, by expanding the mass flow parameter matrix's dimension with V by the number of inflow states and by multiplying it to the $[L]$ matrix:

$$[V] = \begin{bmatrix} V_T & 0 & 0 & \dots \\ 0 & V & 0 & \\ 0 & 0 & V & \\ \vdots & & & \ddots \end{bmatrix} \tag{2.2.3}$$

The definition of the mass flow parameter V has been refined, in 2000 to cover vortex ring state [29], and in 2006 to cover windmill state [30].

In 2009, Murakami and Houston [31] proposed a change in the definition of some parameters of the definition of the mass flow parameter in order to cover all cases in a unified manner.

Another nonlinear extension concerns the wake curvature. The main concern of this extension is to obtain an off axis response coherent with experimental data. In the 90s, the off axis response obtained through simulations was of the opposite sign to the experimental one. In 1995, Rosen and Isser [32] postulated that this was due to the wake curvature. Indeed, during a pitch or roll maneuver, the blade tip vortices tend to accumulate under one side of the rotor while being spread farther under the other side. This leads to a first harmonic inflow gradient, and with the blade having a 90° phase lag at a once per rev harmonic[1], it implies an hub moment in an off axis manner, as explained in [33].

In 1996, Keller [34] introduced a single factor K_R to render the effect of wake distortion on the inflow gradient, that greatly improved the off axis response. The effect of K_R were given by:

$$\frac{\lambda_c}{\lambda_0} = K_R \frac{q}{\lambda_0} \tag{2.2.4}$$

[1] This phase lag is not the same for all rotors, e.g., the R22 has a 72° phase lag.

where q is the pitch rate and $\overline{\lambda_0}$ is the average uniform inflow.

In 1997, Barocela [35] took an other more consistent definition of the parameter representing the wake distortion:

$$\frac{\lambda_c}{\lambda_0} = \kappa_c K_{R_e} \tag{2.2.5}$$

where κ_c is the curvature of the wake centerline.

In 1998 Basset and Tchen-Fo [36] used a multi-vortex-rings wake model to study and simulate wake distorsions. From the simulations, they derived analytical expressions of the distorsion factors by using neural network techniques.

In 2001, Krothapalli, Prasad and Peters [37] used a vortex tube model with a single tube of given circulation to include the effect of wake curvature in the Peters-He model. This augmentation required the modification of the L matrix to:

$$[L_K] = \left[[\widetilde{L}] + [C]K_{R_e}\right] \tag{2.2.6}$$

where $[\widetilde{L}]$ contains both the matrices $[L^c]$ and $[L^s]$, and where $[C]$ is:

$$[C] = \begin{bmatrix} [0] & [C_{pk}]^T \frac{\kappa_c}{2} & [0] & [C_{pk}]^T \frac{\kappa_c}{2} & \cdots \\ [C_{pk}]\,\kappa_c & [0] & [0] & [0] & \\ [0] & [0] & [0] & [0] & \\ [C_{pk}]\,\kappa_c & [0] & [0] & [0] & \\ \vdots & & & & \ddots \end{bmatrix} \tag{2.2.7}$$

where the $[C_{pk}]$ matrix is not known in closed form and links the inflow state variables to the circulation of the vortex tube.

However this model assumed the wake to be instantaneously influenced by the helicopter pitch or roll movement. In 2004, Zhao, Prasad and Peters [38] took into account time constants, determined with free wake results, in order to account for this effect, and added a set of first order equations to the Pitt-Peters model:

$$[\tau_D] \begin{pmatrix} \dot{X} \\ S \\ \kappa_c \\ \kappa_s \end{pmatrix} + \begin{pmatrix} X \\ S \\ \kappa_c \\ \kappa_s \end{pmatrix} = \begin{pmatrix} X \\ S \\ \kappa_c \\ \kappa_s \end{pmatrix}_{qs} \tag{2.2.8}$$

where X, S, κ_c and κ_s are respectively the tangent of half the skew angle, the wake spacing, the longitudinal and lateral wake curvatures, and qs denotes quasi-steady conditions. $[\tau_D]$ is a diagonal matrix containing the time constants obtained through vortex tube theory.

In 2016, Goulos [33], inspired by the work of Zhao [39], who used 5 vortex tubes rather than only one, improved the approach by generalising the form of the distribution of circulatory loading taken to an arbitrary number of terms, allowing a better radial description.

Huang also takes into account the wake contraction thanks to non linear extension, when the mass flow parameter is accounted for. To do so, he considers the continuity equation to compute the equivalent radius the inflow tube should have, and then map the space so that the inflow is expanded above the rotor disc and contracted below. The contraction factor he derived is expressed as:

$$K = \sqrt{\frac{V_\infty + \overline{\nu_0}}{V_\infty + \overline{\nu_1}}} \tag{2.2.9}$$

where V_∞ is the axial free-stream velocity and $\overline{\nu_0}$ and $\overline{\nu_1}$ are the integral of the axial velocity on a disc of radius 1 respectively at the rotor disc and at the wanted height (or normal distance from the disc) of calculation.

Furthermore Huang includes the effect of a nonlinear skew angle, method that was already advised by He in [29]. Indeed the skew angle used in the computation of the $[L]$ matrix as defined earlier does not take into account the induced flow. Huang therefore corrects it in the following manner:

$$\chi_{eff} = \arctan\left(\frac{\mu}{\lambda}\right) \tag{2.2.10}$$

where $\lambda = \lambda_f + \lambda_m$, with λ_f the free stream axial velocity and λ_m the mean axial induced velocity. And thus:

$$X_{eff} = \tan\left(\frac{\mu}{V_T + \lambda}\right) \tag{2.2.11}$$

where χ_{eff} is the effective wake skew angle, and X_{eff} is the parameter to be used in the $[L]$ matrix.

The Huang and Peters model, [1, 2], considers all these nonlinear extensions, in order to give a complete nonlinear inflow model for the three components of the induced velocity by the rotor in the entire flow field.

2.3 Spectral Methods and hyperbolic equations

2.3.1 Spectral methods

Taking a step back on the previously mentioned method of resolution, one can see that it is in fact part of what are known as spectral methods. The spectral methods encompass a large variety of method that deal with partial differential equations in the spectral domain, and are thus a way to discretise the equations. Among those methods are the Galerkin method, the tau method and the collocation method. A thorough review of those methods and their applications can be found in [40].

One of the main advantages of the spectral methods lies in the so-called spectral accuracy, which translates the fact that the interpolation operator of a spectral method converges faster than any power of the number of element chosen in the interpolation. This means that only a few elements in the approximation are enough to have a good description of the approximated function. This spectral accuracy is however closely linked to the regularity of the function being approximated, as was shown in [41], which is concerned with Fourier transform and Legendre and Chebyshev polynomial approximations.

In our case the method of interest is the Galerkin method, which is in fact a method that has been disregarded for a long time, but now raises interest because of its efficiency. A series of paper by Shen in 1994 and 1997, offers a great framework to treat partial differential equations in different coordinate systems [42, 43]. The main idea behind Shen's application of the Galerkin method is to choose an appropriate basis of trial function, fitted both to the equations and to the boundary conditions.

The applications of this method are numerous, but a particular case caught our attention, since it applied a Galerkin method to the incompressible Navier-Stokes equations for a confined fluid, treating both the axisymmetric [3] and non-axisymmetric cases [4], respectively in 1998 and 2002. In those papers, Lopez, Marques and Shen successfully model the behaviour of a viscous fluid confined in a cylinder with a moving wall.

2.3.2 Hyperbolic Equations

In our case, the equations with which we are concerned, once the fluid is considered non viscous and incompressible, are the incompressible Euler equations. They are non linear hyperbolic equations, for which numerous problems might hinder the use of spectral methods. In 2001, a review of the state of the art in those domains has been made in [44], where most problems encountered are addressed. Two main problems are of interest in our case. First the stability of hyperbolic equations treated by spectral methods is not guaranteed.This is sometimes called the blow-up problem of the Euler equations, and is still an active field of research (see [45, 46] in 1994 and 2019). This hints that accounting for the viscosity, as low as it is for our application might help the convergence of the equations. Another problem arises when the inputs of the algorithm are discontinuous, or when the solution of the hyperbolic equation is itself discontinuous. In this case, the problem comes from the difficulty for spectral methods to represent discontinuous or irregular functions, and is most visible with the "infamous" Gibbs phenomenon. This phenomenon has been studied extensively in the past and the mitigation of its effects is still a current field of research [47, 48, 49, 50]. There are a number of different methods that have been developed, but one of the most interesting for our application are filters.

Among the other methods for treating the Gibbs phenomenon are the mollifiers. The mollifiers are similar to the filters in their effects, but the way they are applied is quite different (see [51, 50]). Other post treatment methods exist, using Gegenbauer polynomials [47]. Close to the idea of the mollifiers, is the idea of pseudo spectral viscosity developed in [49].

2.3.2.1 Filtering

Filters have the huge advantage to be simple, and thus efficient to apply, and to mitigate the two unwanted effects of the hyperbolic equation treatment by spectral methods. In 1991, Vandeven defines a filter in [52] by the following properties:

$$\begin{cases} \sigma(0) = 1 \\ \forall l, 1 \leq l \leq p-1,\ \sigma^{(l)}(0) = 0 \\ \forall l, 1 \leq l \leq p-1,\ \sigma^{(l)}(1) = 0 \end{cases} \tag{2.3.1}$$

A filter can be applied to a spectral approximation as follows:

$$\mathfrak{F}_N(u)(x) = \sum_{n=0}^{N} \sigma\left(\frac{n}{N}\right) \hat{u_n} \phi_n(x) \tag{2.3.2}$$

Vandeven, in [52], also presents a number of different filters with various properties. The aim of the filter is mainly to attenuate the high orders of a spectral approximation in order to recover a faster convergence.

In 2008, Hesthaven and Kirby apply the filters specifically to Legendre polynomials in [53]. They show, both experimentally and theoretically, the improvements that can provide a filter, and its ability to recover a fast convergence. The behaviour of filter varies however when applied after each time-step, when trying to maintain the stability of a scheme. Kanevsky, in [54], comments on the fact that it tends to apply stronger filter than expected, because of the cumulative effect of the filter.

2.3.3 Boundary conditions

As with any set of differential equations, boundary conditions are required to guarantee the unicity of the solution.

In our case most of the boundary conditions are set by the fact that we use an open domain. A simple way to deal with this boundary is to have a size of the domain sufficiently large to assume that the effects that one wants to model are not disturbing the boundaries. This is mainly to counter the fact that vortices leaving the domain may generate back flow into the computational domain, which might make the algorithm diverge with the wrong boundary condition.

However this requires very large domain which may hinder the resolution of the spectral method used. Better alternatives are in fact still an active field of research, see [55, 56], as well as simply determining what boundary conditions to choose for incompressible Navier-Stokes equations [57, 58, 59].

However, in 2014, an interesting solution to the problem as been proposed by Dong in [60], and generalised in [61], which allows to reduce the size of the domain, by checking the value of the velocity at the open boundary condition, in order to balance the energy equation thanks to an added pressure term.

3 Huang and Peters' Model

Résumé en français: Modèle de Peters et Huang Dans ce chapitre, le modèle de Peters et Huang est décortiqué pour en fournir une analyse approfondie. L'étude porte non seulement sur la manière dont le modèle résout les équations, mais aussi sur les extensions qui lui sont apportées pour compenser certains de ces défauts.

La résolution des équations passe par une méthode spectrale, dite de Galerkin. Cette dernière fournit une approximation de la solution des équations, qui est une projection sur un sous espace fonctionnel. Le choix de la taille de ce sous espace permet de varier la précision et la performance de l'algorithme de résolution. L'application de cette méthode fournit donc un cadre idéal pour l'application à la dynamique du vol, où la performance est primordiale. La suite de ce chapitre se concentre sur les extensions non linéaires du modèle qui cherchent à compenser certains défauts ou conséquences des hypothèses prisent pour simplifier le traitement des équations d'Euler en incompressible.

La conclusion de cette étude approfondie est un certain manque de définition du domaine de validité du modèle finale. En effet, certaines hypothèses limitent le domaine de validité du modèle, et les solutions apportées pour dépasser ces limites ne sont pas toujours justifiées, ou clairement définies.

In this chapter the model developed by Huang and Peters [1] will be presented. This model inherited from a long line of various induced velocity models, all involving Peters et al. First, the various hypothesis taken by this model will be presented and commented, then its mathematical derivation. This will allow to underline in a final section the various drawbacks and advantages of this method, and it will show the way for further improvements.

3.1 Euler's equations and development

3.1.1 Assumptions for Euler's equations

One can derive the incompressible Euler equations from the Navier-Stokes equation by making the inviscid assumptions:

- Incompressibility: this hypothesis allows to decouple the system and to reduce it to 2 equations: the continuity equation and the momentum conservation equation. It is justified by the low values of the induced velocity, allowing to neglect the compressible effects.
- Inviscid fluid: this hypothesis removes the viscous terms of the equations, and is justified by the high Reynolds number of our application, mainly due to the low kinematic viscosity of air.

Those assumptions are justified in most flight conditions, but will be problematic in extreme scenarios, which are therefore excluded of this theory. This leads to the following equations between the velocity $\overrightarrow{V}$ and the pressure P:

$$\rho \frac{\partial \overrightarrow{V}}{\partial t} + \rho \overrightarrow{\text{grad}} \left(\overrightarrow{V}\right) . \overrightarrow{V} = -\overrightarrow{\nabla} P$$
$$\text{div}\left(\overrightarrow{V}\right) = 0$$

3.1.2 Further assumptions

On top of this, a few other assumptions are taken in order to simplify the Euler's equations:

- Neglecting external forces: which is justified by the low influence of the gravity on our application.
- Velocity potential: allowing to say $\exists \Phi, \overrightarrow{V} = \overrightarrow{\text{grad}}\Phi$. This assumption is justified for non rotational flows, which might be true above the rotor, but can not be justified below it.
- Linearisation: The non-linear terms of the Euler equations are linearised by assuming a variation of the velocity around the free stream velocity V_∞, such that $\overrightarrow{V} = V_\infty \overrightarrow{\xi} + \overrightarrow{\delta\nu}$, $\overrightarrow{\xi}$ being the direction of the streamline.

Those two last assumptions are much harder to justify in our case. The flow we are interested in has no reason to be non rotational below the rotor, and the impact of the linearisation will be problematic for the hover case, where V_∞ is null, and all the cases of low velocities. Nonetheless, this leads to the following form of the equations:

$$\rho\frac{\partial\overrightarrow{\delta\nu}}{\partial t}+\rho\overrightarrow{\text{grad}}\left(\overrightarrow{\delta\nu}\right).V_\infty\overrightarrow{\xi}=-\overrightarrow{\nabla}P$$
$$\text{div}\left(\overrightarrow{\delta\nu}\right)=0$$

Those equations are however easier to treat, since they are now linear, and will show good properties with the potential assumption.

3.1.2.1 Note on the non dimensional equations

There are two ways found in the literature to nondimensionalize the above equations. The first one, used by Peters and his students, consists in nondimensionalizing the time by $\frac{V_\infty}{R}$, $\tau=t\frac{V_\infty}{R}$, velocities by V_∞, $\overrightarrow{\nu}=\frac{\overrightarrow{V}}{V_\infty}$, pressure by ρV_∞^2, $P=\frac{p}{\rho V_\infty^2}$ and finally the distances by R.
The second way to do simply replaces the free stream speed V_∞ by the tip blade speed ΩR in all the cases mentioned above.
The first method gives the following form of the conservation of momentum equation:

$$\frac{\partial\overrightarrow{\nu}}{\partial\tau}-\frac{\partial\overrightarrow{\nu}}{\partial\xi}=\overrightarrow{\text{grad}}p \tag{3.1.1}$$

while the second gives the following:

$$\frac{\partial\overrightarrow{\nu}}{\partial\tau}-\nu_\infty\frac{\partial\overrightarrow{\nu}}{\partial\xi}=\overrightarrow{\text{grad}}p \tag{3.1.2}$$

The main advantages of the first method is that it gives an equation with no other parameters than the velocity and pressure, while the second method introduces a remaining parameter, $\nu_\infty=\frac{V_\infty}{\Omega R}$. However it avoids the issue of dividing by V_∞, which might be a problem in hover cases, while the rotor will always be considered to rotate or to have certain radius.
We will however follow the Peters nondimensionalising convention for the remainder of this chapter which presents the Huang and Peters model as it is.

3.1.2.2 Note on the link between acceleration potential and velocity potential

Following results of [62], we develop the following line of reasoning. One can compute the rotational of the acceleration as follows, by applying the rotational operator to the momentum conservation equation:

$$\overrightarrow{\mathrm{rot}}\left(\frac{d\overrightarrow{V}}{dt}\right) = \rho\left[\frac{d}{dt}\left(\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho}\right) - \overrightarrow{\overrightarrow{\mathrm{grad}}}\overrightarrow{V}.\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho}\right] \tag{3.1.3}$$

$$= \frac{\overrightarrow{\mathrm{grad}}\rho \wedge \overrightarrow{\mathrm{grad}}p}{\rho^2} + \overrightarrow{\mathrm{rot}}\overrightarrow{F} \tag{3.1.4}$$

This shows that, for the rotational of the acceleration to be null, two conditions should be met:

- The forces applied to the system should derive from a potential (which is the case of the gravitational force)
- The fluid should be incompressible ($\overrightarrow{\mathrm{grad}}\rho = 0$) or at least barotropic (which implies $\overrightarrow{\mathrm{grad}}\rho \wedge \overrightarrow{\mathrm{grad}}p = 0$)

If both conditions are met, then $\overrightarrow{\mathrm{rot}}\left(\frac{d\overrightarrow{V}}{dt}\right) = 0$. This means that one can assume an acceleration potential in the domain. In turn one can find that:

$$\frac{d}{dt}\left(\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho}\right) = \overrightarrow{\overrightarrow{\mathrm{grad}}}\overrightarrow{V}.\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho} \tag{3.1.5}$$

And therefore:

$$\frac{d^{n+1}}{dt^{n+}}\left(\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho}\right) = \overrightarrow{\overrightarrow{\mathrm{grad}}}\frac{d^n\overrightarrow{V}}{dt^n}.\frac{\overrightarrow{\mathrm{rot}}\overrightarrow{V}}{\rho} \tag{3.1.6}$$

Under those conditions, if the rotational of the velocity is null at a given time, then the flow is non rotational in a continuous domain.
This gives in turn that the rotational of the velocity is always null only by assuming that an instant t exists where the rotational is null, on top of the two previous assumptions.
In this case, the acceleration potential assumption implies an irrotational flow which therefore implies the existence of a velocity potential for a simply connected domain[a].
This shows that the only assumptions required for a velocity potential in a continuous domain, are an acceleration potential and an instant t with no rotational. This can in turn be translated by:
In a continuous domain,

$$\begin{cases} \text{The fluid is barotropic} \\ \text{The forces derive from a potential} \\ \exists t, \left(\overrightarrow{\mathrm{rot}}\overrightarrow{V}\right)_t = \overrightarrow{0} \end{cases} \implies \exists \phi, \overrightarrow{V} = \overrightarrow{\mathrm{grad}}\phi \tag{3.1.7}$$

Therefore, in the case of a simply connected domain, there is little advantage to assume an acceleration potential rather than a velocity potential.

[a] To be simlpy connected is a property of a topological space. A topological space is simply connected if it is path-connected and if any loop can be continuously contracted to a point [63].

3.1.3 Link to the Laplace equation

Computing the divergence of the momentum equation, and accounting for the velocity potential, one obtains, above the rotor:

$$\Delta p = 0 \tag{3.1.8}$$

Indeed, the first term of acceleration can be expressed as follow:

$$\operatorname{div}\left(\frac{\partial \overrightarrow{V}}{\partial t}\right) = \frac{\partial}{\partial t}\operatorname{div}\left(\overrightarrow{V}\right) = 0 \tag{3.1.9}$$

And the second term can be simplified by using the continuity equation:

$$\operatorname{div}\left(\overrightarrow{\operatorname{grad}}\left(\overrightarrow{\delta\nu}\right).\overrightarrow{\xi}\right) = \sin(\chi)\frac{\partial}{\partial x}\left[\operatorname{div}\left(\delta\nu\right)\right] - \cos(\chi)\frac{\partial}{\partial z}\left[\operatorname{div}\left(\delta\nu\right)\right] = 0 \tag{3.1.10}$$

This gives that the pressure is a potential function[1], since it satisfies the Laplace equation. Furthermore the continuity equation, combined with the potential velocity assumption, implies that:

$$\Delta\phi = 0 \tag{3.1.11}$$

Meaning that the velocity potential is also a potential function. We thus need to find an adequate method to solve the Laplace equation, under the influence of an actuator disc.

3.2 Developing a Galerkin method for the model

This section will present how the Euler equations are solved. First we present the philosophy behind the treatment of equations by the Galerkin method, then we present the ellipsoidal coordinates system, and finally how Peters et al. apply the Galerkin method to this case.

3.2.1 Galerkin's method

The Galerkin's method is a way to discretise a set of partial differential equations. For this it uses a set of functions $(\lambda_i)_{i\in\mathbb{N}}$, preferably orthogonal to each other for a function scalar product, i.e.

$\forall i, j \in \mathbb{N}$:

$$\langle\lambda_i, \lambda_j\rangle = \int_{Xa}^{Xb} \lambda_i(X)\lambda_j(X)w(X)dX = \delta_{i,j} \tag{3.2.1}$$

where $\delta_{i,j}$ is the Kronecker notation and w the weight function associated to the $(\lambda_i)_{i\in\mathbb{N}}$. This set will be called the trial or test functions. The normalised Legendre polynomials and the normalised Chebyshev polynomials form examples of such orthogonal families with $w = 1$ and $w = (1-x^2)^{-\frac{1}{2}}$ respectively.

Thus, the solution u of a partial differential equation, such that: $L(u) = f$, can be approximated on the sub vector space defined by the $(\lambda_i)_{i\in\mathbb{N}}$ so that:

[1]A potential function is by definition a function respecting the Laplace equation, while the potential assumption means that a vector derive from a potential, i.e. $\overrightarrow{f} = \overrightarrow{\operatorname{grad}}(\phi)$

$$u = u_{app} + R \tag{3.2.2}$$

where:

$$u_{app} = \sum_{i=0}^{\infty} a_i(t)\lambda_i(x) \tag{3.2.3}$$

The approximation of the solution is thus made, short a residual R, that is by definition orthogonal to the vector space defined by the $(\lambda_i)_{i\in\mathbb{N}}$.

Injecting the approximation of u in its partial differential equation, it remains to express the scalar product $\langle \lambda_j, L(u_{app}) - f\rangle = 0$ function of the $a_i(t)$ in order to discretise the system, and finally inverse it to find the values of the $a_i(t)$.

By truncating the basis of the $(\lambda_i)_{i\in\mathbb{N}}$ to a finite set of N functions, one can obtain a $N * N$ system and find the coefficients of the approximate u_{app}. With a regular enough u and well chosen trial functions this will give a good approximation of u.

The main challenge of this method thus resides in the expression of $\langle \lambda_j, L(u_{app}) - f\rangle$ for all the λ_j in the truncated basis. In the case of a linear operator, the task can be done, but even such linear operation as derivation will imply a loss of accuracy, depending on the trial functions used, and non linear operator tend to introduce numerous terms out of the truncated basis selected. A study of the loss of accuracy due to the various operations is done in section 5.5.1.

More generally, the Galerkin method is in fact part of a larger category of method called spectral methods. Those methods rely on the discretization of functions through the use of well chosen function basis. The Galerkin is the special case where the projection and the approximation spaces are the same, i.e. the velocity is approximated on the same basis it is projected on. Furthermore, other spectral methods rely on quadrature to evaluate the scalar products they encounters, while the Galerkin method turns to a more efficient method by relying on analytically evaluating those same scalar product.

3.2.1.1 Note on the accuracy of the Galerkin's method

The main advantages of the Galerkin method lies in its accuracy. If we consider the functional space $L^2(I)$ (see definition in appendix C), the Legendre and Chebyshev set (and in fact any orthonormal polynomial set with increasing degree) are complete in $L^2(I)$ [41].
This means that if one considers such a set to approximate u, where $u \in L^2(I)$, then R is null. Furthermore, if one considers the Sobolev spaces of Hilbert type (see appendix C again), one can show the spectral accuracy of the Legendre and Chebyshev Galerkin methods [41]. The spectral accuracy underlines the fact that the truncated series for the approximation of u converge extremely fast (faster than any polynomial power), as long as u is regular enough.

Figures 3.2.1 and 3.2.2 presents respectively Galerkin approximations of a continuous and of a discontinuous function, with 10, 20 and 30 terms. This allows to see the dependency of the accuracy of the approximation to the regularity of the approximated function. Indeed the smooth function is satisfyingly represented with 10 terms, while the discontinuous function is not even close to the step function with 30 terms. One can also observe the apparition of spurious oscillations, called the Gibbs phenomenon.

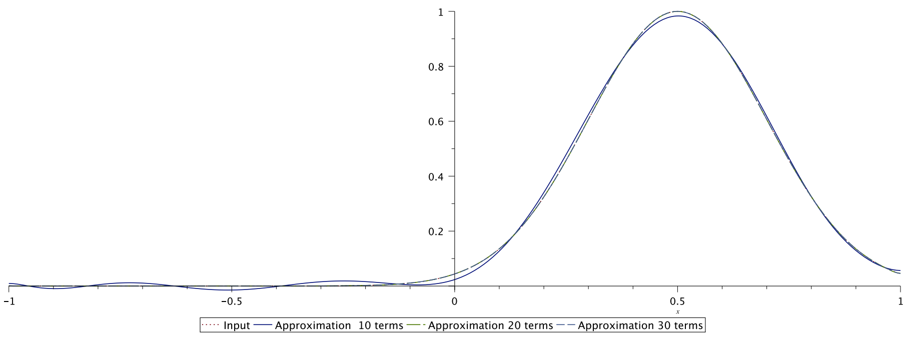


Figure 3.2.1: Approximation of a Gaussian function with 10, 20 and 30 terms

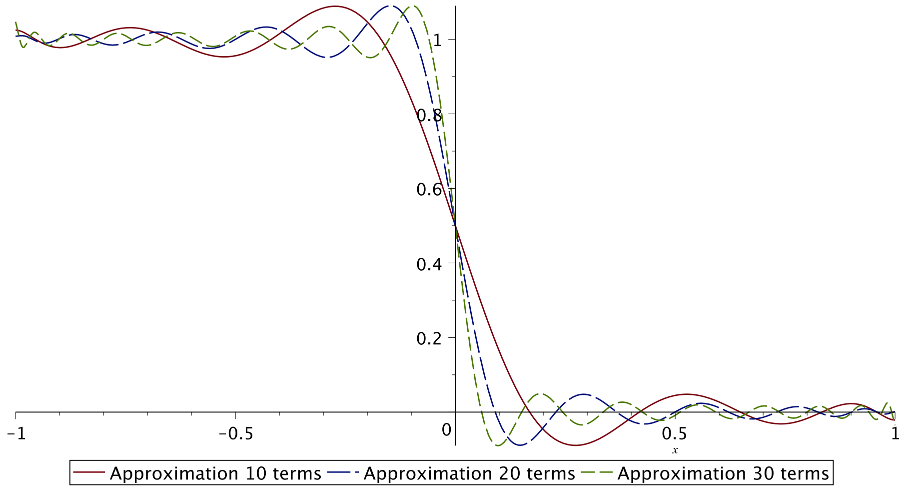


Figure 3.2.2: Approximation of a step function with 10, 20 and 30 terms

3.2.2 The ellipsoidal coordinates system

The choice of this coordinate system to treat the equations is due to several factors. Firstly, it is highly linked to the vision of the rotor taken. In the case of Huang and Peters's model, the rotor is seen as an actuator disc, i.e. an infinitely thin disc immersed in the domain, acting on the fluid by creating a pressure discontinuity across its two sides. Representing this discontinuity might be difficult, and other implementations of an actuator disc exist and will be commented in section 5.4. It so happens that the ellipsoidal coordinates are discontinuous on a disc at the centre of the domain which fits our need nicely. It is indeed possible, thanks to the coordinate discontinuity, to impose a jump in pressure across the disc.

The transform from the cartesian coordinates to the ellipsoidal coordinates is made through the following relationships:

$$\begin{cases} x = & -\sqrt{1-\nu^2}\sqrt{1+\eta^2}\cos(\psi) \\ y = & \sqrt{1-\nu^2}\sqrt{1+\eta^2}\sin(\psi) \\ z = & -\eta\nu \end{cases} \tag{3.2.4}$$

Fig. 3.2.3 gives a representation of the ellipsoidal coordinates. The red lines represent variations of ν (which forms ellipses), the green lines the variations of ψ and the blue lines the variations of η (which forms hyperbolas).

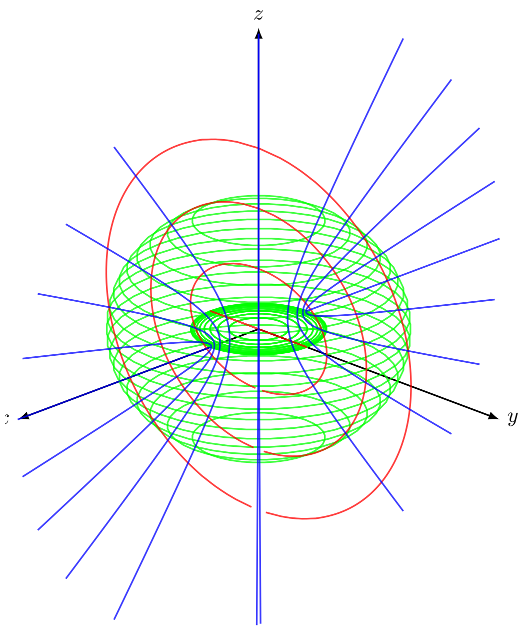


Figure 3.2.3: Representation of the ellipsoidal coordinates

Furthermore the solution of the Laplace equation can be nicely expressed using associated Legendre functions and trigonometric polynomials in ellipsoidal coordinates, that we will now present.

3.2.3 The Laplace equation in ellipsoidal coordinates

In the ellipsoidal coordinates system, the Laplace equation can be put under the following form:

$$\frac{\partial}{\partial \nu}\left[(1-\nu^2)\frac{\partial \Phi}{\partial \nu}\right] + \frac{\partial}{\partial \eta}\left[(1+\eta^2)\frac{\partial \Phi}{\partial \eta}\right] + \frac{\partial}{\partial \psi}\left[\frac{\nu^2+\eta^2}{(1+\eta^2)(1-\nu^2)}\frac{\partial \Phi}{\partial \psi}\right] = 0 \tag{3.2.5}$$

If we suppose that the space variables can be separated in the pressure potential function Φ as follows:

$$\Phi(\nu,\eta,\psi) = \Phi_1(\nu)\Phi_2(\eta)\Phi_3(\psi) \tag{3.2.6}$$

It gives:

$$\left(\frac{1}{\Phi_1}\frac{\partial}{\partial \nu}\left[(1-\nu^2)\frac{\partial \Phi_1}{\partial \nu}\right] + \frac{1}{\Phi_2}\frac{\partial}{\partial \eta}\left[(1+\eta^2)\frac{\partial \Phi_2}{\partial \eta}\right]\right)\frac{(1+\eta^2)(1-\nu^2)}{\nu^2+\eta^2} = -\frac{1}{\Phi_3}\frac{\partial^2 \Phi_3}{\partial \psi^2} \tag{3.2.7}$$

Which gives the following system, with M a separation constant:

$$\frac{1}{\Phi_1}\frac{\partial}{\partial \nu}\left[(1-\nu^2)\frac{\partial \Phi_1}{\partial \nu}\right] + \frac{1}{\Phi_2}\frac{\partial}{\partial \eta}\left[(1+\eta^2)\frac{\partial \Phi_2}{\partial \eta}\right] = M\frac{\nu^2+\eta^2}{(1+\eta^2)(1-\nu^2)} \tag{3.2.8}$$

$$\frac{\partial^2 \Phi_3}{\partial \psi^2} + M\Phi_3 = 0 \tag{3.2.9}$$

By noting that $\frac{(1+\eta^2)(1-\nu^2)}{\nu^2+\eta^2} = \frac{1}{1-\nu^2} - \frac{1}{1-\eta^2}$, one can once again separate the equation, to obtain:

$$\frac{\partial}{\partial\nu}\left[(1-\nu^2)\frac{\partial\Phi_1}{\partial\nu}\right] + \left(\frac{-M}{1-\nu^2} + N\right)\Phi_1 = 0 \tag{3.2.10}$$

$$\frac{\partial}{\partial\eta}\left[(1+\eta^2)\frac{\partial\Phi_2}{\partial\eta}\right] + \left(\frac{M}{1+\eta^2} - N\right)\Phi_2 = 0 \tag{3.2.11}$$

$$\frac{\partial^2\Phi_3}{\partial\psi^2} + M\Phi_3 = 0 \tag{3.2.12}$$

If one takes $M = m^2$ and $N = n(n+1)$, it follows that:

$$\frac{\partial}{\partial\nu}\left[(1-\nu^2)\frac{\partial\Phi_1}{\partial\nu}\right] + \left(\frac{-m^2}{1-\nu^2} + n(n+1)\right)\Phi_1 = 0 \tag{3.2.13}$$

$$\frac{\partial}{\partial\eta}\left[(1+\eta^2)\frac{\partial\Phi_2}{\partial\eta}\right] + \left(\frac{m^2}{1+\eta^2} - n(n+1)\right)\Phi_2 = 0 \tag{3.2.14}$$

$$\frac{\partial^2\Phi_3}{\partial\psi^2} + m^2\Phi_3 = 0 \tag{3.2.15}$$

The solution to the above equations in Φ_1 and Φ_2 are the associated Legendre function of first and second kind respectively, P_n^m and Q_n^m, that are defined in appendix A.1. The solution of the last equation are the trigonometric functions $\cos(m\psi)$ and $\sin(m\psi)$.

This gives that a potential function can be written as a sum of the following terms:

$$\Phi_n^m(\nu,\eta,\psi) = P_n^m(\nu)Q_n^m(i\eta)\cos(m\psi) + P_n^m(\nu)Q_n^m(i\eta)\sin(m\psi) \tag{3.2.16}$$

However, the boundary conditions give some restrictions on the possible values for m and n, since the pressure potential can not have infinite values. This implies $m, n \in \mathbb{N}$ and $n \geq m$, which are the only values for which the associated Legendre functions do not diverge.

3.2.4 Application of the method

We now have a form of our velocity and pressure potential in a coordinate system allowing us to implement easily the presence of a rotor. In the present case, the Galerkin method seems particularly adapted. The functions to be described are regular, the pressure discontinuity being included in the coordinates, and the Φ_n^m, solutions of the Laplace's equation in ellipsoidal coordinates, make good candidates as trial functions. They are not however an orthogonal basis of $L^2(D)$ (see appendix C) in ellipsoidal coordinates, and the author could not find any research on their completeness in $L^2(D)$. However, the fact they do have some orthogonality properties, see [19], and that they are solution to the Laplace equation justify their use[2].

For the sake of conciseness the following symbol is introduced to designate the double sum:

$$\underset{m,n}{\overset{M,N}{+}} = \sum_{m=0}^{M}\sum_{n=m}^{N} \tag{3.2.17}$$

[2] The basis could be orthonormalised, using a Gram-Schmidt process to truly apply a Galerkin method. However, since a simple change of basis allows to pass from one form to the other, the principles of the method remain the same.

Using the potential assumption, one can express the velocity as follows:

$$\overrightarrow{\nu} = \mathop{\Large +}\limits_{m,n}^{\infty,\infty} a_n^{m,c}(t)\overrightarrow{\mathrm{grad}}\Psi_n^{m,c} + a_n^{m,s}(t)\overrightarrow{\mathrm{grad}}\Psi_n^{m,s} \tag{3.2.18}$$

While the pressure is expressed as follows:

$$p = \mathop{\Large +}\limits_{m,n}^{\infty,\infty} \tau_n^{m,c}(t)\Phi_n^{m,c} + \tau_n^{m,s}(t)\Phi_n^{m,s} \tag{3.2.19}$$

The Φ_n^m are the solutions of the Laplace equation in ellipsoidal coordinates. The Ψ_n^m can be chosen differently in a first time, for they need to satisfy the velocity boundary conditions, and might differ from the pressure form. The time dependency of the coefficients is dealt with through the use of the frequency domain. We will assume that it depends on a single frequency ω in the remainder of the chapter, i.e. $a_n^m(t) = a_n^m e^{i\omega t}$. We also drop the dependency with time in the notations for conciseness.

Injecting the expression of the velocity and of the pressure in the momentum conservation equation, one only needs to choose a basis for the projection space. Using a Galerkin method should imply the choice of projection basis, being here the Φ_n^m, but nothing would prevent the use of some other trial functions.

We thus multiply the equations by the gradient of the trial functions and integrate over the domain D. D is here the domain of validity of the velocity potential assumption, and consists of the space situated above the disc, but represented through ellipsoidal coordinate. It is presented on Fig.3.2.4.

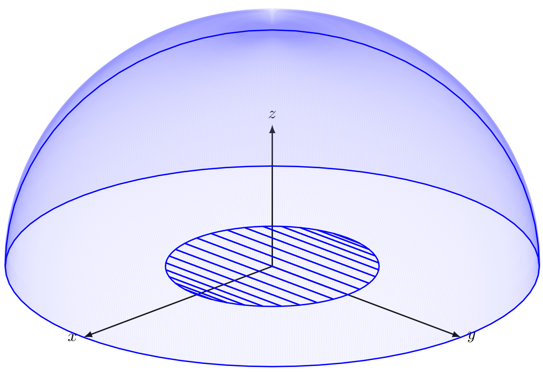


Figure 3.2.4: Representation of the integration domain, with a hashed rotor disc

The momentum equation thus become, $\forall r, j \in \mathbb{N}, r > j$:

$$\begin{aligned}\mathop{\Large +}\limits_{m,n}^{\infty,\infty} \iiint\limits_D \frac{\partial a_n^m}{\partial t}\overrightarrow{\mathrm{grad}}\Psi_n^m\overrightarrow{\mathrm{grad}}\Phi_r^j + a_n^m\overrightarrow{\mathrm{grad}}(\overrightarrow{\mathrm{grad}}\Psi_n^m.\overrightarrow{\xi})\overrightarrow{\mathrm{grad}}\Phi_r^j dD \\ = \mathop{\Large +}\limits_{m,n}^{\infty,\infty} \iiint\limits_D \tau_n^m\overrightarrow{\mathrm{grad}}\Phi_n^m\overrightarrow{\mathrm{grad}}\Phi_r^j dD\end{aligned}$$

One can then apply a particular form of the divergence theorem reminded here:

$$\iiint\limits_D \overrightarrow{\mathrm{grad}}(f).\overrightarrow{\mathrm{grad}}(g)dD = \iint\limits_S g\overrightarrow{\mathrm{grad}}(f).\overrightarrow{n}\,dS - \iiint\limits_D g\Delta(f)dD \tag{3.2.20}$$

$$= \iint\limits_S f\overrightarrow{\mathrm{grad}}(g).\overrightarrow{n}\,dS - \iiint\limits_D f\Delta(g)dD \tag{3.2.21}$$

where $\overrightarrow{n}$ is the normal to the surface of integration S, which is the boundary of the domain D.

In the present case the functions f and g are the Φ_n^m and the Ψ_n^m. Remembering that the velocity potential also respects the Laplace equation, the second term of the two right hand side equations are null. It follows that the momentum equation can be put under both of the following forms:

$$\mathop{+}\limits_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t} \iint\limits_S \Psi_n^m \overrightarrow{\mathrm{grad}}\Phi_r^j.\overrightarrow{n}\,dS + a_n^m \iint\limits_S \overrightarrow{\mathrm{grad}}\Psi_n^m.\overrightarrow{\xi}\,\overrightarrow{\mathrm{grad}}\Phi_r^j.\overrightarrow{n}\,dS$$

$$= \mathop{+}\limits_{m,n}^{\infty,\infty} \tau_n^m \iint\limits_S \Phi_n^m \overrightarrow{\mathrm{grad}}\Phi_r^j.\overrightarrow{n}\,dS$$

or:

$$\mathop{+}\limits_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t} \iint\limits_S \overrightarrow{\mathrm{grad}}(\Psi_n^m).\overrightarrow{n}\,\Phi_r^j dS + a_n^m \iint\limits_S \overrightarrow{\mathrm{grad}}(\overrightarrow{\mathrm{grad}}\Psi_n^m.\overrightarrow{\xi}).\overrightarrow{n}\,\Phi_r^j dS$$

$$= \mathop{+}\limits_{m,n}^{\infty,\infty} \tau_n^m \iint\limits_S \overrightarrow{\mathrm{grad}}\Phi_n^m.\overrightarrow{n}\,\Phi_r^j dS$$

One can show that the previous integrals can be reduced to integrals over the rotor disc plane only, see [19]. Furthermore one can notice that, on the rotor disc plane, noted S_D:

$$\overrightarrow{\mathrm{grad}}(\Phi_n^m).\overrightarrow{n} = \frac{\partial \Phi_n^m}{\partial z} \tag{3.2.22}$$

And, one can furthermore take the notation:

$$\overrightarrow{\mathrm{grad}}(\Phi_n^m).\overrightarrow{\xi} = \frac{\partial \Phi_n^m}{\partial \xi} \tag{3.2.23}$$

Which gives the following forms of the momentum equation:

$$\mathop{+}\limits_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t} \iint\limits_{S_D} \Psi_n^m \frac{\partial \Phi_r^j}{\partial z} dS + a_n^m \iint\limits_{S_D} \frac{\partial \Psi_n^m}{\partial \xi}\frac{\partial \Phi_r^j}{\partial z} dS = \mathop{+}\limits_{m,n}^{\infty,\infty} \tau_n^m \iint\limits_{S_D} \Phi_n^m \frac{\partial \Phi_r^j}{\partial z} dS \tag{3.2.24}$$

or:

$$\mathop{+}\limits_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t} \iint\limits_{S_D} \frac{\partial \Psi_n^m}{\partial z} \Phi_r^j dS + a_n^m \iint\limits_{S_D} \frac{\partial^2 \Psi_n^m}{\partial z \partial \xi} \Phi_r^j dS = \mathop{+}\limits_{m,n}^{\infty,\infty} \tau_n^m \iint\limits_{S_D} \frac{\partial \Phi_n^m}{\partial z} \Phi_r^j dS \tag{3.2.25}$$

The integrals over the rotor disc can also be seen as a scalar product, which gives if we set:

$$\langle f, g\rangle = \iint\limits_{S_D} fgdS \tag{3.2.26}$$

$$\sum_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t}\langle \Psi_n^m, \frac{\partial \Phi_r^j}{\partial z}\rangle + a_n^m\langle \frac{\partial \Psi_n^m}{\partial \xi}, \frac{\partial \Phi_r^j}{\partial z}\rangle = \sum_{m,n}^{\infty,\infty} \tau_n^m\langle \Phi_n^m, \frac{\partial \Phi_r^j}{\partial z}\rangle \tag{3.2.27}$$

or:

$$\sum_{m,n}^{\infty,\infty} \frac{\partial a_n^m}{\partial t}\langle \frac{\partial \Psi_n^m}{\partial z}, \Phi_r^j\rangle + a_n^m\langle \frac{\partial^2 \Psi_n^m}{\partial \xi \partial z}, \Phi_r^j\rangle = \sum_{m,n}^{\infty,\infty} \tau_n^m\langle \frac{\partial \Phi_n^m}{\partial z}, \Phi_r^j\rangle \tag{3.2.28}$$

Limiting the size of the approximation to a given number of terms, we obtain a discretised problem, that can be put under the following matrix form:

$$[M]\left\{\frac{\partial a_n^m}{\partial t}\right\} + [C]\left\{a_n^m\right\} = [D]\left\{\tau_n^m\right\} \tag{3.2.29}$$

The terms of the matrices are therefore the scalar products of the various terms of the equations. The way the approximation is truncated is done following the table method that is explained in [29, 19]. It requires two values: m_{odd} which defines the number of odd terms and m_{even} for the even terms. The author could not find any justification for this method of truncation in the literature, but the way it is made resembles a rough filtering of the higher order terms in order to maintain the stability of the algorithm.

3.2.5 Expression of the velocity potential

In order to solve the previous matrix equation it remains to express the velocity potential in a suitable form.

3.2.5.1 Choice of the form of potential

In the method presented above, the matrices depend on the form chosen for the velocity potential decomposition i.e. on the Ψ_n^m. The choice of this potential is motivated by the fact that it is adapted to the problem at hands. We here describe the reasoning for its choice.

In the literature [19, 13], a common form of the velocity potential is taken as:

$$\Psi_n^m = \int_{-\infty}^{\xi_0} \Phi_n^m d\xi \tag{3.2.30}$$

where the integral follows a straight rectilinear streamline, skewed by the wake skew angle χ, described as follows:

$$\begin{cases} x = x_0 + \xi\sin(\chi) \\ y = y_0 \\ z = -\xi\cos(\chi) \end{cases} \tag{3.2.31}$$

This form can be justified by a series of reasons:

- It satisfies the condition that the potential is 0 far upstream
- It has integrable derivatives, which allows to express the velocity nicely
- It seems a suitable candidate if we consider the form given to the pressure

It does however have a few drawbacks:

- It does not have a closed form expression
- It therefore requires a change of variable

The application of the Galerkin method would require the velocity to be expressed with the test functions Φ_n^m. Indeed, this ensures a good conditioning of the matrices, since the Φ_n^m have good orthogonality properties. This would remove a part of the problem due to the previous form but it renders all the integrals containing even terms in $(n+m)$ divergent.

This form however has its own advantages:

- Satisfy the 0 condition far upstream
- Closed form expression of all components with $(n+m)$ odd
- No change of variable required

A perfect solution for the expression of the velocity would ideally meet the following criteria:

- Satisfy the 0 condition far upstream
- Closed form expression of all components (integrable derivatives)
- No change of variable required (thus no matrix inversion)
- Have a gradient that is not singular (to express the velocity easily)
- Give matrices that are well-conditioned (or at least with a slow evolution of the condition number with higher orders)

It was however hard to find any form meeting all the requirement cited. Therefore, this seems to justify the form proposed in the Morillo and Duffy model [19, 13]. However one of the ideas behind the Huang and Peters model [1] is precisely to change the form of the velocity potential depending on where their strengths and weaknesses are. This underlines the versatility of the spectral method, which allows to choose many forms of the approximation space, but also allows to choose the test functions.

Having chosen a velocity potential, one only needs to compute the matrices and solve for the states in order to obtain the induced velocities above the rotor.

3.2.5.2 Link between the various forms of potential

The chosen form of the velocity potential allows to express the required scalar products easily but is a hindrance when one tries to compute the values of the velocity, for it requires to perform an integral. It will therefore be replaced by another form of the potential, that allows to compute more easily the velocity values. A link can be found between any two kinds of expression of the potential, since they both describe the same velocity:

$$\overrightarrow{v} = \sum_{m,n}^{\infty,\infty} a_n^m \overrightarrow{\mathrm{grad}} \Psi_{a,n}^m = \sum_{m,n}^{\infty,\infty} b_n^m \overrightarrow{\mathrm{grad}} \Psi_{b,n}^m \tag{3.2.32}$$

Thus by applying the Galerkin method to the previous equation, one can obtain the following expression linking the (a_n^m) and the (b_n^m):

$$[A]\{a_n^m\} = [B]\{b_n^m\} \tag{3.2.33}$$

where:

$$[A] = \left[\iint\limits_S \Psi_{a,n}^m \Phi_r^j dS\right] \tag{3.2.34}$$

and

$$[B] = \left[\iint\limits_S \Psi_{b,n}^m \Phi_r^j dS\right] \tag{3.2.35}$$

This relation allows to pass from one description of the velocity to another simply by inverting a matrix, as long as the components of the matrix can be expressed. One can therefore use a more suitable expression of the velocity potential for expressing the velocity, while computing the matrices with another one. One can refer to [19] for the choices made in the literature for this change of variables. In fact, the functional space described by the approximation space is the same, but the basis describing this space differ.

However this method is, theoretically, only viable if all the terms up to infinity are considered, which is of course never the case. This means that the truncation chosen with one type of potential may not be represented exactly by an other choice of potential with the same number of terms. The error added to the approximation of the velocity potential by this process could be measured by the distance between the two approximation spaces. However, the effect of the truncation should be mitigated by definition through the use of a spectral method and retain the spectral accuracy.

In the case of the Morillo and Duffy model, the matrices are computed with the following form of the velocity potential:

$$\Psi_n^m = \int\limits_{-\infty}^{\xi_0} \Phi_n^m d\xi \tag{3.2.36}$$

While the value of the velocity uses this form for the

$$\Psi_n^m = \sigma_n^m \Phi_{n+1}^m + \varsigma_n^m \Phi_{n-1}^m \tag{3.2.37}$$

where σ_n^m and ς_n^m are functions defined in [19]. This allows to solve the momentum conservation equation with a form of the potential which is simple, and then to express the value of the velocities with another form of the potential, more suited to this use.

It gives the following form of the equation:

$$i\omega[M^c]\{a_n^{m,c}\} + [D^c][L^c]^{-1}[M^c]\{a_n^{m,c}\} = [D^c]\{\tau_n^{m,c}\} \tag{3.2.38}$$

Here, only the cosine part is considered, and the frequency formulation is assumed for the velocity with a frequency ω.

3.3 One model to rule them all

The Huang and Peters model is in fact constituted of an agglomeration of different models each having different strengths and weaknesses, but all treating the same equations in almost the same manner. It solves however a single equation and uses the method presented above to pass from one model to another. Those models are presented in table 3.3.1.

Model	*Validity Domain*	*Weaknesses*
Morillo-Duffy [19]	Above and on the rotor	Struggle on the rotor
Nowak-He [29]	On the rotor	Struggle out of the disc
Adjoint method [24]	Extend velocities below the rotor	Limited to frequency domain...
Downstream Velocity [1]	High advance ratio, in the wake	Limited to its validity domain, Uses adjoint method

Table 3.3.1: The different models in use in the Huang and Peters model.

The main difference between the models is that each one expresses the velocity potential Ψ_n^m on a different basis, giving them different properties. For example, the Nowak-He model is practical to express the velocity on the disc, while the Morillo-Duffy model performs better above and out of the disc. Since all the models only differ in the basis they consider, one only needs to compute one solution (the momentum equation is solved only once), and then apply a change of basis to obtain the other solutions, for the cost of a matrix multiplication.

The Morillo-Duffy and Nowak-He models can be used in combination to describe the flow above and on the rotor, see Fig. 3.3.1 and 3.3.2. But they struggle to represent accurately the high wake angle cases. Therefore, Huang and Peters in [2] added a third model, the downstream velocity model, specifically suited to represent the high wake angle case. They then perform a blending between those three velocities, depending on the point of space considered (the closer to the disc, the more the Nowak-He model is predominent) and of the wake angle (the higher the wake angle, the higher the part of the downstream velocity), see Fig. 3.3.3.

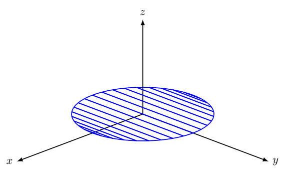


Figure 3.3.1: Schematic representation of the capabilities of the Nowak-He model

Once the velocities on and above the rotor are fully determined, one can apply the adjoint method in order to obtain the value of the velocity below the rotor, as presented on Fig3.3.4.

In the following sections, the way the velocities below the disc are computed will be quickly presented, as well as the way one computes the downstream velocities. And this will lead us to the expression of the final velocity, which blends all the models into one

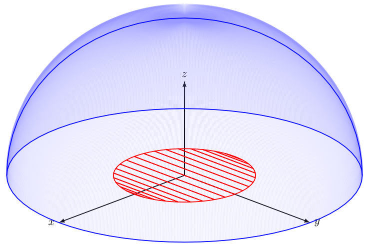


Figure 3.3.2: Schematic representation of the capabilities of the Morillo-Duffy model

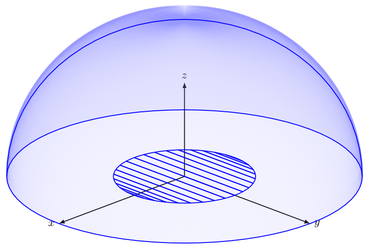


Figure 3.3.3: Schematic representation of the capabilities of the blended model above the rotor. It combines the advantages of Morillo-Duffy above the rotor, and of Nowak-He on the rotor.

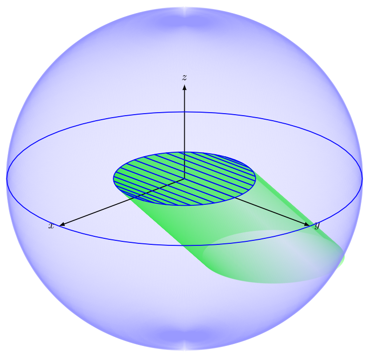


Figure 3.3.4: Schematic representation of the capabilities of the Huang and Peters model

expression, as mentioned above.

3.3.1 Inflow below the disc

The Galerkin method presented above can give an expression of the induced velocity above the actuator disc, but several problems arise when one tries to expand it below the disc, indeed:

- The velocity potential assumption is much harder to justify below the rotor.
- The shape of the wake is initially unknown, and is in principle required to compute the inflow by integration along a streamline.
- The conditions far downstream are unknown, because the wake is unknown. Although one could use open boundary conditions, but they were not considered in the Huang and Peters model.

Nonetheless, Fei shows how to proceed in [24]. Part of the development will be carried out here in order to underline several points where some important hypothesis are taken, and to highlight their impact on the results given by this method.

Following the developments of Fei, who assumes a form of the velocity $\overrightarrow{v} = \overrightarrow{u} + i\overrightarrow{w}$, we begin with the exact solution of the momentum conservation equation by a convolution form, integrated over a streamline:

$$\overrightarrow{u}(\omega, x_0, y_0, \xi_0) = -\int_{-\infty}^{\xi_0} \cos(\omega(\xi_0 - \xi))\overrightarrow{\mathrm{grad}}P d\xi$$

$$\overrightarrow{w}(\omega, x_0, y_0, \xi_0) = \int_{-\infty}^{\xi_0} \sin(\omega(\xi_0 - \xi))\overrightarrow{\mathrm{grad}}P d\xi$$

ξ is, as previously, the parameter of the curvilinear line describing the streamline, and ξ_0 refers to the point of the streamline where the velocity is computed, and is negative above the rotor. The development of the expression is presented in [24, 1], and earlier forms can be found in [19]. It is to be noted that this expression relies solely on the momentum conservation equation, and would require to account for the input pressure jump in order to be extended below the rotor.

Fei's argumentation to expand this model (see [24]) is that the form of the pressure is known also below the disc since it is set by the choice of inputs. He therefore expands the integral form of the velocity to points of the streamlines below the rotor.

This leads Fei to the following expression for $\xi_0 > 0$, i.e. below the disc:

$$\overrightarrow{u} = -\int_{-\infty}^{0} \cos(\omega(\xi_0 - \xi))\overrightarrow{\mathrm{grad}}P d\xi - \int_{0}^{\xi_0} \cos(\omega(\xi_0 - \xi))\overrightarrow{\mathrm{grad}}P d\xi \tag{3.3.1}$$

Several remarks can be made here about the validity of this expression. First, this expression of the velocity does not account for the continuity equation (see section A.3 for a proof). It has the consequence of ignoring the natural wake contraction below the rotor[3]. Then, the hypothesis of straight streamlines is at best dubious for most flight conditions, with the exception of high upstream velocities, where the impact of the induced velocities on the shape of the streamline could be neglected. But this does beg the question of the adaptability of this method to other flight conditions. Finally, the pressure jump across the disc is not accounted for.

In order to use this formulation and to avoid the tedious integration, one needs to link it to the form of the velocity presented above. Fei uses symmetries in the form of the velocity and the adjoint equation, which is the momentum conservation equation where the time is replaced by $-t$, in order to do so. The adjoint equation gives a new form of the velocity, called adjoint velocity, which is used by Fei to express the velocity below the disc. It does however require to solve a new equation. Only the results of this development will be presented here, and we refer the reader to Fei's dissertation for more details [24].

[3]This can be justified by the fact that the free stream velocity is high compared to the induced velocity. This effect is accounted for by Huang and Peters in [2, 1] by post-treating the velocity and applying a re-mapping to match the contraction, but is treated as a fix to the effect of the mass flow parameter. This is coherent with the vision of adapting to the free stream velocity

Fei obtains the following expression for the velocity below the disc:

$$\begin{aligned}\overrightarrow{V}(\tau, x, y, z) = \overrightarrow{V}(\tau - \xi_0, x_0, y_0, 0)\\ + \overrightarrow{V^*}(\tau - \xi_0, -x_0, -y_0, 0) - \overrightarrow{V^*}(\tau, -x, -y, -z)\end{aligned} \tag{3.3.2}$$

where $\overrightarrow{V^*}$ designate the adjoint velocity, τ is the non dimensional time and ξ_0 is the streamline coordinate. Fig. 3.3.5 shows the points considered in this expression. Thus, in order to compute the velocity at a point below the rotor, one needs the values of velocities at three different points: one on the rotor disc, up the considered streamline, and two adjoint velocities on the centro-symmetric streamline. Note that the velocities on the rotor are shifted in time correspondingly to the distance to the rotor of the considered point of calculation.

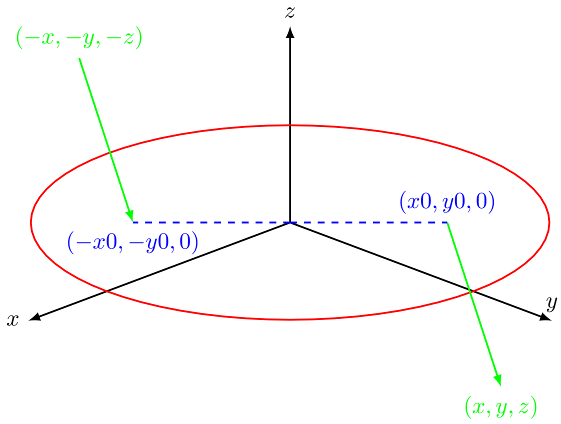


Figure 3.3.5: Points of interest used for the computation of the velocity below the disc

3.3.1.1 Expression of velocity for perfectly edgewise flow

Huang developed in [1, 2] a method using the same theoretical background as the adjoint method of Fei in order to express the velocity in the perfectly edgewise flow case. Huang considers a 90° wake angle and apply to this flow the adjoint method in order to compute the velocities in the plane of the rotor and above. This gives him the following expression:

$$\begin{aligned}\overrightarrow{V_{DS}}(\tau, x, y, z) = \overrightarrow{V_{BL}}(\tau - \sigma\sin(\chi), -s_0, y, z)\\ + \overrightarrow{V^*_{BL}}(\tau - \sigma\sin(\chi), s_0, -y, z) - \overrightarrow{V^*_{BL}}(\tau, \sigma + s_0, -y, z)\end{aligned} \tag{3.3.3}$$

For this expression, Huang considers a distance s_0 to the disc where the wake is converged by the blended model, and replace the distance ξ in the adjoint method, by $\sigma = x - s_0$. ξ represents the distance of the point considered to the rotor plane disc, i.e. the distance along the streamline from the point considered to the space where the velocities are known. Thus, σ serves the same purpose, but from the point considered to the point where the velocities are converged. However, its value might differ depending on the component of the velocity considered, since the convergence of the blended model does not behave similarly for all components. One can refer to [1] for a thorough explanation of this method and of the differences between the components.

Furthermore, since the adjoint method will be applied to this velocity, one also needs the expression of its adjoint, which is given by:

$$\begin{aligned}\overrightarrow{V^*_{DS}}(\tau,x,y,z) = &\overrightarrow{V^*_{BL}}(\tau+\sigma\sin(\chi),-s_0,y,z)\\ &+\overrightarrow{V_{BL}}(\tau+\sigma\sin(\chi),s_0,-y,z)-\overrightarrow{V_{BL}}(\tau,\sigma+s_0,-y,z)\end{aligned} \tag{3.3.4}$$

These formulae suffer from the same criticisms as for the velocity obtained below the rotor, since they are based on the same theoretical concepts. However, it adds a detail that will latter be a problem, when developing a time domain solution. Indeed, to compute the downstream velocity and its adjoint at a time τ, one needs the values of V_{BL} at time $\tau+\sigma\sin(\chi)$ and $\tau-\sigma\sin(\chi)$. In the frequency domain, this is not a concern, since the time of computation is irrelevant, but in the time domain, it will require velocities that are not yet computed. We will see the consequences of this problem in the time domain formulation presented in section 4.1.1.

3.3.2 Blending of the various solutions

Huang and Peters model provides one with several expressions for the velocity:

- Above the disc rotor: V_{MD}
- On the rotor disc: V_{NH} for the z component and V_{HH} for the x,y components
- A blending of those two velocities: V_{BL}
- The velocity for perfectly edgewise flow: V_{DS}
- The blending of the two previously mentioned: V_F

Furthermore the last expression is extended to the cases below the disc thanks to the adjoint method presented earlier.

The blending take the following general form:

$$V_{BL} = \alpha V_{MD} + (1-\alpha)V_{NH} \tag{3.3.5}$$

$$V_F = \beta V_{BL} + (1-\beta)V_{DS} \tag{3.3.6}$$

where α depends on the distance to the disc, while β depends on the wake angle χ. The precise values of this terms are determined by Huang in his thesis [1] thanks to a curve fitting method.

3.3.2.1 Blending functions

Not much is said in the literature about how the blending functions were derived. The only thing we know is that a curve fitting process was used. This raises several questions about the number of cases used, the order of the curve fitting, and about the limitation and validation of this method. Indeed, some cases do not seem to match the exact solution, but it is hard to predict if it is due to a lack of precision of the method, to a limitation of the validity domain of one of the sub-model or to the curve fitting, that overlooked this specific case.

We here present a few cases that do not match the exact solution as well as the presented cases. All are done using $m_{odd}=20$, $m_{even}=10$ which is considered to be a quite accurate approximation. One can however see, in Fig. 3.3.6 and Fig. 3.3.7 that the exact solution and the proposed blended solution do not match, but that the Morillo-Duffy model provides a better approximation of the exact solution. This case is in the middle of the validity case of the models, with $\chi = 45°$, $\omega = 10$, $y = 1$ and $z = -0.3$. This reveals the problem that the blending solution is in fact changing the validity domain of the model, although the Morillo-Duffy model is valid at the points considered.

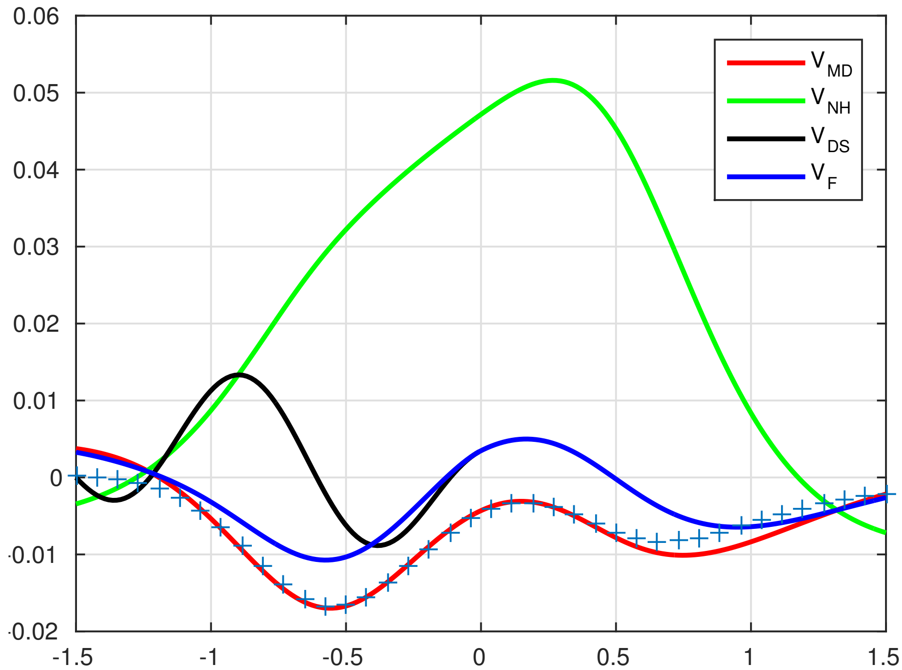


Figure 3.3.6: Comparison of Huang and Peters model with the exact solution. Axial velocity, real part, for $m_{odd} = 20$, $m_{even} = 10$, $\chi = 45°$, $\omega = 10$, $y = 1$, $z = -0.3$

Cases computed below the rotor also give inaccurate results, as can be seen on Fig. 3.3.8 and Fig. 3.3.9, but it is harder to guess where the problem might come from and if another blending could have improved the solution.

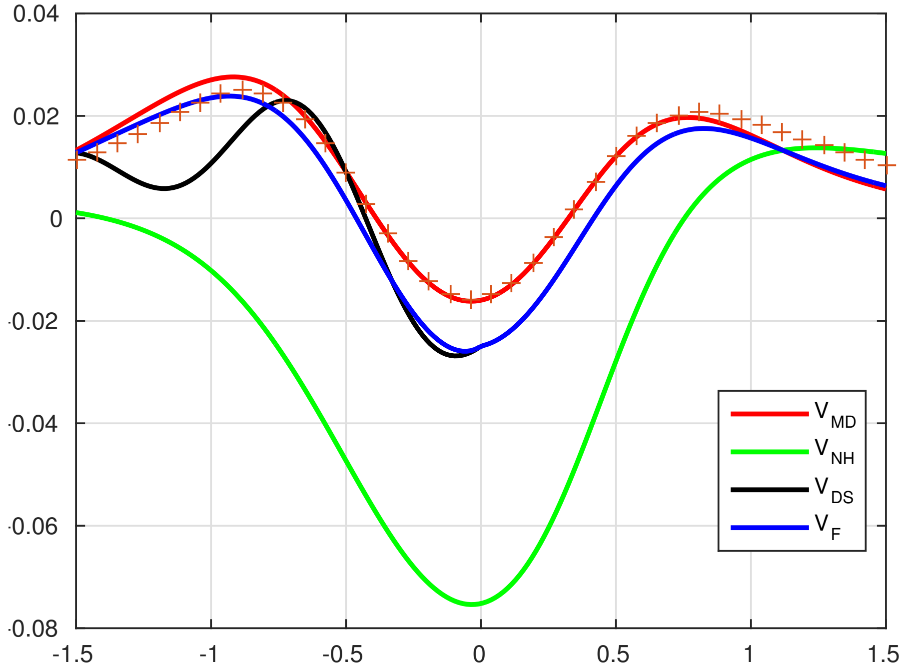


Figure 3.3.7: Comparison of Huang and Peters model with the exact solution. Axial velocity, imaginary part, for $m_{odd} = 20$, $m_{even} = 10$, $\chi = 45°$, $\omega = 10$, $y = 1$, $z = -0.3$

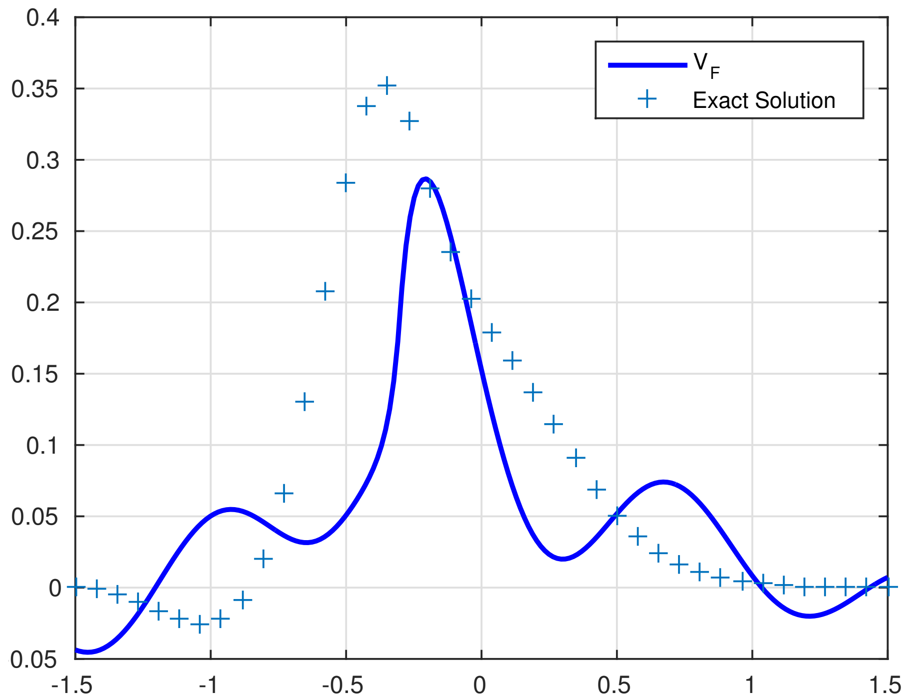


Figure 3.3.8: Comparison of Huang and Peters model with the exact solution. Axial velocity, real part, for $m_{odd} = 20$, $m_{even} = 10$, $\chi = 45°$, $\omega = 10$, $y = 1$, $z = 0.3$

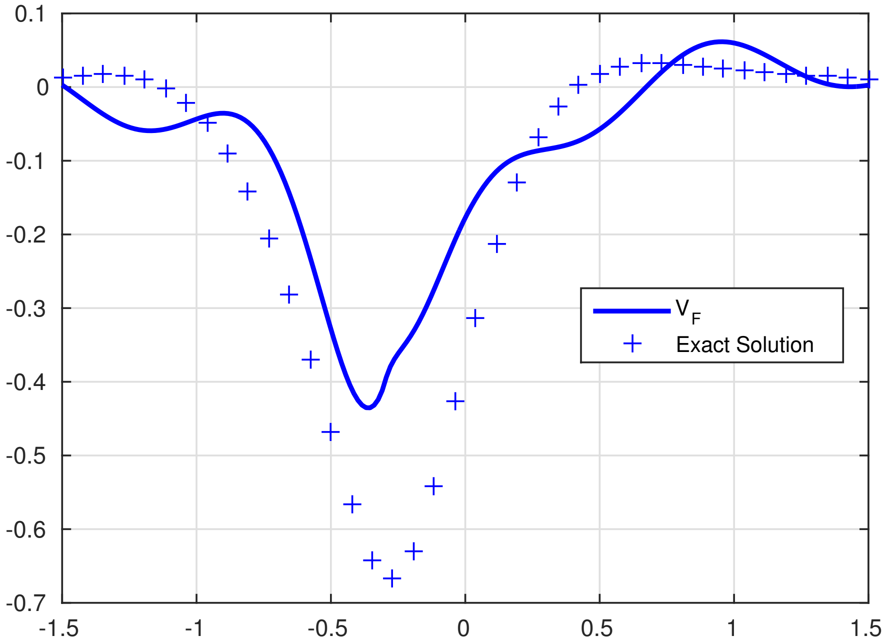


Figure 3.3.9: Comparison of Huang and Peters model with the exact solution. Axial velocity, imaginary part, for $m_{odd} = 20$, $m_{even} = 10$, $\chi = 45°$, $\omega = 10$, $y = 1$, $z = 0.3$

3.3.3 Change of coordinates to express the NH and HH variables

The values of the $[S]$ matrix presented in [1, 2] are not accurate. We here present their derivation and, hopefully, true right values.

Using the Galerkin Method Using the change of basis method presented above, which relies on a Galerkin method, we develop the terms for the matrix allowing to pass from the Morillo-Duffy model basis to the basis of the Nowak-He and Huang-He model.

We look at the matrix $[S]$ dealing with the x component of the velocity which does not seem to reduce to Huang's proposed form.

In order to obtain this change of basis, the expression of the velocity is reminded here, for the x component, as given by Huang:

$$\nu_{x,MD} = \sum_{m\geq 0}\sum_{n\geq m} a_n^m v_{x,n}^m \tag{3.3.7}$$

$$\nu_{x,HH} = \sum_{(m+n)odd} \alpha_n^m \frac{P_n^m}{\nu} Q_{m+1}^m \cos(m\psi) + \sum_{(m+n)even} a_n^m v_{x,n}^m \tag{3.3.8}$$

$$\tag{3.3.9}$$

With the scalar product on the rotor disc surface[4]:

$$\langle f, g\rangle = \iint_S fgdS \tag{3.3.10}$$

The Galerkin method gives:
$\forall (r,j), j > r$:

$$\langle \nu_{x,MD}, \Phi_j^r\rangle = \langle \nu_{x,HH}, \Phi_j^r\rangle \tag{3.3.11}$$

Which can be put under the form proposed by Huang:

$$\begin{pmatrix} (\alpha_n^m)_{odd} \\ (a_n^m)_{even} \end{pmatrix} = \begin{bmatrix} S & 0 \\ 0 & I \end{bmatrix} (a_n^m) \tag{3.3.12}$$

But where the components of the S have a different value for the $r = m-1$ cases:

$$S_{n,j}^{m,r} = 0 \quad \text{for } r \neq m \pm 1 \tag{3.3.13}$$

For $r = m+1$:
If $m = 0$:

$$S_{n,j}^{m,r} = \begin{cases} \sigma_n^0 \sqrt{(j+1)j} & \text{for} \quad j = n+1 \\ \varsigma_n^0 \sqrt{(j+1)j} & \text{for} \quad j = n-1 \\ 0 \quad \text{for} \quad j \neq n \pm 1 \end{cases} \tag{3.3.14}$$

[4]The Jacobian on the disc, in ellipsoidal coordinates, is $|\nu|$

Otherwise:

$$S_{n,j}^{m,r} = \begin{cases} \frac{1}{2}\sigma_n^m\sqrt{(j+m+1)(j-m)} & \text{for} \quad j=n+1 \\ \frac{1}{2}\varsigma_n^m\sqrt{(j+m+1)(j-m)} & \text{for} \quad j=n-1 \\ 0 \quad \text{for} \quad j \neq n \pm 1 \end{cases} \tag{3.3.15}$$

For $r = m - 1$:

$$S_{n,j}^{m,r} = \begin{cases} -\frac{1}{2}\sigma_n^m\sqrt{(j+m)(j-m+1)} & \text{for} \quad j=n+1 \\ -\frac{1}{2}\varsigma_n^m\sqrt{(j+m)(j-m+1)} & \text{for} \quad j=n-1 \\ 0 \quad \text{for} \quad j \neq n \pm 1 \end{cases} \tag{3.3.16}$$

The values of the scalar product were computed thanks to appendix A.1.1 which presents the expression of the derivatives of Φ_n^m by x. σ_n^m and ς_n^m are helping functions defined in [19, 1]. These values allow to reproduce the results presented in [1], which did not present the same values for this matrix.

3.3.4 Extensions of the Model

Once the model completed with the blending of all various velocities, Huang expands it through the use of different non linear extensions. Those extensions attempt to improve the validity domain of the model by tackling some of its downsides. We will review these non linear extensions here, as well as some of them present in the literature, and comment on them.

3.3.4.1 Mass flow parameter

Euler's equations have been linearised in order to develop Huang and Peters method. However this has a downside, which is that it will ignore the non linearity of this phenomenon. An intuitive way to see this is in the V_∞ parameter introduced earlier by the linearisation. This parameter will be null in the hover case, which would lead to an equation without any convection, which is unacceptable.

In order to take into account this non linearity, the mass flow parameter method was introduced and consists in replacing V_∞ by a matrix $[V]$ depending on the inflow through a parameter, called the mass flow parameter. The main problem is now to find what should be put in this matrix.

As shown by various authors, it can be simple to determine those coefficients in specific cases, as in hover, climb, axial flow, etc...through momentum theory. The main problem consists in finding a general mass flow parameter, for any flight conditions, respecting all the limit cases that can be solved.

As of today, the solution is the one first given by Peters and Haquang in [28], although this expression has been criticised for the special cases of auto-rotation or windmilling. It is formed by the following diagonal matrix:

$$[V] = \begin{bmatrix} V_T & 0 & 0 & \cdots \\ 0 & V_{MFP} & 0 & \\ 0 & 0 & V_{MFP} & \\ \vdots & & & \ddots \end{bmatrix} \tag{3.3.17}$$

Where $V_T = \sqrt{\mu^2 + \lambda^2}$ and V_{MFP} is the mass flow parameter expressed as:

$$V_{MFP} = \frac{\mu^2 + \lambda(\lambda + \lambda_m)}{\sqrt{\mu^2 + \lambda^2}} \tag{3.3.18}$$

where μ is the advance ratio, $\lambda = \lambda_f + \lambda_m$ is the total inflow velocity, with λ_m the mean axial induced velocity due to rotor thrust and λ_f the axial free stream velocity. All are nondimensionalised by the blade tip speed.

The mass flow parameter depends on the induced velocity, hence the non linearity, but requires a sensible expression for λ_m. The various expressions of λ_m present in the literature depend on the model of induced velocity, but all seem to focus on the fact that only the velocities generating some thrust are accounted for. This means that the main part taken into account is the mean value of the induced velocity, thus ignoring the oscillating terms.

Thus, by definition of the thrust coefficient, which can be expressed in the case of the current model as:

$$\begin{aligned} C_T &= \iint_S \Delta p dS \\ &= \frac{2}{\sqrt{3}} \tau_1^{0,c} \end{aligned}$$

And following the momentum theory definition for the axial induced velocity due to rotor thrust:

$$\lambda_m = \frac{C_T}{2V_T} \tag{3.3.19}$$

This gives that λ_m is only linked to the $\tau_1^{0,c}$ term in steady conditions. However in order to account for a non steady state, the following expression is taken:

$$\lambda_m = \frac{1}{\sqrt{3}}(1, 0, -, 0)[L]^{-1}[M](a_n^m) \tag{3.3.20}$$

Which therefore only account for the terms having an influence on the $\tau_1^{0,c}$ term.

The above expression is however not the one presented in the literature (Duffy [64] and Huang [1] both have a different one), but the author believes it is more coherent with the one used by both the Pitt and Peters model and the He and Peters model.

Nevertheless, this expression could be criticised because of the presence of some even terms in τ_n^m. Indeed those terms create some thrust by adding mass to the flow. This would highly modify the above expression of λ_m but would only concern the very specific cases of mass flow injection.

Finally, this non linear method will require to add an extra loop to the method in order to converge the value of λ_m.

3.3.4.2 Wake skew angle non linearity

The idea behind this extension is to account for the effect of the induced flow in the value of the wake skew angle. This is done by accounting for the mean value of the axial induced velocity at the rotor disc in the expression of the wake skew angle.

However, this new definition of the skew angle, given in [29, 1], raises many questions. Indeed, does such a wake skew angle, define globally makes sens ? If one looks at each points locally, the impact on the wake skew angle of the induced velocity is far from constant. Although the mean axial induced velocity is a good approximation, is it accurate enough ? No definite answers to those questions could be found in the literature.

This extension also requires a convergence step. But it could admittedly be done simultaneously with the mass flow parameter convergence.

3.3.4.3 Accounting for the wake deformations

This non linear extension aims at accounting for the wake deformation effects on the induced velocities. It was initially developed as an extension of the Peters and He model in [37], following previous developments made on the Pitt and Peters model. It was further refined by Zhao et al. in [38, 39]. It consists in modifying the $[L]$ matrix of the model in order to account for pitch and roll rates in manoeuvring flights, defined through a rotor wake distorsion parameter κ. Zhao model accounts for the dynamic evolution of this parameter (as well as the dynamic evolution of the wake skew angle) in order to accurately describe the effect of manoeuvring flights.

Although this is more accurate than the original straight tube wake description, one can again wonder if this extension allows to describe the full range of possibilities of the helicopter. For example, the wake curvature parameter only allows to describe a constant wake curvature, which is limiting. Furthermore, it does not represent the wake at the transition between, for example, a hover flight and a constant pitching manoeuvre motion. This explains the need for a time delay in the application of the curvature that Zhao computed thanks to comparison with a vortex tubes method. The time delay simply represents the time necessary for the wake to have a predominant part redefined by the curved wake.

Finally, although this method is at present applied to the equations in the Huang and Peters model, there is no mention of its impact on the adjoint equation and therefore on the velocities below the rotor. Indeed, the adjoint equation heavily relies on the straight streamline assumption in order to be developed. Accounting for wake curvature should therefore also be done in this method. Furthermore, the distribution of the velocities below the rotor should also account for wake curvature. It is however easier to apply the wake deformation to the computed velocities, than to account for the wake curvature in the adjoint method. We refer the reader to section 4.3.1 for an exploration on the impact of removing the straight streamline assumption.

3.4 Results and comparison

The Huang and Peters model has been implemented, and it was verified that the results obtained were identical to the ones presented by Huang in [1]. A few examples are reported here in order to demonstrate the validity of our implementation.

Figure 3.4.1 reproduces figure 4.2 of [1], figure 3.4.2 reproduces figure 4.5 of [1] (both with $m_{odd} = 12$ and $m_{even} = 8$), and figures 3.4.3 and 3.4.4 reproduce figures 4.14 a) and b) respectively (with $m_{odd} = 6$ and $m_{even} = 4$).

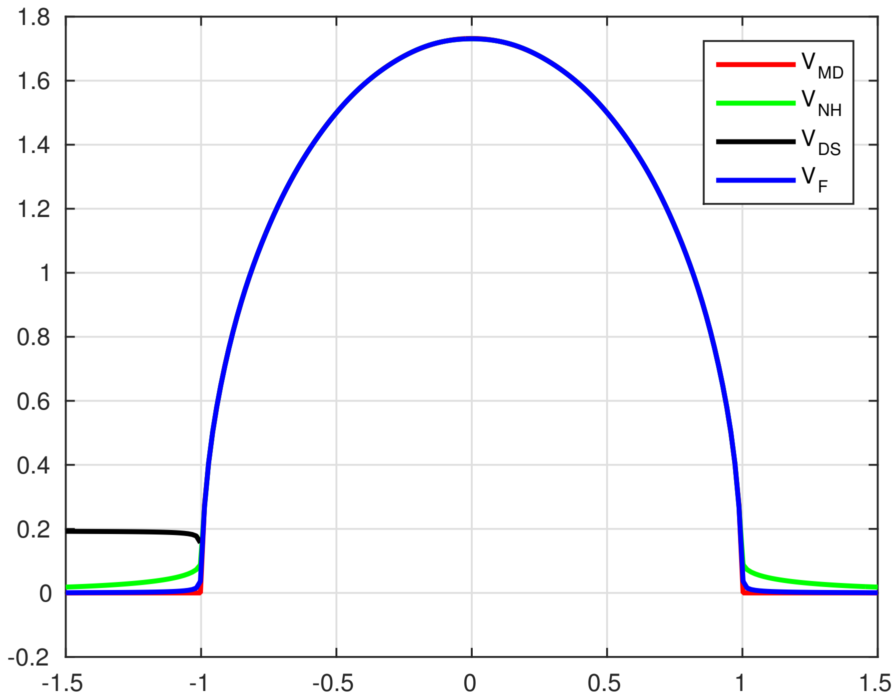


Figure 3.4.1: Axial velocity on the disc for $\omega = 0$, $y = z = 0$, $\chi = 0$ and τ_0^1 as pressure input.

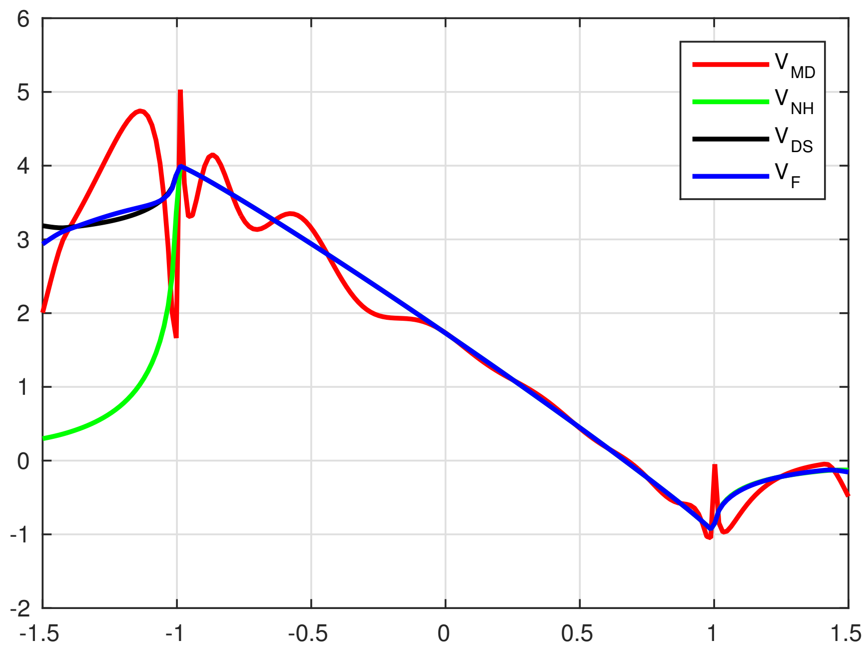


Figure 3.4.2: Axial velocity on the disc for $\omega = 0$, $y = z = 0$, $\chi = 85$ and τ_0^1 as pressure input.

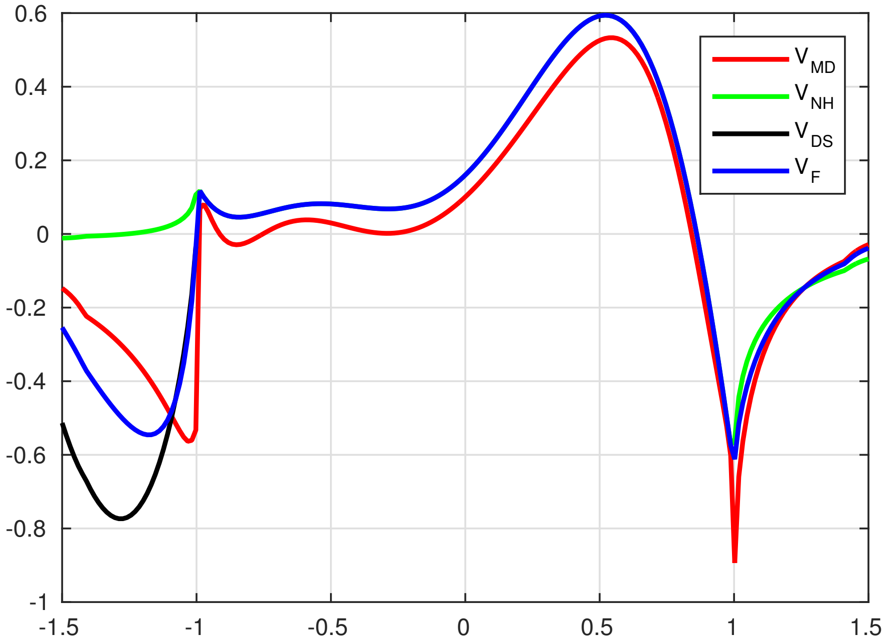


Figure 3.4.3: Axial velocity on the disc for $\omega = 4$, $y = z = 0$, $\chi = 60$ and τ_0^1 as pressure input, real part.

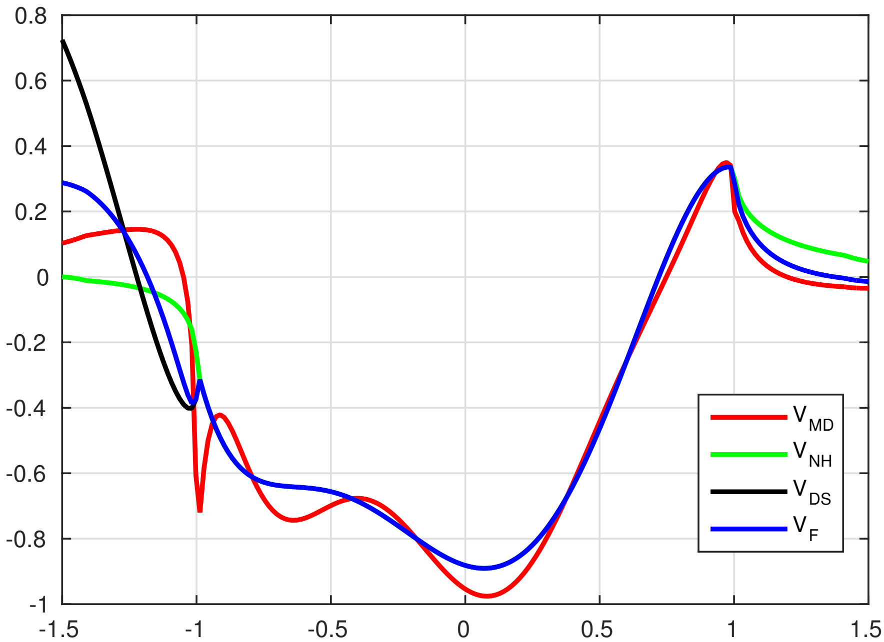


Figure 3.4.4: Axial velocity on the disc for $\omega = 4$, $y = z = 0$, $\chi = 60$ and τ_0^1 as pressure input, imaginary part.

3.5 Conclusion on Huang and Peters model

The analysis of the Huang and Peters model allows us to draw a number of conclusions. First of all, this model has a number of advantages. It is efficient: considering the precision and resolution it offers on the rotor disc, it takes little time to solve the equations for a given frequency. It is adaptable, meaning that by choosing the number of terms present in the Galerkin approximation, one can adapt the model to the needs of the considered application, from a fast but not entirely accurate model, to a not so fast, but accurate model. It gives all the components of the velocities at all points on and around the rotor, thus providing a unified dynamic model for computing the rotor induced velocities both on the rotor blades and on the other helicopter components. This means, for example, that one can use this model to study the aerodynamic interaction due to the rotor wake on the other rotorcraft components and to obtain a flight dynamics model able to represent phenomena such as the pitch up effect due to the rotor wake blowing on the horizontal stabiliser for a certain range of forward speeds.

However, a number of downsides could also be drawn from the study of this model. The easiest way to put it is to say that the validity domain of this model is limited. First it is limited by the assumptions it takes in order to treat the equations. Indeed, if one takes the linearisation assumption for example, it implies a high value of the free stream velocity in comparison to the induced velocity value. This implies that the validity domain for this model is limited to these cases, and will be hardly applicable to represent the near hover cases. To improve its validity domain, the model is augmented with non linear extensions, such as the mass flow parameter in order to treat the hover case. The main issue with this fix, is that it seems to create more problems (requiring another non linear extension, and a convergence step), and that it is hard to find a justification for it. The same drawbacks can be observed for the wake skew angles non linear extension.

Another fact limiting the validity domain of the model is the reference to the so called "exact solution" for validation. This method could be used for the validation of the results obtained above the rotor for high free stream velocity. However, below the rotor, this expression requires a lot more assumptions in order to be justified, if it can truly be justified. Nonetheless, it is used as a foundation for the derivation of the adjoint method.

Lastly, the use of blending functions limits the validity domain of the model. As mentioned in 3.3.2.1, the lack of information on the derivation of the blending functions prevents to truly define the validity domain of the model.

Finally the attempts made to improve the model, as non linear extensions, are costly, and do not always bear all their promises.

The main issue with all the above limitations, is that the author could not find a clear definition of the validity domain. From this point, two roads could have been taken. The first one is a nice but boring path, requiring the careful study of the validity domain, and giving as sole result a clear view of what's not possible. The second road is much more exciting, but looks like a dark unkempt forest path. It leads, if one can make it to the end, to the improvement of the model.

The remaining part of this thesis consists in the exploration of this second path, and in the description of what was found on the way.

4

Extension and Improvements

Résumé en français: Extension et améliorations Ce chapitre tire les conclusions du précédent et tente d'améliorer le modèle de Peters et Huang de diverses manières. Tout d'abord une extension au domaine temporel de la formulation frequentielle de Peters et Huang est proposée. En effet, celà permet de coupler ce modèle à un modèle par éléments de pale, et ainsi de le comparer à d'autres modèles existants. Cette comparaison révèle de nouveau certains problèmes, et met le doigt sur l'origine de ces problèmes.

Dans un second temps, des améliorations plus théoriques sont proposées. Elles permettent d'améliorer l'efficacité et la précision du modèle.

Enfin, plusieurs tentatives d'améliorations sont présentées. Celles-ci ne sont pas concluantes, mais montrent le chemin qui a conduit à une refonte plus profonde présentée dans le dernier chapitre.

4.1 Extension to the time domain

In this section the model proposed by Huang and Peters is extended to the time domain, where it can be more easily compared to various models who already live in this domain and are coupled with a blade element model. To do so it has been coupled to a code already implementing other induced velocity models, and also capable to integrate a wide variety of geometries.

4.1.1 Motivation

The need for a time domain extension of the Huang and Peters model arose from the necessity to couple it with a blade element model. Indeed, the main advantage of the frequency domain is to give a solution for all frequencies considered, giving a valid solution for all points in time. But this advantage is lost when coupling the model with a blade element model. Moreover, a time marching model is more suited for many applications such as flight dynamics simulation. The coupling between both models does not allow to foresee the input of the induced velocity model for all the frequencies, except in a stationary case. Indeed, the variation of the inputs of the model are unknown, because the influence of the induced velocities on the blades will significantly modify their behaviour, and thus the inputs of the model. This means that it is difficult to compute the frequency representation of the blade airloads (the inputs of the model). A problem that is not present with a time domain formulation.

Furthermore, having a time domain model means having only one set of coefficients describing the velocity, whereas a frequency domain solution requires a set of coefficients for each considered frequency.

4.1.2 Principle of the extension

In practice the only facts limiting the use of Huang and Peters model to the frequency domain are the validation they use, where the time dependence is assumed to be in frequential form, and by the use of the adjoint theorem.

The first limiting factor can be dealt with by saying that thanks to the linearity of the equation and because a temporal signal can simply be expanded on an infinity of frequencies, the validation done in the frequential domain validates the temporal domain.

I.e., if the pressure and velocities are viewed as follows, the validation done the frequency domain is still valid in the time domain.

$$\overrightarrow{\nu} = \int_{-\infty}^{\infty} \overrightarrow{\nu}_{\omega}(x, y, z) e^{i\omega t} d\omega \tag{4.1.1}$$

$$p = \int_{-\infty}^{\infty} p_{\omega}(x, y, z) e^{i\omega t} d\omega \tag{4.1.2}$$

For the second problem, the same remark as previously still holds true, meaning that the adjoint theorem can be extended to the time domain, although the use of symmetries will be harder to justify. However, as mentioned in section 3.3.1.1, the downstream velocity will be problematic to implement, as it refers to times that are not yet computed.

4.1.3 Testing the extension

The time domain version of the code was implemented in Python and tested through comparisons of the steady-state response. Although this method does not cover all cases, it allows to validate the time domain model with the cases that validate the frequency domain model. However, it does not validate the transitory phases. But this will be done thanks to the comparison to other induced velocity model. Indeed, if the steady state response agrees with the frequency domain model, and if the transitory phases can be compared to other models, the implementation can be deemed to be satisfying.

Hereafter a few examples of comparisons are shown, validating the agreement between Huang's frequency formulation and the time domain model.

4.1.4 Coupling the model

In this section the temporal domain extension is coupled with the home-made ONERA code AMB (Aero Multi Body), implemented in Python by Philippe Beaumier. AMB will be presented, as well as the integration of the induced velocity model.

4.1.4.1 AMB

AMB allows to represent all kind of aircraft, although we will only use it in order to model isolated rotors. It allows to compute the induced velocities with several methods, notably prescribed and free wake models, and integrate a blade element model for a wide range of flight conditions. It was originally dedicated to presizing of rotorcrafts [65]. One drawback of this code is that it does not provide a way to compute trimmed solutions, i.e. the required rotor pitch angles for generating the forces and moments corresponding to the equilibrium of the specified flight condition. Therefore, the following computations will be made by imposing the blades kinematics (pitch, flap and lead-lag angles). Furthermore, this will allow to ensure that the only differences between two computations are only the induced velocity models, since all other elements will be identical.

In order to converge the induced velocities, AMB uses a loop on the forces generated by the rotor, and a relaxation method can be used on the induced velocities. In the implementation, rotors are represented with a *disque_rotor* class which allows to treat them as an actuator disc, and can give a basic evaluation of the induced velocities based on momentum theory. Our implementation of the temporal domain model will come in place of this existing induced velocity model, in order to ease the integration, by creating a daughter class of the *disque_rotor* class.

4.1.4.2 Integration

We will first present the architecture of the module made and its integration to AMB, before going into more details about the specifics of the time domain application.

Architecture Fig 4.1.1 presents the Unified Modelling Language (UML) Class diagram of the module implemented. It is constituted of 6 classes. The first one *ViInputs* reads an input text file and computes some required values from them. It also stores some required values for a quicker the computation of the matrices. The *ViMatrices* class computes all required matrices. Notice that all matrices are computed at the initialisation, but that one can compute new values for $[L_c]$ and $[L_s]$ if the skew wake angle is to be changed. The *ViStatesTemp* class inherits from the *ViMatrices* class, and adds all the required states for the computation of the velocity. It also provides a method to compute the velocities at the next time step. In order to compute the velocity value at a given point, one needs first to compute the associated Legendre functions, which is done by the *LegendrePoly* class. One also needs to be able to go from cartesian to ellipsoidal coordinates, which is made possible thanks to the *Coordinates* class.

All those classes are then implemented in the *InducedVelocity* class, which provides the tools to set the pressure inputs at each time step, and to compute the value of the velocity at any given point.

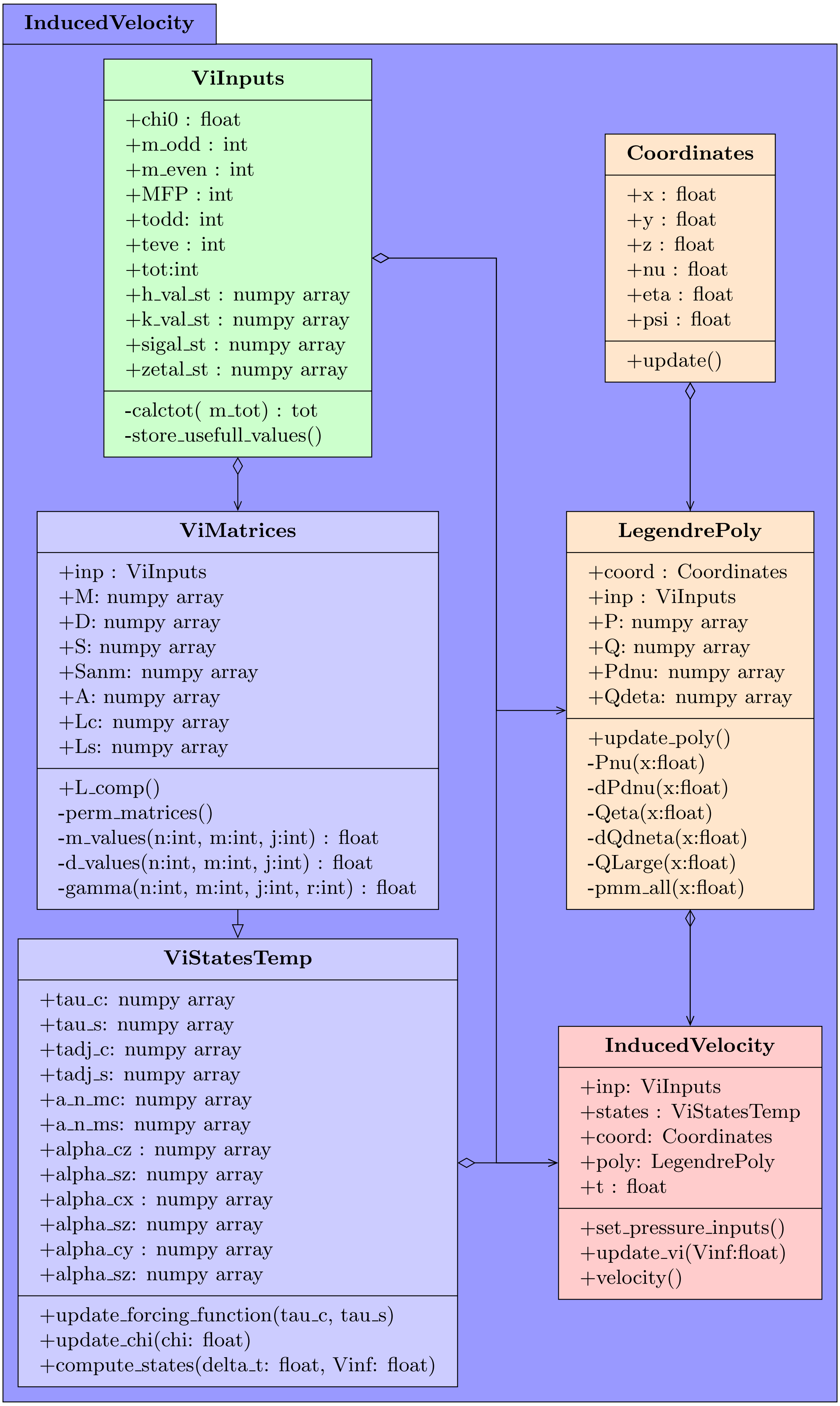


Figure 4.1.1: UML Class Diagram of the temporal implementation of Huang and Peters method.

Time scheme The implementation of the time scheme was made with the simplest possible scheme, a Euler forward step, with a time step δ_t, and is presented on the cosine part:

$$[M^c]\frac{a^{m,c}_{n,k+1} - a^{m,c}_{n,k}}{\delta_t} + [D^c][L^c]^{-1}[M^c]\left\{a^{m,c}_{n,k}\right\} = [D^c]\left\{\tau^{m,c}_{n,k}\right\} \tag{4.1.3}$$

Which can be simply solved with:

$$a^m_{n,k+1} = a^m_{n,k} + \delta_t[M^c]^{-1}\left(-[D^c][L^c]^{-1}[M^c]\left\{a^{m,c}_{n,k}\right\} + [D^c]\left\{\tau^{m,c}_{n,k}\right\}\right) \tag{4.1.4}$$

It was thought that in a first time, in order to assess the capacities of the algorithm, this method would be sufficiently accurate, and could be easily controlled by the size of the time step. After test and implementation, it was shown that the only drawback of this simple time step was the fact that it limited the size of the approximation that could be considered.

Computation of the inputs of the model In the case of a coupling of the induced velocity code with an external blade lift theory, the computation of the pressure inputs may not be explicit. The following derivation applies the method presented by Peters and He in [8]. It does require some adjustments to be applied to the Huang and Peters model, since it is originally applied to the Peters and He finite state induced velocity model, which uses an other polynomial basis for the τ^m_n than the one considered here.

The rotor has a finite number of blade, Q, generating lift. Thus the pressure distribution on the rotor is constituted of 'pressure spikes', rotating with the blades.

The q^{th} blade, at azimuth ψ_q, generate a pressure $p_q(r, y, t)$, where y is defined from $-b$ to $+b$, b being the half chord of the blade, also seen by the angle $\psi = \psi_q + \sin^{-1}(\frac{y}{r})$. This pressure is taken under the following form:

$$p_q(r, y, t) = \sum_{m,n} P^m_n(\nu)\left[\tau^{m,c}_n \cos(m\psi) + \tau^{m,s}_n \sin(m\psi)\right] \tag{4.1.5}$$

It is thus possible to express the pressure coefficients thanks to the orthogonality properties of the Legendre's polynomials:

$$\tau^{m,c}_n = \sum_{q=1}^{Q}\langle p_q(r, y, t), \Phi^{m,c}_n\rangle \tag{4.1.6}$$

$$\tau^{m,s}_n = \sum_{q=1}^{Q}\langle p_q(r, y, t), \Phi^{m,s}_n\rangle \tag{4.1.7}$$

These equations are the main differences with the derivation presented in [8], since the factors used in the paper are not the $\Phi^{m,c}_n$.

Blade lifting line theory does however not provide the chordwise pressure. The aim of this derivation is thus to express the pressure coefficients, through the lift generated by a blade, given by any blade lift theory.

Following [13], it is therefore assumed that the pressure can be put under the following form:

$$P_q(r, y, t) = \hat{L}_q(r, t)\frac{P_y(y)}{\int\limits_{-b}^{b} P_y(y)\,dy} \tag{4.1.8}$$

With $\hat{L}_q$ the non dimensional lift generated by blade q. The pressure coefficients can then be expressed as:

$$\frac{1}{2}\left(\tau_n^{m,c} - i\tau_n^{m,s}\right) = \frac{1}{2\pi}\sum_{q=1}^{Q}\int_0^1 f_m L_q \Phi_n^m d\nu \, e^{-im\psi_q} \tag{4.1.9}$$

with:

$$f_m = \frac{\int_{-b}^{b} P_y(y) e^{-im\sin^{-1}\left(\frac{y}{r}\right)}\, dy}{\int_{-b}^{b} P_y(y)\, dy} \tag{4.1.10}$$

For a lifting line theory, $f_m = 1$, which will be the case considered here, since AMB uses a lifting line method.

It is to be noted however, that the pressure states τ_n^m are, in our case, not exactly extracted with the above equation. Indeed, the basis used is not orthonormal, which will thus require an inversion of the obtained terms in order to truly represent the τ_n^m in the basis used by the algorithm. Another way to achieve the same result, is to remark that the given integral is the result of the scalar product by the gradient of the test function used, and thus to consider that the computed states are in fact $[D]\{\tau_n^m\}$ rather than the τ_n^m. This allows their direct integration to the equations, since they will be anyway multiplied by the matrix $[D]$.

Another problem raised by this method is due to cases considering flapping motion of the blades out of the rotor disc plane. Indeed in those cases the very definition of the rotor disc is threatened. Several solutions to this conundrum might be considered: either take the rotor disc as fixed relatively to the rotor mast, or take it to be the rotor blade tip path. Both definitions are identical in the case of rigid blades with no flap, and both need adjustment when considering either flap motion and/or elastic blades. These adjustments are required because the theory requires to compute the pressure distribution on the rotor disc, and blades may not be contained in this disc. A usual method to fix this problem consists in projecting the force on the disc along the direction of the normal to the disc, regardless of the position of the blades.

Adjoint equation and induced velocities below the rotor The implementation of the adjoint equation in the time domain is not as easy as it could seem. Indeed, their are a few problems to solve.

The first one is the treatment of the velocities computed thanks to their adjoint. Indeed, the computation of these velocities require to access previous time of the simulation. This requires, first, to store all the previous time step states, then, to interpolate between the time steps computed when required by the point at which the velocity is computed. This was a hindrance for the algorithm performance, but could be solved efficiently[1].

The second major problem is the adjoint of the downstream velocity. In a specific region fore of the rotor and below it, the adjoint of the downstream velocity is required to be known at a time that is not yet computed in the simulation. The specific region is constituted of the part of space D defined by:

$$D = \{(x, y, z) \,|\, x^2 + y^2 + z^2 \geq \cos(\chi) \,\text{and}\, x \leq 0, z < 0\} \tag{4.1.11}$$

[1]The use of the cPickle module greatly improved the time required to pack and unpack the data.

This definition is intimately linked to the definition of the downstream velocity and to the region where it is required to compute its adjoint. It may seems strange that this problem occurs fore of the disc, when the downstream velocity is used aft of the disk. However, it the adjoint expression of this velocity that is problematic, and not the velocity itself, hence the non intuitive shape of the domain. The method used to compute this domain is presented in appendix A.4. Therefore, below the disc and with some advance ratio, the adjoint of the downstream velocity V_{DS} may require to be computed at a time in the future 3.3.1.1). In practice, for helicopter flight dynamics, the induced velocities have to be computed on the main rotor blades and outside of the rotor mainly on the tail components. Therefore the issue revealed here is not a significant downside of the Huang and Peters model for practical application.

Among the solutions to solve this issue that were experimented, one was to consider the last known step to be the step required, and another was to extrapolate the time values of the coefficients. However no truly convincing fix were found for this problem, and it was chosen to take the downstream velocity as null in these cases.

Considering now the dynamic behaviour of the wake, no adaptation or extension, as proposed by Huang or Zhao, is considered. This means that a change in skew wake angle during a simulation would give what is represented on Fig 4.1.2. It represents schematically on the left a green wake for an axial flight condition, and on the right, in green, what would be expected if the wake skew angle suddenly change at the next time step, and, in red, what happens in the current version of the model. Improving the depiction of the dynamics of the wake would thus not only require to account for the non linear extension designed for the wake curvature, but also to modify the streamlines shapes accordingly. In flight dynamic simulation, any variation of the speeds in translation or rotation will cause a rotor wake deformation that will not be well captured by this too rough representation of the rotor wake with rectilinear streamlines varying in block quasi-steadily.

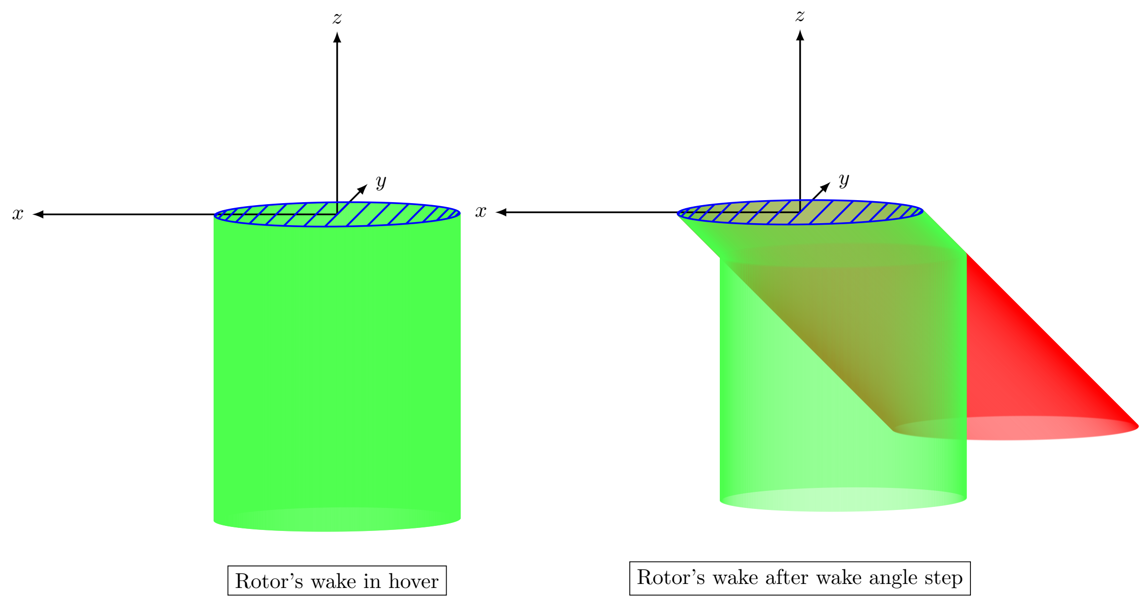


Figure 4.1.2: Schematic representation of the wake behaviour for a change in wake skew angle.

4.1.5 Results

The implementation of the Huang and Peters model in AMB allows to compare it to various types of models, such as a prescribed wake model and a free wake model.

4.1.5.1 Comparison to prescribed and free wake methods

Prescribed and free wake methods presentation The prescribed and free wake models implemented in AMB generate vortex sheets attached to each blade lifting line accounting for trailing vortices. The shed vortices due to the azimuthal variation of the circulation on the blades are neglected in a first approximation for reducing the computational time as AMB is dedicated to presizing. Furthermore, in the cases of helicopter rotors, the shed vortices have a smaller impact than the trailing vortices. It then uses the Biot and Savart induction law in order to compute the value of the induced velocity at the required points. In the case of the prescribed model the vortices are all only convected with the free stream velocity. With the free wake model however, the induced velocities generated by the wake also have an impact on the shape of the wake itself. The induced velocities of other vortices are accounted for on each vortex element. It makes the free wake model computationally more expensive but allows a better representation of the wake geometry and thus of the induced velocities.

Comparison The comparison are made using the 7A rotor, represented in Fig 4.1.3, which is a two bladed rotor made by Onera for wind tunnel measurements. Its main characteristics are presented in table 4.1.1 The simulation is performed over 5 rotor's rotations in order for the wake to develop, and in ascending axial flight condition, with $V_\infty = 10m/s$. The blade pitch angle setting is a collective angle of $10°$ and no flap or lead lag angle are considered. Therefore, the movement of the rotor is prescribed and the induced velocities have no impact on it, other than modifying the airspeed seen by each blade element.

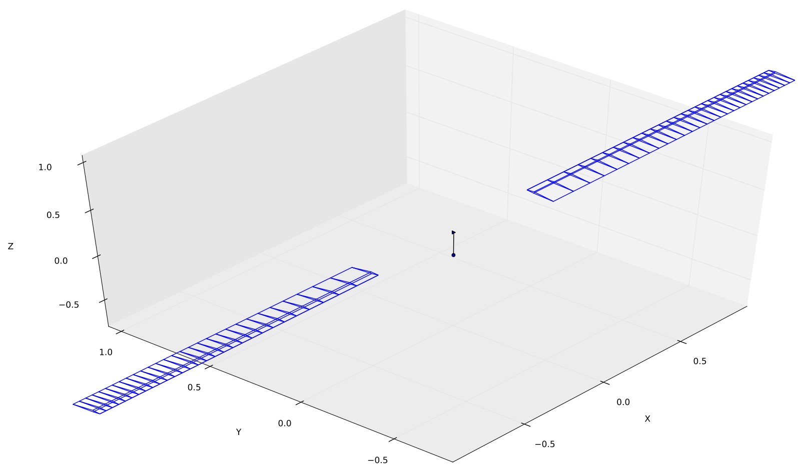


Figure 4.1.3: Representation of the 7A rotor in AMB.

For the parameters of the model, we use $m_{odd} = 6$ and $m_{even} = 4$, and we make the time step small enough to ensure the convergence of the time marching scheme. It is chosen

Radius R	2.1 m
Blade mean chord	0.14m
Rotational Speed Ω	1020 rpm
Blade root radius	0.425
Airfoils used along the span	
0.425m $\mapsto$ 1.575m	OA213
1.575m $\mapsto$ 2.1m	OA209
Linear twist angle	
0.425m	$0°$
1.575m	$-4.545°$
1.89m	$-3.490°$
2.1m	$-4.320°$

Table 4.1.1: Characteristics of the 7A rotor.

to be of 1° of rotation, which makes the non dimensional time step effectively seen by the model of $\frac{\pi}{180}$.

We compare in Fig 4.1.4, 4.1.5 and 4.1.6 the axial, radial and orthoradial induced velocities on the blades respectively and the forces on Fig 4.1.7, for all three models considered here. One can see that the agreement for the axial velocity is satisfying. The behaviour at the tip of the blade could be improved but the general trend is respected, which also gives a good representation of the forces on the blades. However, the representations of the radial and orthoradial velocities are less satisfying. Especially the models exhibit opposite trends on the tangential velocity. Although the two vortex wake models do not include shed vortices, their trends agrees with the expected swirl values. It is possible that the fact that the Huang and Peters model only considers the lift force, in the form of a pressure gradient is detrimental to the accurate representation of the orthoradial velocity. Nevertheless, further comparisons are still required before drawing a definitive conclusions.

Fig 4.1.8 and 4.1.9 compare the velocities generated by the rotor in the plane defined by $y = 0$ for the same case with Huang and Peters model and the free wake model respectively. There are a few points that caught our attention. First, the wake is developed in a coherent way in the case of Huang and Peters model although not all the details of the wake are rendered. There is no wake contraction, since the model used is the Huang and Peters model without non linear extensions. Then there is a visible representation of the passage of the blades in the wake, although the blade tip vortices are not rendered with as much precision as with the free wake model. Finally, the free wake model experiences blade root vortices that are due to the way one models the blades. There is indeed no rotor head or hub represented in the model. This behaviour is smoothed in the Huang and Peters model.

It is to be noted that finer description of the velocities were researched, but required a much smaller time step in order to converge. A case with $m_{odd} = 8$ and $m_{even} = 6$ required a non dimensional time step of $0.2\frac{\pi}{180}$ in order to converge. No significant improvement can be seen on the components in the plane of the rotor, and the improvement on the axial velocity is marginal.

A case with $m_{odd} = 12$ and $m_{even} = 8$ was tested, but required a non dimensional time step of less than $0.01\frac{\pi}{180}$, which would have required 100 steps to represent 1° of rotation, and was therefore not carried out. It is to be noted that the lack of stability is surely linked to the temporal scheme used here which is conditionally stable.

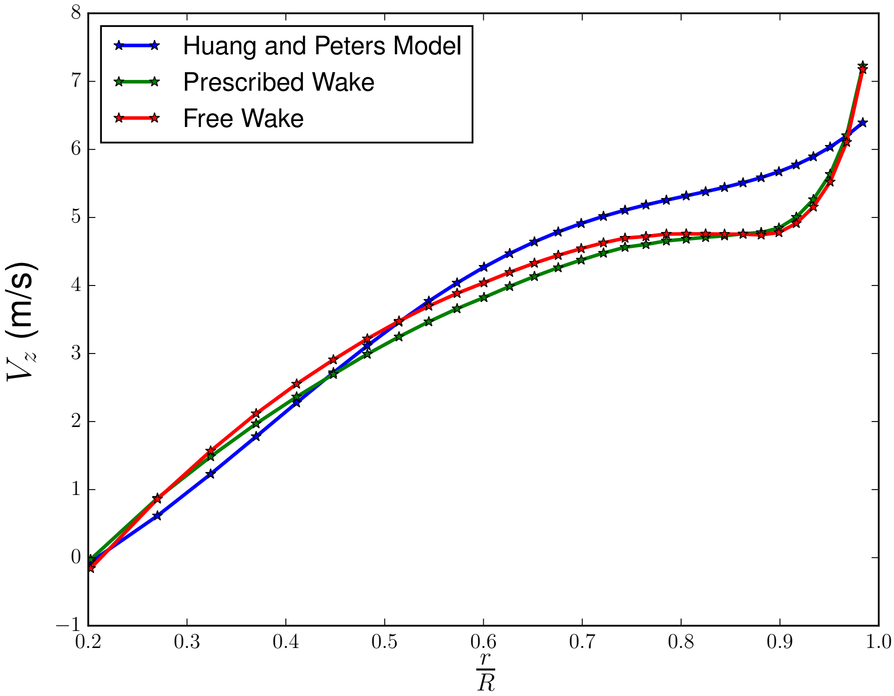


Figure 4.1.4: Comparisons on the blades of the axial velocity given by Huang and Peters model, and prescribed and free wake models, after 5 rotations.

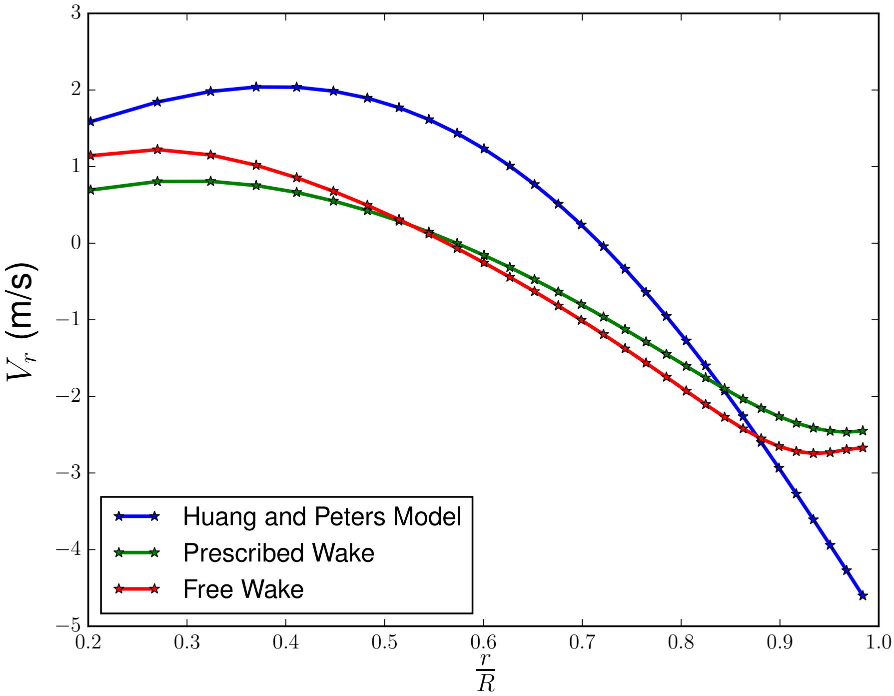


Figure 4.1.5: Comparisons on the blades of the radial velocity given by Huang and Peters model, and prescribed and free wake models, after 5 rotations.

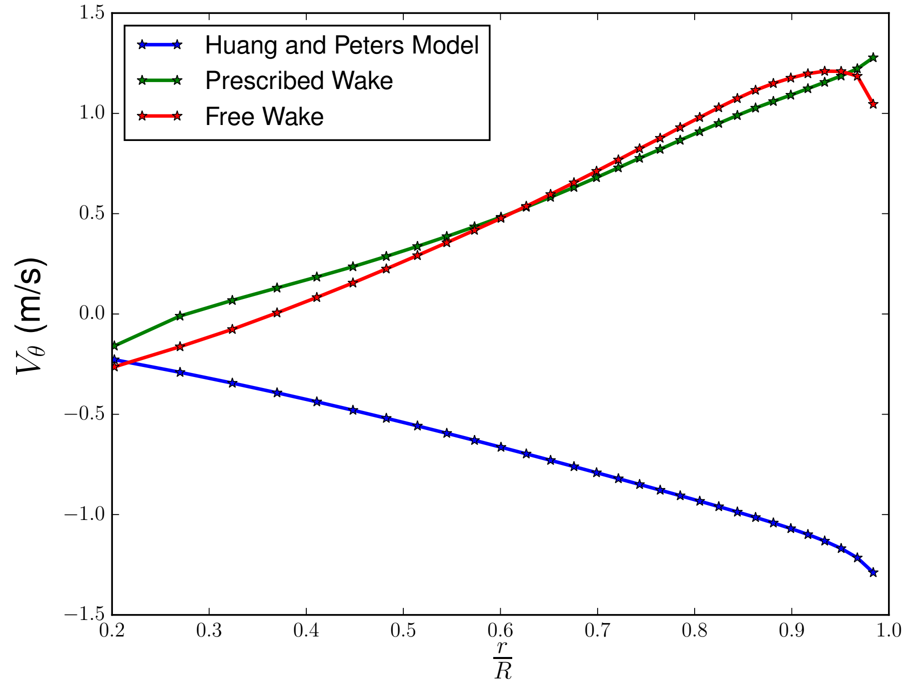


Figure 4.1.6: Comparisons on the blades of the tangential velocity given by Huang and Peters model, and prescribed and free wake models, after 5 rotations.

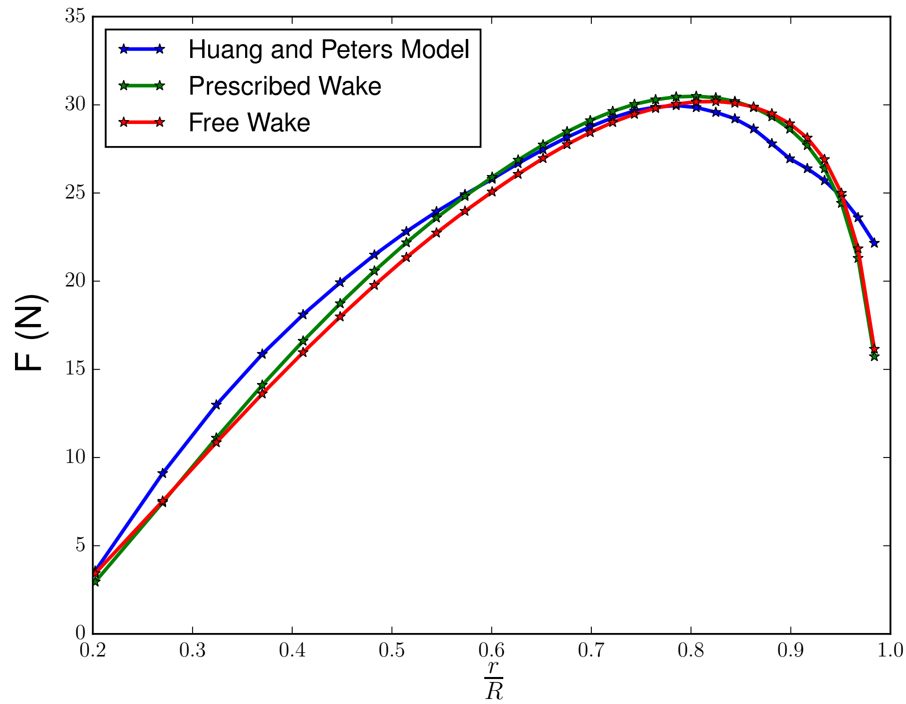


Figure 4.1.7: Comparisons on the blades of the lift distribution given by Huang and Peters model, and prescribed and free wake models, after 5 rotations.

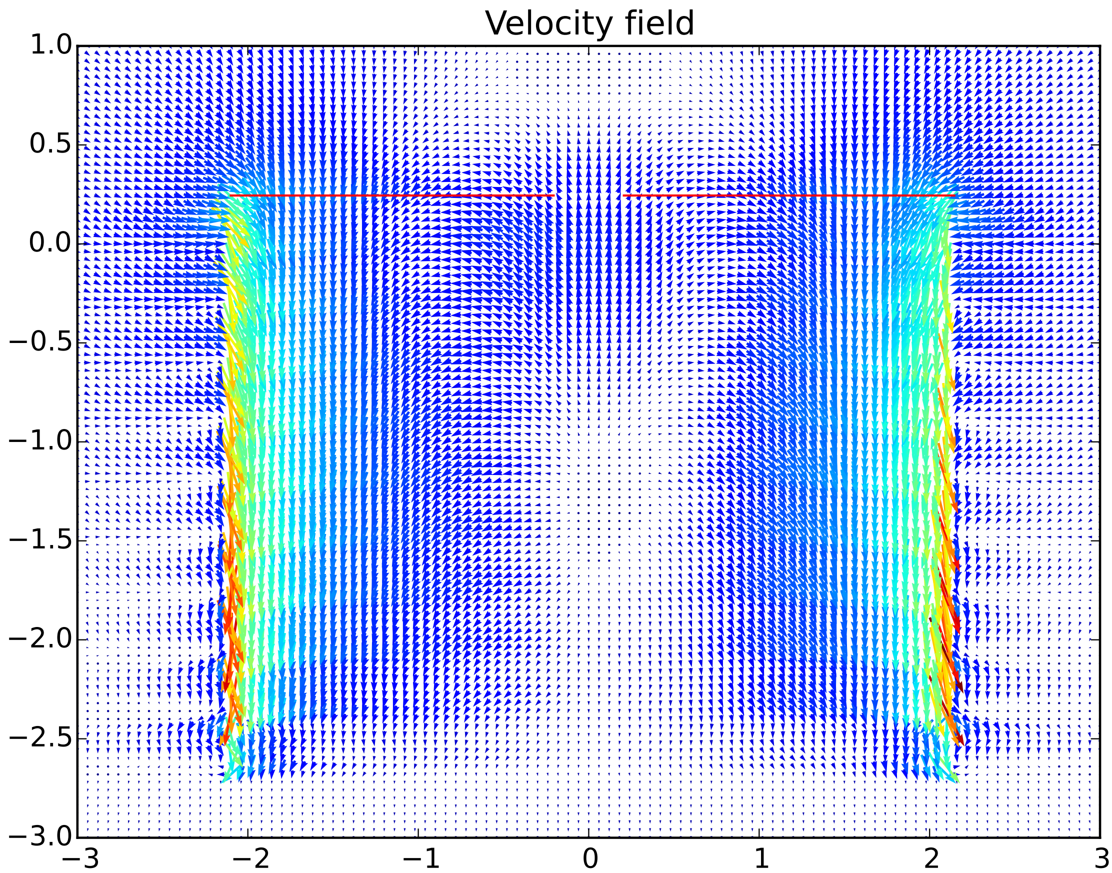


Figure 4.1.8: Wake given by Huang and Peters model, after 5 rotations.

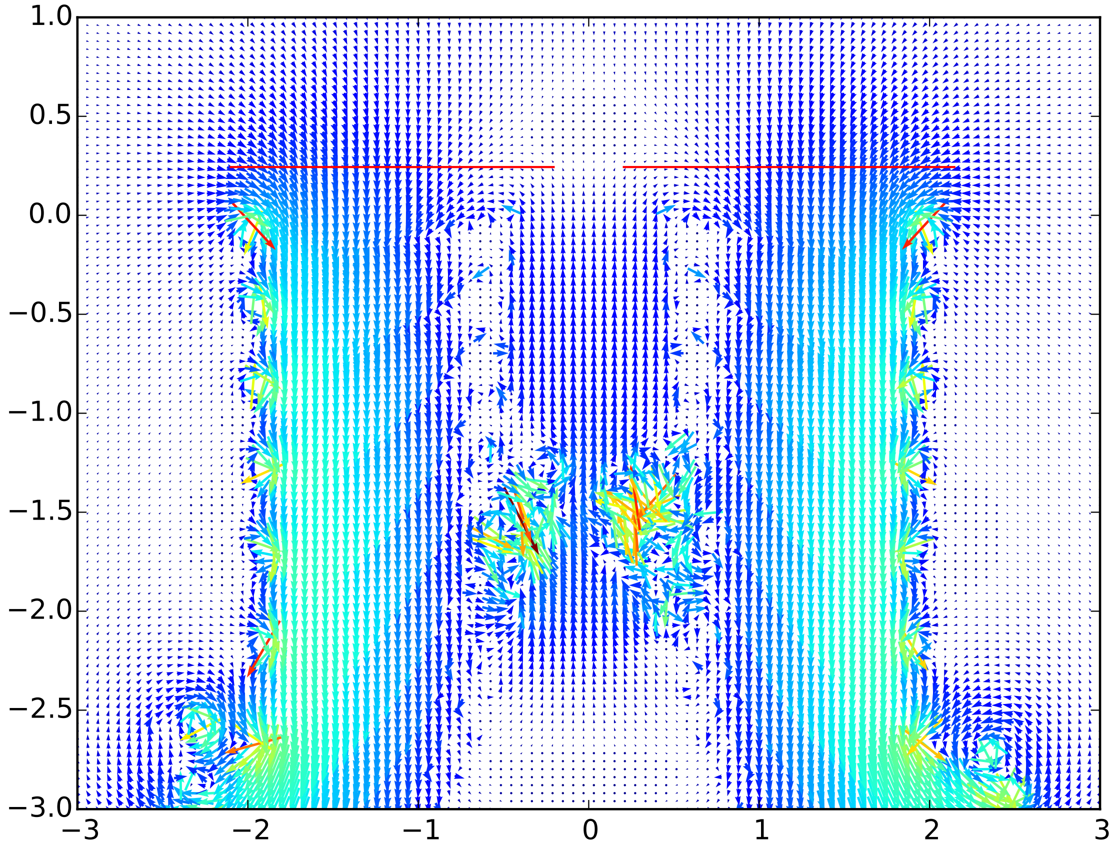


Figure 4.1.9: Wake given by the free wake models, after 5 rotations.

4.1.6 Conclusion

The Huang and Peters model agrees in a limited range with the prescribed and free wake models. The agreement is rather good for the axial velocity, but for the radial component, the trend is correct, whereas for the tangential component, the Huang and Peters model gives an opposite trend compared with the vortex wake models results. If we assume that the coupling was done in an acceptable manner[2], we may have come across a shortcoming of the new method. The fact that the radial and tangential velocities are not satisfyingly represented could be a consequence of the form of the inputs. Indeed, the Huang and Peters model sees the inputs, that are rotor forces, as a discontinuity of pressure. Therefore it only accounts for the lift generated by the rotor (as all the Peters et al. models), but the gradient of the pressure has a direct impact on all the components of the velocity. It could be interesting to account for the drag, and to reduce the impact of the pressure on the other components, or to find another way to account for the forces generated by the blades.

However, the fact those discrepancies might be due to a mistake in our time domain implementation of the model prevents us to draw any definitive conclusion on the subject. An interesting paper, [66], was published about a coupling of the Huang and Peters model with a blade element model, but does not show any result concerning the two problematic components of the induced velocity (only the axial velocity is considered in this paper).

Furthermore, a better time marching scheme could be useful to improve the convergence of the model and test in a meaningful way a more precise approximation. Finally, besides the accuracy, an important point of comparison is the computational time. It should be underlined that the Huang and Peters model is faster than the vortex models it is compared with here. The comparison is presented in table 4.1.2. It shows that the computational cost of the Huang and Peters model is one order of magnitude lower than the one of the prescribed wake model, and 3 orders of magnitude lower than the one of the free wake model. This is a non negligible advantage of this model.

Method	*Computation time (min)*
Prescribed Wake	0.01
Free wake	2.5
Huang and Peters: $m_{odd} = 6$ and $m_{even} = 4$	2.78E-3
Huang and Peters: $m_{odd} = 8$ and $m_{even} = 6$	4.305E-3
Huang and Peters: $m_{odd} = 12$ and $m_{even} = 8$	8.33E-3

Table 4.1.2: Mean time taken by the different models for one time iteration.

[2] This has been validated by the verification of the steady state cases done, but there are a lot of other factors that should be accounted for and that could have been overlooked in the implementation of the time domain method presented here.

4.2 Improvements of the model

The aim of this chapter is to present an improvement of Peters and Huang model, and to present a possible modification. More precisely, it aims at improving the Morillo-Duffy model used within the Huang and Peters model in order to reduce the number of matrix inversions done in the model, and to improve its efficiency for varying wake angles. The possible modification concerns the use of another method to compute the associated Legendre functions of second kind for large values.

4.2.1 Less inversion

4.2.1.1 Principle

In the general case, the linearised Euler's equation can be put under the following form:

$$\frac{\partial \overrightarrow{\nu}}{\partial \tau} + \cos(\chi)\frac{\partial \overrightarrow{\nu}}{\partial z} - \sin(\chi)\frac{\partial \overrightarrow{\nu}}{\partial x} = \overrightarrow{\text{grad}}p \tag{4.2.1}$$

Where we have simply expressed the derivatives along ξ, the curvilinear coordinate along the streamline, with the wake skew angle χ.

The usual Galerkin method, as described in section 3.2.4, is given as:

$$\begin{aligned}\sum_{m,n} \iiint_D \frac{\partial a_n^m}{\partial t}\overrightarrow{\text{grad}}\Psi_n^m.\overrightarrow{\text{grad}}\Lambda_j^r + a_n^m\overrightarrow{\text{grad}}(\overrightarrow{\text{grad}}\Psi_n^m.\overrightarrow{\xi}).\overrightarrow{\text{grad}}\Lambda_j^r dD \\ = \sum_{m,n}\iiint_D \tau_n^m\overrightarrow{\text{grad}}\Phi_n^m.\overrightarrow{\text{grad}}\Lambda_j^r dD\end{aligned} \tag{4.2.2}$$

This then gives (considering only the cosine part), when applying the gradient theorem, and by applying the z derivatives to the weight functions:

$$\begin{aligned}\sum_{m,n} \frac{\partial a_n^m}{\partial t}\iint_S \Psi_n^m\frac{\partial \Lambda_j^r}{\partial z}dS + a_n^m\iint_S\left(\cos(\chi)\frac{\partial \Psi_n^m}{\partial z} - \sin(\chi)\frac{\partial \Psi_n^m}{\partial x}\right)\frac{\partial \Lambda_j^r}{\partial z}dS \\ = \sum_{m,n}\tau_n^m\iint_S \Phi_n^m\frac{\partial \Lambda_j^r}{\partial z}dS\end{aligned} \tag{4.2.3}$$

Here the choices of velocity potential and of test functions is still open, and one can still move the z derivatives on the term of its choice, thanks to the gradient theorem, as long as the velocity potential is a potential function. One can thus put the equations under the following form:

$$A_1\frac{d}{dt}(\{a_n^m\}) + (\cos(\chi)D_1 - \sin(\chi)S_1)\{a_n^m\} = M\{\tau_n^m\} \tag{4.2.4}$$

As one can see, this expression does not rely on an inversion of any matrices (at least before solving the equation). Furthermore the dependency on the χ angle is explicit, meaning that changing the wake skew angle does not require to compute a new matrix.

The game is now to make the correct choices for our three degrees of freedom (Ψ_n^m, Λ_j^r, and placement of the z derivative), so that we can restrict the integrals to the rotor disc. A second point that should be kept in mind when making these choices is the conditioning of the matrices. Most attention should be given to the matrix generated by the x derivative terms, S_1, which tends to be close to singular. Since the matrix to be inverted is in fact

$i\omega A_1 + \cos(\chi)D_1 - \sin(\chi)S_1$ in the frequency domain, the conditioning of S_1 tends to be less of a problem, as long as the wake angle is low enough so that D_1 compensates it.

4.2.1.2 Application

In this section we show the choices we have made for our degree of freedom and explain why these choices were made. We refer the reader to appendix D, for the full expression of the matrices, and the treatment of all the singular cases.

We take the following form of the velocity potential and of the test function:

$$\Psi_n^m = \sigma_n^m \Phi_{n+1}^m + \varsigma_n^m \Phi_{n-1}^m + \sigma_{n+1}^{m+1} \Phi_{n+2}^{m+1} + \varsigma_{n+1}^{m+1} \Phi_n^{m+1} \tag{4.2.5}$$

and

$$\Lambda_j^r = \sigma_j^r \Phi_{j+1}^r + \varsigma_j^r \Phi_{j-1}^r \tag{4.2.6}$$

First, the use of combinations of potential functions ensure that we can use the gradient theorem as intended. Then, the z derivatives of each element is easily expressed with the Φ_n^m elements, as shown in [19]. Finally, one can notice that the integrals can be reduced to the rotor disc if at least one of the term below the integral is odd in ν (as remarked in [19]). Keeping in mind that the x derivative does also change the parity, one can change the position of the z derivatives in order to reduce the integrals to the rotor disc.

The choice of the form of the velocity potential is motivated by its capacity to improve the condition number of the S_1 matrix. Indeed, the x derivative tends to generate singular matrices because the image of the Φ_n^m basis by this derivative is expressed with the $\frac{P_n^m}{\nu}$. The idea of the Ψ_n^m is here to combine various terms so that their x derivative can be expressed on the P_n^m basis. The full derivation of the S_1 matrix is presented in appendix D.

This therefore gives a new formulation of the Morillo-Duffy model, that does not require to inverse the L matrix, and that shows explicitly the dependency on χ, which means that there is no need to compute all the terms in L when changing the value of χ.

4.2.1.3 Results

We show here a few results illustrating the capacity of the method to reproduce results of the Morillo-Duffy model.

One can observe that for the given case, which uses $m_{odd} = 12$ and $m_{even} = 8$, both models are in pretty good agreement with the exact solution. This tends to validate the improvement made. Furthermore, since this improvement is based on the same assumptions and concept as the Morillo-Duffy model, it could be integrated to the Huang-Peters model in order to improve its performances in the case of an evolution of χ. The computational time on a single case would not be affected, but the variation of the value of χ could be instantaneously accounted for. This is even more impacting with a large number of element, since the lower theoretical bound on the complexity would be at least $O(n^2 ln(n))$, where n is the size of the matrix, and O the big O Landau notation.

4.2.1.4 Limitations

Although this new formulation gives interesting results, it still struggles to represent high advance ratio cases. Indeed, despite the use of a good formulation of the velocity potential, the condition number of the S_1 matrix is still too high to give physical results when considering a high wake angle. This is linked to the fact that, since the wake is in the rotor plane at high advance ratio (or perfectly edgewise flow case), there is no way to represent

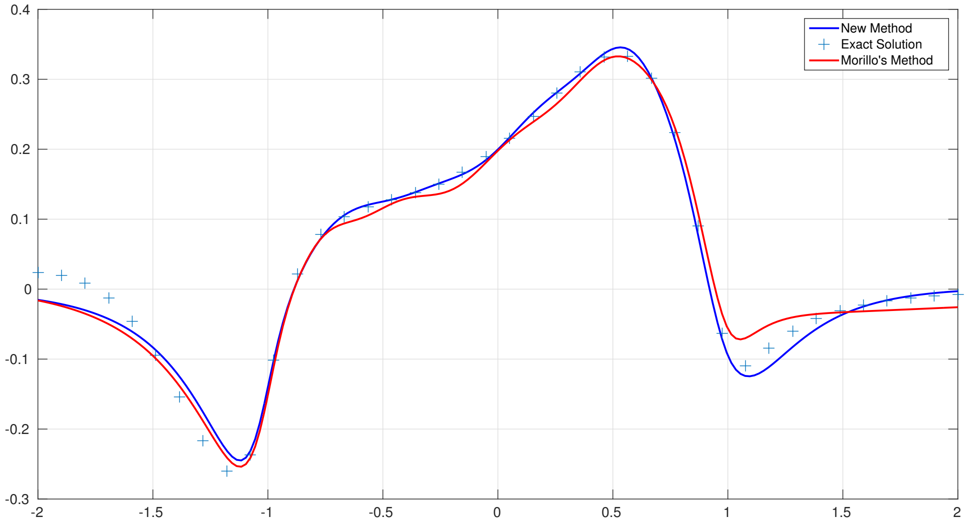


Figure 4.2.1: Comparison of the axial velocity with Morillo-Duffy model: real part

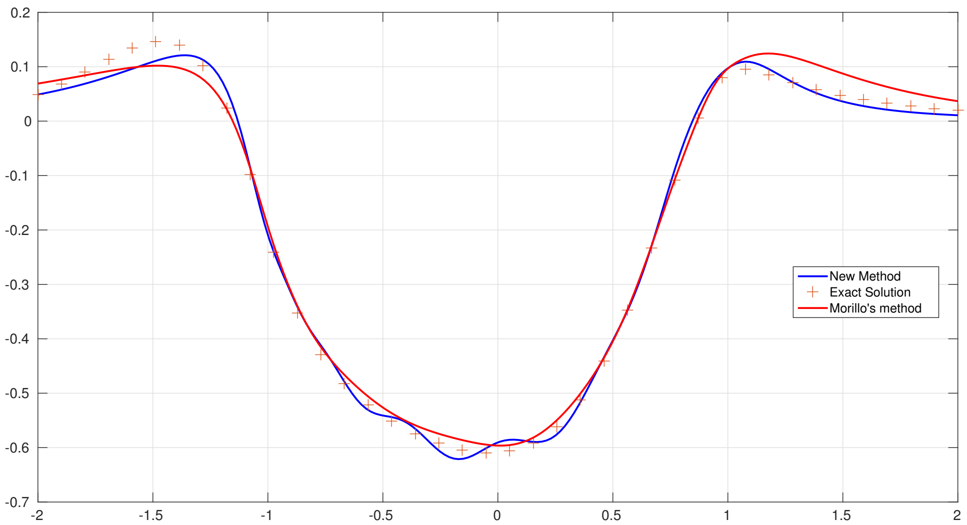


Figure 4.2.2: Comparison of the axial velocity with Morillo-Duffy model: imaginary part

it accurately with the chosen potentials which all converge to 0. This is also a limitation of the Morillo-Duffy model, which points the way to a better choice of velocity potentials that could be made in order to have a method that avoids inversion on the whole range of χ. They may be found in the other solutions to the Laplace equations, i.e. the associated Legendre polynomials of first and second kind, with less constraints on n and m.

4.2.2 Large values of the associated Legendre functions of second kind

As mentioned by Fei in [24], the computation of the values of the associated Legendre polynomials of the second kind, $Q_n^m(\eta)$ for large values of η is difficult. Indeed the use of the recurrence formula to compute them is problematic, and tends to diverge. Large values here represents values above $\eta = 1$ for polynomials of high order ($m > 10$) and above $\eta = 1.5$ to $\eta = 2$ for lover values of m.

Fei proposed a method to smooth the values of the Q_n^m, which consists in approximating them with a $\frac{1}{\eta^{2q}}$ development. However, this method does not verify the regularity of the approximation at the transition. We here propose a similar approach, but we pay special attention to the transition between the recursively computed solution and the approximation.

4.2.2.1 Principle

We begin with the same hypothesis as Fei, which is that for large enough values of η, Q_n^m can be approximated by:

$$Q_n^m(\eta) = \sum_{k=0}^{p} \frac{A_{2k}}{\eta^{q+2k}} \tag{4.2.7}$$

Using the fact, demonstrated by Fei, that $q = n + 1$, one can then set two conditions: one for the continuity of the approximation at the change, and one for the continuity of its derivative. Naming η_{max} the value of change between the two methods, it gives:

$$\begin{aligned} Q_n^m(\eta_{max}) &= \sum_{k=0}^{p} \frac{A_{2k}}{\eta_{max}^{n+2k+1}} \\ \frac{\partial Q_n^m}{\partial \eta}(\eta_{max}) &= \sum_{k=0}^{p} -(n+2k+1)\frac{A_k}{\eta_{max}^{n+2k+2}} \end{aligned} \tag{4.2.8}$$

In order to determine the rest of the A_k coefficients, one can add $p - 2$ equations to the system with the same method as Fei, which consists in injecting the form chosen for the Q_n^m into the differential equation verified by the Q_n^m.

This gives a matrix equation, with an inversion to do in order to calculate the A_k.

4.2.2.2 Comparison

Figure 4.2.3 and Figure 4.2.4 compare three various methods for the computation of Q_2^0 and Q_{10}^0: the recursive one, Fei's method, and the proposed method. The value of η_{max} is set to 1, and we observe the behaviour of all the methods around the value of η_{max}. The recurrence formula being exacte for the cases chosen, it will our reference here.

One can see that the proposed method follows the recursive method with more accuracy than Fei's method at the truncation. Indeed, the rest of the domain is accurately represented.

It is to be noticed that the scale on those figures are small, and that the effect of the improvement is marginal in the case of a single polynomial, and even less important with higher orders. However, the sum of a lot of those polynomials, as is done in the computation of the velocities, leads to visible effects on the results.

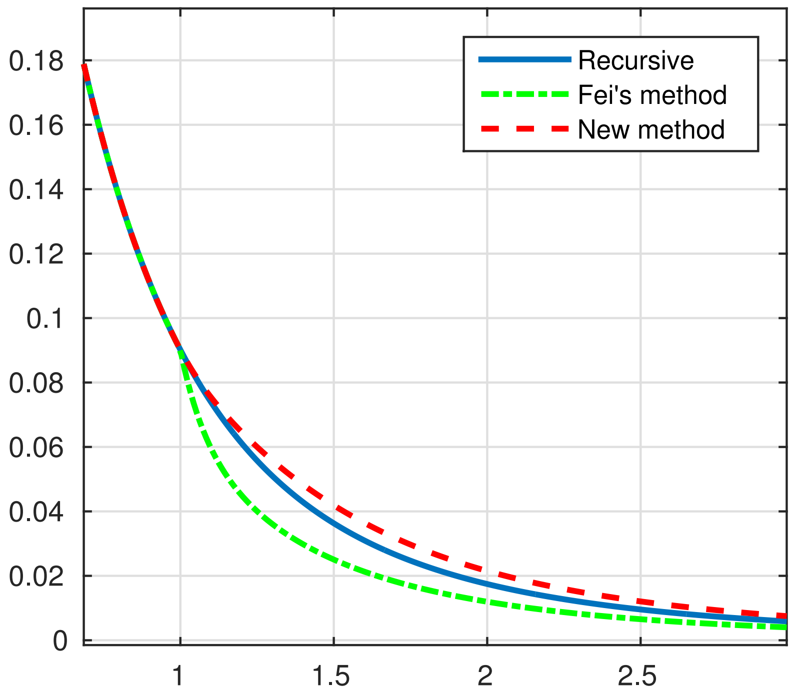


Figure 4.2.3: Comparison of the three method of computation of Q_2^0, with $\eta_{max} = 1$

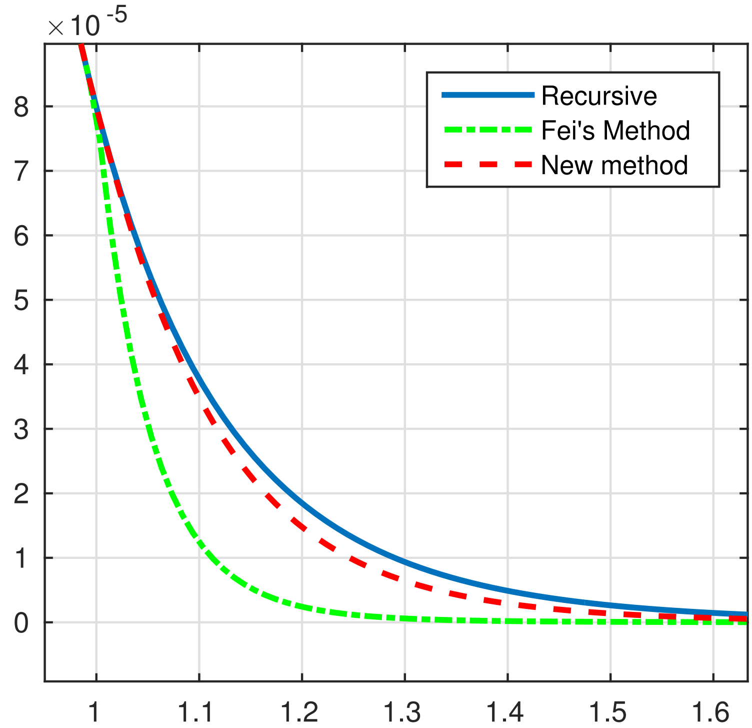


Figure 4.2.4: Comparison of the three method of computation of Q_{10}^0, with $\eta_{max} = 1$

4.3 Failed attempts to improve the model

4.3.1 Improving the exact solution

4.3.1.1 Motivation

The main last advancement in the models of Peters et al. were made by accessing the values of the velocities below the rotor. The expression of the velocity below the rotor is mainly based on the exact solutions developed analytically in the frequency domain. As mentioned before, this exact solution does not account for the continuity equation and assumes straight streamlines.

Thus a method improving the said "exact solution" could lead to a different, but more comprehensive adjoint method. Furthermore, the modification of the straight streamline could later lead to a way to account for wake deformation. We here lay the beginning of a reflection that seemed promising, but that never yield any results.

4.3.1.2 Method

In order to improve the integral form of the induced velocity, a new form of the streamlines will be sought, so that the integral form does not differ, and so that the continuity equation is respected. I.e. we will seek to find a path so that the shape of the streamline makes the exact solution respect the continuity equation.

The streamline is therefore described by the following parametric curve:

$$\Gamma : \overrightarrow{\gamma}(\xi, x_0, y_0, \chi) = \begin{cases} x = & f(\xi, x_0, y_0, \chi) \\ y = & g(\xi, x_0, y_0, \chi) \\ z = & h(\xi, x_0, y_0, \chi) \end{cases} \tag{4.3.1}$$

where ξ is the parameter of the curve, x_0 and y_0 represent the coordinate at which the streamline crosses the rotor disc plane, and f, g, and h are $\mathscr{C}^1$ functions on $\mathbb{R}$. The other dependencies will be omitted for clarity.

This new chosen path does in fact influence not only the exact solution but also the way the equations used for this theory were derived. Indeed, the second term of the conservation of the momentum equation depends on it:

$$\overrightarrow{V}.\overrightarrow{\text{grad}}\overrightarrow{V} = (V_\infty \overrightarrow{\gamma} + \overrightarrow{\delta v}).\overrightarrow{\text{grad}}\left(V_\infty \overrightarrow{\gamma} + \overrightarrow{\delta v}\right) \tag{4.3.2}$$

Thus by assuming both that $V_\infty \overrightarrow{\gamma} >> \overrightarrow{\delta v}$ and that $\overrightarrow{\text{grad}}\left(\overrightarrow{\delta v}\right) >> \overrightarrow{\text{grad}}(V_\infty \overrightarrow{\gamma})$, one can find the same equations as used in the theory developed by Huang and Peters.

Furthermore using the complex notation used to derive the exact solution, see section 3.3.1:

$$\text{div}\,(\overrightarrow{v}) = 0 \Leftrightarrow \begin{cases} \text{div}\,(\overrightarrow{u}) = 0 \\ \text{div}\,(\overrightarrow{w}) = 0 \end{cases} \tag{4.3.3}$$

Considering only the term $\frac{\partial u_x}{\partial x}$ of the divergence, and applying Leibniz's rule, we have:

$$\begin{aligned}\frac{\partial u_x}{\partial x} &= \frac{\partial}{\partial x}\left(\int_{-\infty}^{\xi_0} \cos(\omega(\xi_0-\xi))\frac{\partial P}{\partial x}d\xi\right)\\ &= \frac{\partial P}{\partial x}\frac{\partial \xi_0}{\partial x} + \int_{-\infty}^{\xi_0} -\omega\sin(\omega(\xi_0-\xi))\frac{\partial P}{\partial x}\frac{\partial(\xi_0-\xi)}{\partial x} + \cos(\omega(\xi_0-\xi))\frac{\partial^2 P}{\partial x^2}d\xi\end{aligned} \tag{4.3.4}$$

Hence, by considering the real part equation, and using the assumption that the pressure respect the Laplace equation:

$$\begin{aligned}\operatorname{div}(\overrightarrow{u}) &= 0\\ \Leftrightarrow \frac{\partial P}{\partial x}\frac{\partial \xi_0}{\partial x} + \frac{\partial P}{\partial y}\frac{\partial \xi_0}{\partial y} + \frac{\partial P}{\partial z}\frac{\partial \xi_0}{\partial z} &\\ + \int_{-\infty}^{\xi_0} -\omega\sin(\omega(\xi_0-\xi))\left(\frac{\partial P}{\partial x}\frac{\partial(\xi_0-\xi)}{\partial x} + \frac{\partial P}{\partial y}\frac{\partial(\xi_0-\xi)}{\partial y} + \frac{\partial P}{\partial z}\frac{\partial(\xi_0-\xi)}{\partial z}\right)d\xi &= 0\end{aligned}$$

The imaginary part of the divergence equation gives similar results with the exception of the first term of the previous equation. Speaking of this first term, taking $\omega = 0$, it is the only term remaining in the equations. This means that to respect the continuity equation, this term should be null. However, this term could be seen as:

$$\frac{\partial P}{\partial x}\frac{\partial \xi_0}{\partial x} + \frac{\partial P}{\partial y}\frac{\partial \xi_0}{\partial y} + \frac{\partial P}{\partial z}\frac{\partial \xi_0}{\partial z} = \langle \overrightarrow{\mathrm{grad}} P | \overrightarrow{v}_\Gamma \rangle = 0 \tag{4.3.5}$$

That is to say as the scalar product between the gradient of the velocity and the tangent to the parametric curve defined by Γ. This can be further interpreted as the fact that the stream line should be perpendicular to the pressure isolines in order to respect the continuity equation. This is normally the kind of relation obtained for the equipotential for potential flows.

The derivation of the parametric curve can be expressed through the derivative of the functions f, g and h by assuming they are bijections. However, this leads to highly involved equation that could not be solved. Therefore, in a first time and in order to simplify the method, an axisymmetric case is considered, with $\chi = 0$, allowing to make a more simple and manageable assumption on the shape of the streamline:

$$\Gamma_{\chi=0} : \left\{\begin{aligned} x =&\ r(\xi)\cos(\theta)\\ y =&\ r(\xi)\sin(\theta)\\ z =&\ \xi\end{aligned}\right. \tag{4.3.6}$$

The aim here would be to deduce the function r from the continuity equation. To do so we inject the form of the streamline in the previous equations. It is to be noted that since the r function is assumed to be a bijection, the derivative of its inverse can be expressed as the inverse of its derivative. We here show a rapid derivation for a case where the streamline is in the plane $y = 0$:

$$\begin{aligned} \operatorname{div}(\overrightarrow{u}) &= 0 \\ \Leftrightarrow \frac{\partial P}{\partial x}\frac{1}{r'(x)} + \frac{\partial P}{\partial z} + \int_{-\infty}^{\xi_0} -\omega\sin(\omega(\xi_0-\xi))\left(\frac{\partial P}{\partial x}\left(\frac{1}{r'(x)}-\xi\right)+\frac{\partial P}{\partial z}(1-\xi)\right)d\xi &= 0 \end{aligned}$$

Unfortunately, no analytic solution for the form of r could be found. Although this could be solved numerically while advancing the streamline through the predetermined pressure gradient, this is far too computationally expensive for the kind of method we wish to develop.

This development gave us insight on the strength of the straight streamline assumption and on the difficulty to modify this assumption in the case of the exact solution. There is thus little doubt on the consequence this change would have on the adjoint method derivation.

4.3.2 Freeing the model from the yoke of the velocity potential

4.3.2.1 Motivation

Imposing the hypothesis of a velocity potential creates a certain number of problems. Most notably, because the potential functions all converges to zero at infinity, it prevents the model to replicate the perfectly edgewise flow, as mentioned by Morillo [19]. Indeed the perfectly edgewise flow implies a non null velocity in the rotor disc plane, which prevents its description by the given velocity potential. Furthermore, the potential assumption creates an asymmetry between the treatment of the domain above the rotor and the domain below the rotor. It does, however, allow a good representation of the effect of the rotor, since its influence should only be localised on the disc. This is the origin of the need for an alternative form of the velocity, provided by the downstream velocity in the case of the Huang and Peters model. However, this velocity creates some unwanted problems in the time domain, as mentioned earlier. A second downside of this solution is that it does not respect the continuity equation. Several solutions were thought of to improve the behaviour or simply get rid of the problems created by this velocity. It remains that a good way to solve this problem would simply be to avoid the velocity potential assumption, while keeping the spirit of the method, which does give satisfying results in many cases.

It is important to note that without the hypothesis of a velocity potential, there is no longer a link between the velocity and the Laplace equation. However one can show that the pressure still respects the Laplace equation, if we linearise the convective terms. This will force us to rework the equations from the beginning in order to cope with what is at our disposal. This approach lays the foundation of the induced velocity model developed in chapter 5.

4.3.2.2 First attempt: separating all components

We go back to the linearised Euler's equations:

$$\frac{\partial\overrightarrow{\nu}}{\partial t} + \overrightarrow{\overrightarrow{\operatorname{grad}}}(\overrightarrow{\nu})\,.\,\overrightarrow{\xi} = -\overrightarrow{\nabla}p$$
$$\operatorname{div}(\overrightarrow{\nu}) = 0$$

However, we now assume a form for the three components of the velocity, as well as for the pressure. This gives a lot of various options to apply the Galerkin method. Indeed one can choose the function basis, the test functions, and the scalar product to apply for each of the 3 velocity equations and for the continuity equation separately.

Two ways of making the numerous choices at hand are considered. The first one is to assume a form of the velocity and pressure so as to deduce weight function and a scalar product behaving in a good manner in order to compute closed form values. The second method consists in selecting a scalar product to then form an orthogonal (or even orthonormal) basis of polynomials, which will then be used to describe the velocity and the pressure and to pray for them to behave nicely in the equations.

The main problem one is confronted to when considering these options is the fact that the derivatives of the equations tend to introduce terms hard to represent on the same basis.

It is therefore decided to use the first option and to compute numerically the integrals, to store the results and use them when required. The following form of the velocity and pressure are adopted:

$$V_i = \sum_{m=0}^{\infty} \sum_{n_1=m}^{\infty} \sum_{n_2=0}^{\infty} a_{n_1,n_2}^{m,i} \Phi_{n1,n2}^{m} \tag{4.3.7}$$

$$p = \sum_{m=0}^{\infty} \sum_{n_1=m}^{\infty} \sum_{n_2=0}^{\infty} \tau_{n_1,n_2}^{m} \Phi_{n1,n2}^{m} \tag{4.3.8}$$

Where $\Phi_{n1,n2}^{m} = P_{n_1}^{m}(\nu) P_{n_2}^{m}\left(\frac{2\eta}{\eta_{max}} - 1\right)\cos(m\psi)$, i represents any of the three components of the velocity and where η_{max} is the maximal value of η where boundary conditions are set.

The scalar product chosen is then:

$$\langle f, g\rangle = \iiint\limits_{D} f g \, dD \tag{4.3.9}$$

And finally the chosen test functions will be the $\Phi_{j1,j2}^{r}$.

This allows to obtain a new set of equations: 3 coming from the momentum equation, under the following form:

$$+\frac{da_{n_1,n_2}^{m,i}}{dt}\langle \Phi_{n1,n2}^{m}, \Phi_{j1,j2}^{r}\rangle + a_{n_1,n_2}^{m,i}\left\langle \frac{\partial \Phi_{n1,n2}^{m}}{\partial \xi}, \Phi_{j1,j2}^{r}\right\rangle = +\tau_{n_1,n_2}^{m}\left\langle \frac{\partial p}{\partial x_i}, \Phi_{j1,j2}^{r}\right\rangle$$

And one from the continuity equation, as follow:

$$+\left\langle a_{n_1,n_2}^{m,x}\frac{\partial \Phi_{n1,n2}^{m}}{\partial x} + a_{n_1,n_2}^{m,y}\frac{\partial \Phi_{n1,n2}^{m}}{\partial y} + a_{n_1,n_2}^{m,z}\frac{\partial \Phi_{n1,n2}^{m}}{\partial z}, \Phi_{j1,j2}^{r}\right\rangle = 0$$

These equations can be set under the following matrix form, assuming a discretisation of time on frequencies ω:

$$\begin{pmatrix} i\omega A + B & 0 & 0 \\ 0 & i\omega A + B & 0 \\ 0 & 0 & i\omega A + B \\ W_1 & W_1 & W_1 \end{pmatrix} \begin{pmatrix} a_{n_1,n_2}^{m,x} \\ a_{n_1,n_2}^{m,y} \\ a_{n_1,n_2}^{m,z} \end{pmatrix} = \begin{pmatrix} M_{\tau,x} & 0 & 0 \\ 0 & M_{\tau,y} & 0 \\ 0 & 0 & M_{\tau,z} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \tau_{n_1,n_2}^{m} \\ \tau_{n_1,n_2}^{m} \\ \tau_{n_1,n_2}^{m} \end{pmatrix} \tag{4.3.10}$$

This allows to solve for the values of the $a^{m,i}_{n_1,n_2}$ by using the pseudo or generalised inverse.

However, it proved difficult to obtain significant results thanks to this method. Indeed, the integral over the whole domain are not that easy to compute, and the pseudo inverse gives only a least square method solution, which might lead to inaccurate results, on top of its heavy computational cost.

Note on the 3D integrals In order to simplify the derivation and to make the development more similar to the one made by Peters et al., we seek to reduce the integrals used to surface integrals. It is in fact possible to do it for most of the terms encountered, since:

$$\operatorname{div}\left(\Lambda\overrightarrow{V}\right) = \Lambda\operatorname{div}\left(\overrightarrow{V}\right) + \overrightarrow{V}.\overrightarrow{\operatorname{grad}}\Lambda \tag{4.3.11}$$

where Λ represents the test function and $\overrightarrow{V}$ represents the approximation of the velocity.
This gives, by using the divergence theorem, since $\operatorname{div}\left(\overrightarrow{V}\right) = 0$:

$$\iiint_D \overrightarrow{V}.\overrightarrow{\operatorname{grad}}\Lambda dD = \iint_S \Lambda\overrightarrow{V}.\overrightarrow{n}\,dS \tag{4.3.12}$$

and:

$$\operatorname{div}\left(\Lambda\overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}\right) = \Lambda\operatorname{div}\left(\overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}\right) + \left(\overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}\right).\overrightarrow{\operatorname{grad}}\Lambda \tag{4.3.13}$$

and since:

$$\operatorname{div}\left(\overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}\right) = \operatorname{div}\left(\overrightarrow{\operatorname{grad}}\overrightarrow{V}^T\right).\overrightarrow{\xi} = \overrightarrow{\operatorname{grad}}\left(\operatorname{div}\overrightarrow{V}\right).\overrightarrow{\xi} = 0 \tag{4.3.14}$$

This gives:

$$\iiint_D \overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}.\overrightarrow{\operatorname{grad}}\Lambda dD = \iint_S \Lambda\left(\overrightarrow{\operatorname{grad}}\overrightarrow{V}.\overrightarrow{\xi}\right).\overrightarrow{n}\,dS \tag{4.3.15}$$

4.3.2.3 Use of a vector potential

Principle The continuity equation is difficult to translate without 3D integrals. One way to simplify this problem is to see the velocity as the rotational of some vector field, since it has no divergence:

$$\exists\overrightarrow{G}, \overrightarrow{V} = \overrightarrow{\operatorname{rot}}\overrightarrow{G} \implies \operatorname{div}\left(\overrightarrow{V}\right) = 0 \tag{4.3.16}$$

We then can choose a field $\overrightarrow{G}$ such that its third component is null (since $\overrightarrow{G}$ is defined short a potential function) which gives the following form of the velocity:

$$\begin{cases} V_x = & -\frac{\partial G_x}{\partial z} \\ V_y = & \frac{\partial G_y}{\partial z} \\ V_z = & \frac{\partial G_y}{\partial x} - \frac{\partial G_x}{\partial y} \end{cases} \tag{4.3.17}$$

One can thus choose any form for $\overrightarrow{G}$, and have the continuity equation respected, and then inject the results in the momentum equation and proceed to the projection on a space of polynomials to determine the coefficients defining $\overrightarrow{G}$.

Going through the use of a rotational field however require this field to be derivable twice, which might be hard to accomplish in ellipsoidal coordinates, because of their inherent singularities. Changing of coordinate system might (if well chosen) highly simplify the derivatives, but also mean that the definition of the discontinuity at the rotor disc will not be as practical.

Cylindrical coordinates Keeping in mind the previous considerations, the idea of seeing the space in cylindrical coordinates is considered. This allows to easily define the rotor disc area, while drastically simplifying the derivatives used. However, it loses the accuracy the ellipsoidal coordinates has on the disc, and treats symmetrically the out of disc and on disc regions. Indeed, in cylindrical, r must render both the areas on the disc and out of the disc. A solution could be to restrain the space and to only be concerned with the space close to the disc. But this raises the concern of the boundary conditions, since, with a small domain, the boundaries would be close to the disc, and it would thus be false to assume a null gradient of the velocity as was the case in Huang and Peters model. And taking a too large radius around the rotor disc would require too many polynomials states to give a meaningful representation, since the detail of the disc would be highly localised.

The choice of the cylindrical coordinates has pros and cons. The good opportunities offered by this coordinate system are:

- The Laplace equation gives an expression to the pressure expressed through known functions (Bessel functions and trigonometric polynomials)
- The expression of the rotational can give a simple and continuous expression to the velocity (avoiding divisions by zero of Bessel functions or their derivatives)

Indeed, in cylindrical coordinates $(\overrightarrow{u_x}, \overrightarrow{u_r}, \overrightarrow{u_\theta})$, the rotational is:

$$\overrightarrow{\mathrm{rot}}\overrightarrow{G} = \begin{pmatrix} \frac{\partial r G_\theta}{r\partial r} - \frac{\partial G_r}{r\partial\theta} \\ \frac{\partial G_x}{r\partial\theta} - \frac{\partial G_\theta}{\partial x} \\ \frac{\partial G_r}{\partial x} - \frac{\partial G_x}{\partial r} \end{pmatrix} \tag{4.3.18}$$

Hence the following expression of the velocity, by choosing $G_\theta = 0$:

$$\left\{ \begin{array}{r} V_x = -\frac{\partial G_r}{r\partial\theta} \\ V_r = \frac{\partial G_x}{r\partial\theta} \\ V_\theta = \frac{\partial G_r}{\partial x} - \frac{\partial G_x}{\partial r} \end{array} \right. \tag{4.3.19}$$

The main disadvantage of this choice of coordinates is the fact that there is no discontinuity directly representing the rotor disc. But this problem could be used to our advantage by redefining the way the inputs of the model are seen, hence avoiding to reduce the inputs to a pressure discontinuity on the disc, but by seeing directly the effects of the blades on the fluid.

Redefining the inputs of the model *"Bereft of disc, work lies ahead."*

A problem arising when using other coordinates than the ellipsoidal coordinates is that the pressure discontinuity is no longer embedded by the coordinates but has now to be created in the middle of the integration domain considered. Some of the possible solution considered are listed here:

- Divide the space in two domains, one above the disc and one below:

One way to include discontinuities in the domain would be to discretise the various zones of discontinuities and to solve each zone individually. This would however introduce boundary conditions on the rotor disc plane, not only in pressure but also in velocity, which would be far less convenient to compute.

- Write the general form of the pressure, then solve for the coefficients respecting the best the discontinuities due to blades

This would probably cause a lot of Gibbs phenomenon to appear and add a step of computation to the model.

- Add a new version of the input, in order to only account for the effect of the blades.

This would be mostly done by applying forces rather than pressure discontinuity to the domain. Although it would probably suffer of the same Gibbs phenomenon problem, It would probably allow a more generic representation of the blades' effects.

- Place vortices to recreate the effects due to blades. Then use Biot and Savart law to translate the inputs in a more meaningful manner for the model.

This method would be similar to free- and prescribed-wake methods by imposing directly velocities on the domain. The main problem here would be to translate efficiently and meaningfully the vortex effects and positions.

COMPUTATION OF THE MATRICES TERMS In the following subsections the author explores various ways to exploit the equations obtained above in cylindrical coordinates.

Using the Bessel functions

In this section, we will assume the following forms of the velocity, pressure and test functions:

$$\overrightarrow{\nu} = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} a_n^m \overrightarrow{\mathrm{rot}}\overrightarrow{G_n^m} \tag{4.3.20}$$

$$p = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} \tau_n^m J_m(nr)\cos(m\theta)\cosh(nz) \tag{4.3.21}$$

$$\Lambda_j^p = J_p(jr)\cos(p\theta)\cosh(jz) \tag{4.3.22}$$

where the (z, r, θ) are the cylindrical coordinates, J_m is the m^{th} Bessel function, issued from the resolution of the Laplace equation in the cylindrical coordinates and where:

$$\overrightarrow{G_n^m} = \begin{pmatrix} J_m(nr)\cos(m\theta)\sinh(nz) \\ J_m(nr)\cos(m\theta)\cosh(nz) \\ 0 \end{pmatrix} \tag{4.3.23}$$

Giving:

$$\overrightarrow{\mathrm{rot}}\overrightarrow{G_n^m} = \begin{pmatrix} \nu_{n,z}^m \\ \nu_{n,r}^m \\ \nu_{n,\theta}^m \end{pmatrix} = \begin{pmatrix} m\frac{J_m(nr)}{r}\sin(m\theta)\cosh(nz) \\ -m\frac{J_m(nr)}{r}\sin(m\theta)\sinh(nz) \\ n\left(J_m(nr) - \frac{dJ_m}{dr}(nr)\right)\cos(m\theta)\sinh(nz) \end{pmatrix} \tag{4.3.24}$$

Injecting the form of the velocity and of the pressure in the momentum conservation equation gives:

$$\sum_{m,n} \frac{\partial a_n^m}{\partial t} \overrightarrow{\text{rot}}\overrightarrow{G_n^m} + a_n^m \overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi} = \sum_{m,n} \tau_n^m \overrightarrow{\text{grad}}\Lambda_n^m \tag{4.3.25}$$

Projecting the whole equation on $\overrightarrow{\text{grad}}\Lambda_j^p$ gives:

$$\sum_{m,n} \frac{\partial a_n^m}{\partial t} \left\langle \overrightarrow{\text{rot}}\overrightarrow{G_n^m} | \overrightarrow{\text{grad}}\Lambda_j^p \right\rangle + a_n^m \left\langle \overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi}\, | \overrightarrow{\text{grad}}\Lambda_j^p \right\rangle = \sum_{m,n} \tau_n^m \left\langle \overrightarrow{\text{grad}}\Lambda_n^m | \overrightarrow{\text{grad}}\Lambda_j^p \right\rangle \tag{4.3.26}$$

Applying the relations obtained in 4.3.2.2, and thus reducing the integrals to surface integrals, gives:

$$\sum_{m,n} \frac{\partial a_n^m}{\partial t} \underbrace{\iint_S \overrightarrow{\text{rot}}\overrightarrow{G_n^m}.\overrightarrow{n}\,\Lambda_j^p dS}_{=A_{n,j}^{m,p}} + a_n^m \underbrace{\iint\limits_S \overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi}.\overrightarrow{n}\,\Lambda_j^p dS}_{=D_{n,j}^{m,p}} = \sum_{m,n} \tau_n^m \underbrace{\iint\limits_S \overrightarrow{\text{grad}}\Lambda_n^m.\overrightarrow{n}\,\Lambda_j^p dS}_{=M_{n,j}^{m,p}} \tag{4.3.27}$$

The surface S represents the boundary of the cylindrical domain. It is constituted of D_0 and D_H which are the top and the bottom of the cylinder, and of its side, C.

The terms of the previous equation can be expressed by developing the integrals on the surfaces constituting S as:

$$\begin{aligned}
A_{n,j}^{m,p} &= m\cosh(0\cdot n)\cosh(0\cdot j) \iint\limits_{D_0} J_m(nr)J_p(jr)dr\ \sin(m\theta)\cos(p\theta)d\theta \\
&- m\cosh(-nH)\cosh(-jH) \iint\limits_{D_H} J_m(nr)J_p(jr)dr\ \sin(m\theta)\cos(p\theta)d\theta \\
&- mJ_m(nR)J_p(jR) \iint\limits_{C} \sinh(nz)\cosh(jz)dz\ \sin(m\theta)\cos(p\theta)d\theta \\
&= m(\cosh(0\cdot n)\cosh(0\cdot j) - \cosh(nH)\cosh(jH)) \int_0^R J_m(nr)J_p(jr)dr \int_0^{2\pi} \sin(m\theta)\cos(p\theta)d\theta \\
&- mJ_m(nR)J_p(jR) \int_0^H \sinh(nz)\cosh(jz)dz \int_0^{2\pi} \sin(m\theta)\cos(p\theta)d\theta
\end{aligned}$$

$$\begin{aligned}
M_{n,j}^{m,p} &= n\sinh(0\cdot n)\cosh(0\cdot j) \iint\limits_{D_0} J_m(nr)J_p(jr)rdr\ \cos(m\theta)\cos(p\theta)d\theta \\
&- n\sinh(nH)\cosh(jH) \iint\limits_{D_H} J_m(nr)J_p(jr)rdr\ \cos(m\theta)\cos(p\theta)d\theta \\
&+ n\frac{dJ_m}{dr}(nR)J_p(jR) \iint\limits_{C} \cosh(nz)\cosh(jz)dz\ \cos(m\theta)\cos(p\theta)d\theta
\end{aligned}$$

With the test functions Λ_n^m expressed as:

$$\overrightarrow{\text{grad}}\Lambda_n^m = \begin{pmatrix} nJ_m(nr)\cos(m\theta)\sinh(nz) \\ n\frac{dJ_m}{dr}(nr)\cos(m\theta)\cosh(nz) \\ -m\frac{J_m(nr)}{r}\sin(m\theta)\cosh(nz) \end{pmatrix} \tag{4.3.28}$$

For the last matrix, it is preferable to express first intermediate terms. With:

$$\overrightarrow{\text{grad}}(\overrightarrow{\nu}) = \begin{pmatrix} \frac{d\nu_z}{dz} & \frac{d\nu_z}{dr} & \frac{d\nu_z}{rd\theta} \\ \frac{d\nu_r}{dz} & \frac{d\nu_r}{dr} & \frac{d\nu_r}{rd\theta} - \frac{\nu_\theta}{r} \\ \frac{d\nu_\theta}{dz} & \frac{d\nu_\theta}{dr} & \frac{d\nu_\theta}{rd\theta} + \frac{\nu_r}{r} \end{pmatrix} \tag{4.3.29}$$

And the expression of the rotational given in 4.3.24, one can derive:

$$\overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right) = \begin{pmatrix} nm\frac{J_m(nr)}{r}\sin(m\theta)\sinh(nz) \\ -mn\frac{J_m(nr)}{r}\sin(m\theta)\cosh(nz) \\ \frac{dV_\theta}{dz} \\ \\ m\left(\frac{n}{r}\frac{dJ_m}{dr}(nr) - \frac{J_m(nr)}{r^2}\right)\sin(m\theta)\cosh(nz) \\ -m\left(\frac{n}{r}\frac{dJ_m}{dr}(nr) - \frac{J_m(nr)}{r^2}\right)\sin(m\theta)\sinh(nz) \\ \frac{dV_\theta}{dr} \\ \\ m^2\frac{J_m(nr)}{r^2}\cos(m\theta)\cosh(nz) \\ \left(-m^2\frac{J_m(nr)}{r^2} + \frac{n}{r}\left(J_m(nr) + \frac{dJ_m}{dr}(nr)\right)\right)\cos(m\theta)\sinh(nz) \\ \frac{dV_\theta}{rd\theta} + \frac{V_r}{r} \end{pmatrix} \tag{4.3.30}$$

and using:

$$\overrightarrow{\xi} = \begin{pmatrix} -\sin(\chi) \\ 0 \\ \cos(\chi) \end{pmatrix}_{(x,y,z)} = \begin{pmatrix} \cos(\chi) \\ -\sin(\chi)\cos(\theta) \\ \sin(\chi)\sin(\theta) \end{pmatrix}_{(z,r,\theta)} \tag{4.3.31}$$

It gives for the disc surfaces D_0 and D_H:

$$\overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi}.\overrightarrow{n} = \pm\left[\cos(\chi)\frac{d\nu_z}{dz} + \sin(\chi)\left(-\cos(\theta)\frac{d\nu_z}{dr} + \sin(\theta)\frac{d\nu_z}{rd\theta}\right)\right] \tag{4.3.32}$$

and for the cylindrical surface C:

$$\overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi}.\overrightarrow{n} = \pm\left[\cos(\chi)\frac{dV_r}{dz} + \sin(\chi)\left(-\cos(\theta)\frac{dV_r}{dr} + \sin(\theta)\left(\frac{dV_r}{rd\theta} - \frac{V_\theta}{r}\right)\right)\right] \tag{4.3.33}$$

With the notation:

$$D_{n,j}^{m,p} = \cos(\chi)C_{n,j}^{m,p} + \sin(\chi)S_{n,j}^{m,p} \tag{4.3.34}$$

$C_{n,j}^{m,p}$ and $S_{n,j}^{m,p}$ can be expressed as follow:

$$\begin{aligned}
C_{n,j}^{m,p} &= nm\sinh(0\cdot n)\cosh(0\cdot j)\iint_{D_0} J_m(nr)J_p(jr)dr\ \sin(m\theta)\cos(p\theta)d\theta \\
&- nm\sinh(-nH)\cosh(-jH)\iint_{D_H} J_m(nr)J_p(jr)dr\ \sin(m\theta)\cos(p\theta)d\theta \\
&- mnJ_m(nR)J_p(jR)\iint_{C}\cosh(nz)\cosh(jz)dz\ \sin(m\theta)\cos(p\theta)d\theta
\end{aligned}$$

$$\begin{aligned}
S_{n,j}^{m,p} &= m\left(\cosh(0\cdot n)\cosh(0\cdot j)-\cosh(-nH)\cosh(-jH)\right)\times \\
&\left[\int_0^R \frac{J_m(nr)}{r}J_p(jr)dr\int_0^{2\pi}\left(\frac{m+1}{2}\sin((m+1)\theta)-\frac{m-1}{2}\sin((m-1)\theta)\right)\cos(p\theta)d\theta\right. \\
&\left.+\int_0^R \frac{-n}{2}\frac{dJ_m}{dr}(nr)J_p(jr)dr\int_0^{2\pi}\left(\sin((m+1)\theta)+\sin((m-1)\theta)\right)\cos(p\theta)d\theta\right] \\
&+\int_0^{-H}\sinh(nz)\cosh(jz)dz\times \\
&\left[\frac{-mJ_m(nR)J_p(rR)}{R}\int_0^{2\pi}\left(\frac{m+1}{2}\sin((m+1)\theta)-\frac{m-1}{2}\sin((m-1)\theta)\right)\cos(p\theta)d\theta\right. \\
&+n\frac{dJ_m}{dr}(nR)J_p(jR)\int_0^{2\pi}\left(\frac{m+1}{2}\sin((m+1)\theta)+\frac{m-1}{2}\sin((m-1)\theta)\right)\cos(p\theta)\theta \\
&\left.+\frac{n}{2}J_m(nR)J_p(jR)\int_0^{2\pi}\left(\sin((m+1)\theta)-\sin((m-1)\theta)\right)\cos(p\theta)d\theta\right]
\end{aligned}$$

This method was however not implemented due to the difficulty of computing analytical values for the matrices used. Although the scalar product of two Bessel function can be computed analytically, the values of the derivative integrals, and of the integrals divided by r are not so obvious. It would have been possible to use a numerical method to compute the integrals and then store all the values for later use in the matrices. However, it was thought to be more convenient to rely on analytical computation of the matrices values.

Using the Chebyshev Polynomials

The use of the Chebyshev polynomials is here proposed, following ideas of Shen in [42], in order to solve the previous problem of the analytical values of the integrals. Indeed they are Jacobi polynomial, with numerous relations allowing to compute the value of many kinds of integrals.

In this version we express $G_{n,j}^{m}$ as:

$$\overrightarrow{G_{n,j}^{m}} = \begin{pmatrix} r\phi_n(r)\phi_j(z)\cos(m\theta) \\ r\phi_n(r)\phi_j(z)\cos(m\theta) \\ 0 \end{pmatrix} \tag{4.3.35}$$

Where ϕ_k is expressed as a combination of Chebyshev polynomials of the first kind, the T_n.

It follows:

$$\overrightarrow{\text{rot}}\overrightarrow{G_n^m} = \begin{pmatrix} m\phi_n(r)\phi_j(z)\sin(m\theta) \\ -m\phi_n(r)\phi_j(z)\sin(m\theta) \\ \Big(r\phi_n(r)\phi_j'(z) + (\phi_n(r) + r\phi_n'(r))\,\phi_j(z)\Big)\cos(m\theta) \end{pmatrix} \tag{4.3.36}$$

and since:

$$\overrightarrow{\text{grad}}(\overrightarrow{\nu}) = \begin{pmatrix} \frac{d\nu_z}{dz} & \frac{d\nu_z}{dr} & \frac{d\nu_z}{rd\theta} \\ \frac{d\nu_r}{dz} & \frac{d\nu_r}{dr} & \frac{d\nu_r}{rd\theta} - \frac{\nu_\theta}{r} \\ \frac{d\nu_\theta}{dz} & \frac{d\nu_\theta}{dr} & \frac{d\nu_\theta}{rd\theta} + \frac{\nu_r}{r} \end{pmatrix} \tag{4.3.37}$$

We have:

$$\overrightarrow{\text{grad}}\Big(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\Big) = \begin{pmatrix} m\phi_n(r)\phi_j'(z)\sin(m\theta) \\ -m\phi_n(r)\phi_j'(z)\sin(m\theta) \\ \frac{d\nu_\theta}{dz} \\ \\ m\phi_n'(r)\phi_j(z)\sin(m\theta) \\ -m\phi_n'(r)\phi_j(z)\sin(m\theta) \\ \frac{d\nu_\theta}{dr} \\ \\ \frac{m^2}{r}\phi_n(r)\phi_j(z)\cos(m\theta) \\ \Big[-(\frac{1}{r} + \frac{m^2}{r})\phi_n(r)\phi_j(z) - \phi_n(r)\phi_j'(z) + \phi_n'(r)\phi_j(z)\Big]\cos(m\theta) \\ \frac{d\nu_\theta}{rd\theta} + \frac{\nu_r}{r} \end{pmatrix} \tag{4.3.38}$$

The ϕ_k are chosen to respect the boundary conditions of the problem, as prescribed by Shen in [43], while allowing a simple form of their derivatives, in order to obtain sparse matrices:

$$\phi_k = T_k - \frac{2k}{k+2}T_{k+2} + \frac{k}{k+4}T_{k+4} \tag{4.3.39}$$

with:

$$\phi_k' = 2k(T_{k+3} - T_{k+1}) \tag{4.3.40}$$

Those polynomials respects the null Neumann boundary conditions such that:

$$\frac{\partial\overrightarrow{\nu}}{\partial\overrightarrow{n}}\Big|_S = 0 \tag{4.3.41}$$

However the Chebyshev polynomials are orthogonal with respect to the weight $w(x) = \frac{1}{\sqrt{1-x^2}}$. It is thus required to multiply by this weight to obtain sparse matrices, but doing so would prevent one from using the divergence theorem. Indeed in the cylindrical coordinate system, the Jacobian is $|r|$. Incorporating the weight in the scalar product is thus problematic since it differs from the Jacobian, and would invalidate the form of the divergence theorem used. Therefore, it is chosen to modify the test function rather than the scalar product, in the following manner:

$$\widetilde{\Lambda}_{n,j}^p = \Lambda_{n,j}^p w(r,z) = \frac{\phi_n(r)\phi_j(z)\cos(m\theta)}{\sqrt{1-r^2}\sqrt{1-z^2}} \tag{4.3.42}$$

which translates the fact that the boundaries are situated far enough of the cause of the movement.

Injecting the form of the velocity and of the pressure in the momentum conservation equation gives:

$$\sum_{m,n} \frac{\partial a_n^m}{\partial t} \overrightarrow{\mathrm{rot}} \overrightarrow{G_n^m} + a_n^m \overrightarrow{\mathrm{grad}} \left(\overrightarrow{\mathrm{rot}} \overrightarrow{G_n^m} \right) . \overrightarrow{\xi} = \sum_{m,n} \tau_n^m \overrightarrow{\mathrm{grad}} \Lambda_n^m \tag{4.3.43}$$

Projecting the whole equation on $\overrightarrow{\mathrm{grad}\widetilde{\Lambda}^p_{n_2,j_2}}$ gives:

$$\sum_{m,n} \frac{\partial a_n^m}{\partial t} \left\langle \overrightarrow{\mathrm{rot}} \overrightarrow{G_n^m} | \overrightarrow{\mathrm{grad}} \widetilde{\Lambda}^p_{n_2,j_2} \right\rangle + a_n^m \left\langle \overrightarrow{\mathrm{grad}} \left(\overrightarrow{\mathrm{rot}} \overrightarrow{G_n^m} \right) . \overrightarrow{\xi} | \overrightarrow{\mathrm{grad}} \widetilde{\Lambda}^p_{n_2,j_2} \right\rangle = \sum_{m,n} \tau_n^m \left\langle \overrightarrow{\mathrm{grad}} \Lambda_n^m | \overrightarrow{\mathrm{grad}} \widetilde{\Lambda}^p_{n_2,j_2} \right\rangle \tag{4.3.44}$$

It follows that:

$$\sum_{m,n} \frac{\partial a^m_{n_1,j_1}}{\partial t} \overbrace{\iint_S \overrightarrow{\mathrm{rot}} \overrightarrow{G^m_{n_1,j_1}} . \overrightarrow{n} \widetilde{\Lambda}^p_{n_2,j_2} dS}^{=A^{m,p}_{n,j}} + a^m_{n_1,j_1} \overbrace{\iint\limits_S \overrightarrow{\mathrm{grad}} \left(\overrightarrow{\mathrm{rot}} \overrightarrow{G^m_{n_1,j_1}} \right) . \overrightarrow{\xi} . \overrightarrow{n} \widetilde{\Lambda}^p_{n_2,j_2} dS}^{=D^{m,p}_{n,j}}$$
$$= \sum_{m,n} \tau^m_{n_1,j_1} \underbrace{\iint\limits_S \overrightarrow{\mathrm{grad}} \Lambda^m_{n_1,j_1} . \overrightarrow{n} \widetilde{\Lambda}^p_{n_2,j_2} dS}_{=M^{m,p}_{n,j}}$$

The various terms of the equation are expressed with the use of the following notation:

$$c_{k,j} = \int_{-1}^{1} \phi_k(x) \phi_j(x) w(x) x dx \tag{4.3.45}$$

$$d_{k,j} = \int_{-1}^{1} \phi_k(x) \phi_j(x) w(x) dx \tag{4.3.46}$$

$$e_{k,j} = \int_{-1}^{1} \phi'_k(x) \phi_j(x) w(x) x dx \tag{4.3.47}$$

$$f_{k,j} = \int_{-1}^{1} \phi'_k(x) \phi_j(x) w(x) dx \tag{4.3.48}$$

$$A^{m,p}_{n,j} = m \left(\phi_{j_1}(1) \phi_{j_2}(1) - \phi_{j_1}(-1) \phi_{j_2}(-1) \right) c_{n_1,n_2} \int_{-1}^{1} \sin(m\theta) \cos(p\theta) d\theta$$
$$- m \left(\phi_{n_1}(1) \phi_{n_2}(1) - \phi_{n_1}(-1) \phi_{n_2}(-1) \right) d_{j_1,j_2} \int_{-1}^{1} \sin(m\theta) \cos(p\theta) d\theta$$

$$C_{n,j}^{m,p} = -m\left(\phi_{n_1}(1)\phi_{n_2}(1) - \phi_{n_1}(-1)\phi_{n_2}(-1)\right) f_{j_1,j_2} \int_{-1}^{1} \sin(m\theta)\cos(p\theta)d\theta$$

$$\begin{aligned} S_{n,j}^{m,p} &= m\left(\phi_{j_1}(1)\phi_{j_2}(1) - \phi_{j_1}(-1)\phi_{j_2}(-1)\right) \\ &\left[\frac{-1}{2} e_{n_1,n_2} \int_0^{2\pi} \left(\sin((m+1)\theta) + \sin((m-1)\theta)\right)\cos(p\theta)d\theta\right. \\ &\left. + \frac{m}{2} d_{n_1,n_2} \int_0^{2\pi} \left(\sin((m+1)\theta) - \sin((m-1)\theta)\right)\cos(p\theta)d\theta\right] \\ &+ \frac{1}{2}\left(\phi_{n_1}(1)\phi_{n_2}(1) - \phi_{n_1}(-1)\phi_{n_2}(-1)\right) \\ &\left(-(m^2+1)d_{j_1,j_2} - f_{j_1,j_2}\right) \int_0^{2\pi} \left(\sin((m+1)\theta) - \sin((m-1)\theta)\right)\cos(p\theta)d\theta \end{aligned}$$

It was however noticed that with the current form of the rotational, the velocity components in r and z would always be of opposite sign and of equal norm, which is not an adequate constraint to give. Therefore the $\overrightarrow{G}$ field is redefined, in order to cover a more general case, as:

$$\overrightarrow{G_{n,j}^{m}} = \begin{pmatrix} r\phi_n(r)\phi_j(z)\cos(m\theta) \\ r\phi_n(r)\phi_j(z)\cos(m\theta) \\ r\phi_n(r)\phi_j(z)\sin(m\theta) \end{pmatrix} \tag{4.3.49}$$

The r factor being added in order to stabilise the rotational form with the derivation.

Finally to avoid the singular matrices, the test function is no longer taken in the form of a gradient, but as follows:

$$\overrightarrow{\Lambda_{k,h}^{p}} = \begin{pmatrix} \phi_k(r)\phi_h(z)\cos(p\theta) \\ \phi_k(r)\phi_h(z)\cos(p\theta) \\ r\phi_k(r)\phi_h(z)\sin(p\theta) \end{pmatrix} \tag{4.3.50}$$

Note the difference in the form of the θ component which is here to regularise the θ derivative.

This allowed to obtain stable matrices. The form of the velocity and of its gradient are described below.

$$\overrightarrow{\mathrm{rot}}\overrightarrow{G_n^m} = \begin{pmatrix} \left(r\phi_n'(r) + (m+2)\phi_n(r)\right)\phi_j(z)\sin(m\theta) \\ \left(-m\phi_j(z) - r\phi_j'(z)\right)\phi_n(r)\sin(m\theta) \\ \left(r\phi_n(r)\phi_j'(z) - \left(\phi_n(r) + r\phi_n'(r)\right)\phi_j(z)\right)\cos(m\theta) \end{pmatrix} \tag{4.3.51}$$

and:

$$\overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right) = \begin{pmatrix} \left(r\phi_n'(r) + (m+2)\phi_n(r)\right)\phi_j'(z)\sin(m\theta) \\ \left(-m\phi_j'(z) - r\phi_j''(z)\right)\phi_n(r)\sin(m\theta) \\ \left(r\phi_n(r)\phi_j''(z) - \left(\phi_n(r) + r\phi_n'(r)\right)\phi_j'(z)\right)\cos(m\theta) \\ \\ \left(r\phi_n''(r) + (m+3)\phi_n'(r)\right)\phi_j(z)\sin(m\theta) \\ \left(-m\phi_j(z)\phi_n'(r) - r\phi_j'(z)\phi_n'(r) - \phi_j'(z)\phi_n(r)\right)\sin(m\theta) \\ \left((r\phi_n'(r) + \phi_n(r))\phi_j'(z) - \left(2\phi_n'(r) + r\phi_n''(r)\right)\phi_j(z)\right)\cos(m\theta) \\ \\ m\left(\phi_n'(r) + \frac{(m+2)}{r}\phi_n(r)\right)\phi_j(z)\cos(m\theta) \\ \left(\frac{1-m^2}{r}\phi_j(z)\phi_n(r) - (m+1)\phi_n(r)\phi_j'(z) + \phi_n'(r)\phi_j(z)\right)\cos(m\theta) \\ \left(-(m+1)\phi_n(r)\phi_j'(z) + m\phi_n'(r)\phi_j(z)\right)\sin(m\theta) \end{pmatrix} \tag{4.3.52}$$

Furthermore, with:

$$\overrightarrow{\xi} = \begin{pmatrix} -\sin(\chi) \\ 0 \\ \cos(\chi) \end{pmatrix}_{(x,y,z)} = \begin{pmatrix} \cos(\chi) \\ -\sin(\chi)\cos(\theta) \\ \sin(\chi)\sin(\theta) \end{pmatrix}_{(z,r,\theta)} \tag{4.3.53}$$

$$\overrightarrow{\text{grad}}\left(\overrightarrow{\text{rot}}\overrightarrow{G_n^m}\right).\overrightarrow{\xi}.\overrightarrow{n} = \pm\left[\cos(\chi)\frac{d\nu_z}{dz} + \sin(\chi)\left(-\cos(\theta)\frac{d\nu_z}{dr} + \sin(\theta)\frac{d\nu_z}{rd\theta}\right)\right] \tag{4.3.54}$$

However, once again, the model proved to be inadequate. Indeed the matrices turned out to be singular as soon as a too high number of terms were included.

Last resort: going full 3D

As a last resort to improve the conditioning of the matrices, we take the following form of the potential vector:

$$\overrightarrow{G^{m}_{n,j}} = \begin{pmatrix} \alpha^m_{n,j}\phi_n(r)\phi_j(z)\cos(m\theta) \\ \beta^m_{n,j}\phi_n(r)\phi_j(z)\cos(m\theta) \\ \gamma^m_{n,j}\phi_n(r)\phi_j(z)\sin(m\theta) \end{pmatrix} \tag{4.3.55}$$

This was made in the hope that the bad conditioning of the matrices was in fact the result of the constraints laid on the model by the chosen form of the potential vector, that would be too constraint to be able to depict the equations.

We do not present the full development of this method here, since it is highly similar to the previous ones, and gives the same results: matrices with a bad condition number, as soon as the number of elements included in the approximation becomes useful.

Conclusion on the potential vector This formulation had a lot of advantages, but it appeared that it was difficult to obtain non singular matrices with it.

It is thought that this difficulty originates from the fact that the kernel of the rotational operator is tremendously large. Indeed, the kernel contains all fields that can be expressed as a gradient. And because of the properties of the chosen polynomials, which derivatives can be expressed on the same basis, given a large enough number of element, a combination will appear that can be expressed with a gradient, and that will be in the kernel of the rotational. It is therefore difficult to use this assumption in a meaningful way.

In order for this assumption to work efficiently, one would have to find a form of the vector potential with which linear combinations can not be expressed by a gradient. This seems difficult to combine with the property of having easily expressible derivatives for the functions describing the potential vector.

5 A new model of rotor induced velocities

Résumé en français: Un nouveau modèle de vitesses induites par un rotor
Dans ce chapitre, le modèle de vitesses induites développé à la suite des études précédentes est présenté. Il consiste en l'application d'une méthode de Galerkin aux équations d'Euler en incompressible, inspiré de l'approche de Lopez, Marques et Shen [4]. Cela permet de dépasser les défauts de l'approche de Peters en se débarrassant des hypothèses injustifiées, tout en conservant la plupart des avantages de cette approche.

Cependant, cette méthode présente certains défauts. En effet, les équations hyperboliques qu'elles traitent n'ont pas d'amortissement et les termes non linéaires, qui ne sont désormais plus linéarisés, génèrent des instabilités. Mais ces instabilités peuvent être réglées par l'emploi de méthodes de filtrage adaptée aux méthodes spectrales.

Après avoir démontré la pertinence de ce modèle par rapport au modèle de Peters et Huang, il est comparé à d'autres modèles de vitesses induites, de façon à en valider les résultats.

This chapter is dedicated to the development of a new rotor induced velocity model that strives to offer a more accurate description than the one given by the existing dynamic inflow models, while remaining as efficient as possible, with all the advantages of the previous models.

5.1 Motivation

From the conclusion of the study of the current state of the art (see 3.5) it appears that most of the limitations of the model stem from its assumptions. The previous chapter emphasised the difficulty to overcome the drawbacks of the method without modifying these assumptions. Table 5.1 lists the assumptions taken, their domain of validity, and their consequences, both positive and negative.

Assumption	*Validity*	*Advantages*	*Drawbacks*
Incompressibility	Low Mach number	Only two equations instead of the full Navier-stokes equations	Reduced domain of validity and pressure definition
Negligible viscosity	High Reynolds	Uncouple the equations	Leads to hyperbolic equations
Velocity potential	See note 3.1.2	Simplify the equations and the expression of the velocity	Hard to justify the validity of the assumption
Linearisation	High velocity	Uncouple and simplify equations	Introduce static parameters, invalid for hover and near hover cases
Straight Streamlines	High velocity	Simplify integration along a streamline	Same as the linearisation assumption
Frequency Domain	Always	Solve the whole simulation in one step	Number of states[1] and coupling with blade element models

The new model we wish to develop here aims to get rid of the unwanted assumptions. From this new framework, a spectral method will be built, allowing to compute the three components of the induced velocities, at all points around the rotor and for all flight conditions.

We will remove the potential assumption because it is judged to be too limiting to describe the flow below the rotor. We will not linearise the non linear terms, in order to better capture the behaviour of the velocities, to remove the dependency of the treatment of the equations on the flight conditions and to keep the method fully dynamic. Finally, we will develop the method in the time domain in order to have only one set of parameters that varies in time, and to easily couple it with a time marching blade element model.

Removing these assumptions rid us of a powerful tool for solving the equations. Fortunately for us, the resolution of the incompressible Euler and Navier-Stokes equations is an active field of research.

In our case we will follow the methods of treating the incompressible Euler equations prescribed by [3, 4], heavily supported by the theory developed in [42, 43]. They are mostly

[1] The time domain expression only requires one set of coefficients, while the frequency domain formulation requires as many sets of coefficients as frequencies considered.

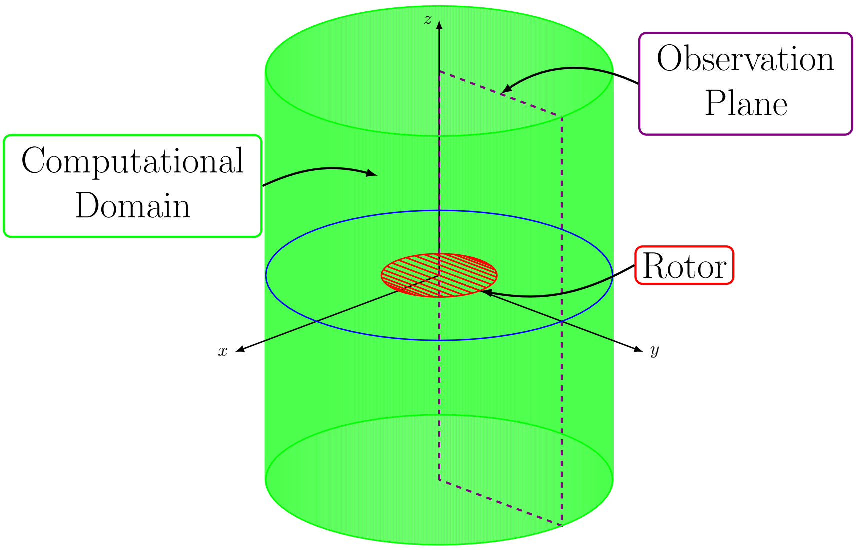


Figure 5.2.1: Computational domain

concerned with the Navier-Stokes incompressible equations, but the method can be applied in the present case nonetheless.

5.2 Developing the model

Treating the equations at hand to describe the induced velocities can be done in several steps. We will discretise time by using a second order projection scheme, as proposed in [4]. Then we will apply a spatial discretisation, by applying a Galerkin method at each time step. The spatial method used is similar to the one used by Peters et al. in their work so as to retain adaptability. But the basis of projection chosen is quite different, and is inspired by the work of Shen in [42, 43].

5.2.1 Choice of domain

Before implementing a solution to the Euler incompressible equations, one needs to choose a computational domain in which the equations will be solved. In our case, the ellipsoidal coordinates would have been ideal to describe the rotor, but the singularities they introduce are a real hindrance for the regularity of the solution.

We therefore chose a cylindrical computational domain D_c, of aspect ratio $\Lambda = \frac{H}{R_c}$, described on Fig. 5.2.1, with the rotor generating the forces on the fluid at its centre:

$$D_c = \left\{ 0 \leqslant \overline{r} \leqslant R_c, 0 \leqslant \theta < 2\pi, \frac{-H}{2} \leqslant \overline{z} \leqslant \frac{H}{2} \right\} \tag{5.2.1}$$

It is to be noted that this bounded domain is chosen because we follow Shen's method which uses Legendre polynomials. However, an unbounded domain could have been modelled with the use of different orthogonal polynomials having more adapted supports, such as the Laguerre and Hermite polynomials.

Since we chose to use Legendre polynomials to describes this domain, we have to scale the axial and radial coordinates into $[-1; 1]$. To do so we use the coordinates:

$$x = \frac{2\overline{r}}{R_c} - 1 \tag{5.2.2}$$

$$z = \frac{2\overline{z}}{H} \tag{5.2.3}$$

For convenience, we will also use the coordinate $r = x + 1$.

5.2.2 Euler equations

The non dimensional Euler incompressible equations are define as:

$$\begin{aligned} \frac{\partial \overrightarrow{\nu}}{\partial \tau} + \overrightarrow{\nabla}(\overrightarrow{\nu}) \cdot \overrightarrow{\nu} &= \overrightarrow{\nabla} p \\ \nabla \cdot \overrightarrow{\nu} &= 0 \end{aligned} \tag{5.2.4}$$

And can be expressed in cylindrical coordinates as follows:

$$\begin{aligned} \frac{\partial u}{\partial \tau} + u\frac{\partial u}{\partial \overline{r}} + v\frac{\partial u}{\overline{r}\partial\theta} - \frac{v^2}{\overline{r}} + w\frac{\partial u}{\partial \overline{z}} &= -\frac{\partial p}{\partial \overline{r}} \\ \frac{\partial v}{\partial \tau} + u\frac{\partial v}{\partial \overline{r}} + v\frac{\partial v}{\overline{r}\partial\theta} + \frac{uv}{\overline{r}} + w\frac{\partial v}{\partial \overline{z}} &= -\frac{\partial p}{\overline{r}\partial\theta} \\ \frac{\partial w}{\partial \tau} + u\frac{\partial w}{\partial \overline{r}} + v\frac{\partial w}{\overline{r}\partial\theta} + w\frac{\partial w}{\partial \overline{z}} &= -\frac{\partial p}{\partial \overline{z}} \\ \frac{\partial(\overline{r}u)}{\overline{r}\partial\overline{r}} + \frac{\partial v}{\overline{r}\partial\theta} + \frac{\partial w}{\partial \overline{z}} &= 0 \end{aligned} \tag{5.2.5}$$

This expression is however not well suited for the non axisymmetric case. As mentioned by Lopez in [4], a good way to add some symmetry in the non axisymmetric case is to use the complex variables $u^+ = u+iv$ and $u^- = u-iv$ in place of the u and v components of the velocity. This change in coordinate also allows an easier definition of the pole conditions on the axis. It gives the following form of the equations:

$$\begin{aligned} \frac{\partial u^+}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial u^+}{\partial \overline{r}} + \frac{u^+ - u^-}{2i\overline{r}}\left(\frac{\partial u_+}{\partial \theta} + iu_+\right) + w\frac{\partial u^+}{\partial \overline{z}} &= -\left(\frac{\partial p}{\partial \overline{r}} + i\frac{\partial p}{\overline{r}\partial\theta}\right) \\ \frac{\partial u^-}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial u^-}{\partial \overline{r}} + \frac{u^+ - u^-}{2i\overline{r}}\left(\frac{\partial u_-}{\partial \theta} - iu_-\right) + w\frac{\partial u^-}{\partial \overline{z}} &= -\left(\frac{\partial p}{\partial \overline{r}} - i\frac{\partial p}{\overline{r}\partial\theta}\right) \\ \frac{\partial w}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial w}{\partial \overline{r}} + \frac{u^+ - u^-}{2i}\frac{\partial w}{\overline{r}\partial\theta} + w\frac{\partial w}{\partial \overline{z}} &= -\frac{\partial p}{\partial \overline{z}} \\ \frac{\partial}{\overline{r}\partial\overline{r}}\left(\overline{r}\frac{u^+ + u^-}{2}\right) + \frac{\partial}{\overline{r}\partial\theta}\left(\frac{u^+ - u^-}{2i}\right) + \frac{\partial w}{\partial \overline{z}} &= 0 \end{aligned} \tag{5.2.6}$$

The last form of the equations is obtained by accounting for the change in scale. Indeed, changing the variables requires to account for the scale in the derivatives: $\frac{\partial f}{\partial \overline{r}} = \frac{2}{R_c}\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial \overline{z}} = \frac{2}{H}\frac{\partial f}{\partial z}$. This gives:

$$\begin{aligned}
\frac{R_c}{2}\frac{\partial u^+}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial u^+}{\partial x}+\frac{u^+ - u^-}{2ir}\left(\frac{\partial u_+}{\partial \theta}+iu_+\right)+\frac{w}{\Lambda}\frac{\partial u^+}{\partial z} &= -\left(\frac{\partial p}{\partial x}+i\frac{\partial p}{r\partial \theta}\right)\\
\frac{R_c}{2}\frac{\partial u^-}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial u^-}{\partial x}+\frac{u^+ - u^-}{2ir}\left(\frac{\partial u_-}{\partial \theta}-iu_-\right)+\frac{w}{\Lambda}\frac{\partial u^-}{\partial z} &= -\left(\frac{\partial p}{\partial x}-i\frac{\partial p}{r\partial \theta}\right)\\
\frac{R_c}{2}\frac{\partial w}{\partial \tau}+\frac{u^+ + u^-}{2}\frac{\partial w}{\partial x}+\frac{u^+ - u^-}{2ir}\frac{\partial w}{\partial \theta}+\frac{w}{\Lambda}\frac{\partial w}{\partial z} &= -\frac{1}{\Lambda}\frac{\partial p}{\partial z}\\
\frac{\partial}{r\partial x}\left(r\frac{u^+ + u^-}{2}\right)+\frac{\partial}{r\partial \theta}\left(\frac{u^+ - u^-}{2i}\right)+\frac{1}{\Lambda}\frac{\partial w}{\partial z} &= 0
\end{aligned} \tag{5.2.7}$$

5.2.3 Time discretisation

In order to discretise the time, we set $\overrightarrow{\nu_k}$ as the approximation of the velocity at time $\tau = k\delta_t$, with components (u_k, v_k, w_k) in the axisymmetric case and $(u_{+,k}, u_{-,k}, w_k)$ for the non axisymmetric case. Applying the first step of the second order semi-implicit projection scheme presented in [3, 4] gives an estimate of the velocity $\overrightarrow{\tilde{\nu}}_{k+1}$:

$$\frac{1}{2\delta_t}\left(3\overrightarrow{\tilde{\nu}}_{k+1} - 4\overrightarrow{\nu_k} + \overrightarrow{\nu}_{k-1}\right) = -\overrightarrow{\text{grad}}(p_k) - 2\overrightarrow{N}_k + \overrightarrow{N}_{k-1} \tag{5.2.8}$$

where we use the notation $\overrightarrow{N}_k = \overrightarrow{\nabla}(\overrightarrow{\nu_k}) \cdot \overrightarrow{\nu_k}$. The non-linear terms are expressed explicitly with a simple time scheme.

We then have a relation linking both p_{k+1} and $\overrightarrow{\nu}_{k+1}$ to its estimation computed in the previous equation:

$$\frac{3}{2\delta_t}(\overrightarrow{\nu}_{k+1} - \overrightarrow{\tilde{\nu}}_{k+1}) = -\overrightarrow{\text{grad}}(p_{k+1} - p_k) \tag{5.2.9}$$

Applying the divergence operator to this equation gives an equation to evaluate p_{k+1}, thanks to the continuity equation:

$$-\frac{3}{2\delta_t}\text{div}(\overrightarrow{\tilde{\nu}}_{k+1}) = -\Delta(p_{k+1} - p_k) \tag{5.2.10}$$

Finally we have $\overrightarrow{\nu}_{k+1}$:

$$\overrightarrow{\nu}_{k+1} = \overrightarrow{\tilde{\nu}}_{k+1} - \frac{2\delta_t}{3}\overrightarrow{\text{grad}}(p_{k+1} - p_k) \tag{5.2.11}$$

One can notice the use of an explicit scheme for the non linear terms, allowing to decouple the equations, and the fact that $\overrightarrow{\nu}_k$ will always respect the continuity equation. Thus, this algorithm taken from [3, 4] solves two of the problems that we faced with the previous model. It guarantees that the continuity equation is respected in all the domain, and it accounts for the non linear terms without linearising them or introducing any parameters.

5.2.4 Space discretisation

Once the time has been discretised, we are left at each time step with three different equations to solve. We will apply on them a Galerkin method. Following the methods advocated by Shen in [42, 43], we will project the radial and orthoradial components on combinations of Legendre polynomials. Since these polynomials will have to respect the boundary conditions, we will describe them after the boundary conditions definition in section 5.3. The azimuthal component will be projected on trigonometric polynomials in order to exploit its periodicity. The choice of the Legendre polynomials rather than the Chebyshev or other orthogonal Jacobi polynomials might seem arbitrary, but they are prefered in the current application for two reasons: their weight is simple, and they have been thoroughly studied, offering a large number of available relations, as shown in [67].

The components of the velocity or of the pressure are therefore expressed as follows:

$$f_k(x,\theta,z) = \sum_{n=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-\infty}^{\infty} \hat{f}_{k,m}^{n,j}\psi_n(x)\phi_j(z)e^{im\theta} \tag{5.2.12}$$

where the polynomials ϕ_j and ψ_n will depend on the boundary conditions, and where f_k represent the quantities of interest at time k. We here use the x variable since its domain is $[-1;1]$ which is the domain of definition of the Legendre polynomials. Note that $x=-1$ corresponds to the axis ($r=0$).

One can then truncate the approximation to the desired order. We will here consider the maximal orders to be N_r, N_z and N_θ respectively for the radial, axial and azimuthal description.

With the specific form taken for the non linear terms (see equation 5.2.7), one can notice that the azimuthal orders are uncoupled. This allows to separate the various orders in azimuth, and to define for all values of m:

$$f_k^m(x,z) = \sum_{n=0}^{N_r}\sum_{j=0}^{N_z} \hat{f}_{k,m}^{n,j}\psi_n(x)\phi_j(z) \tag{5.2.13}$$

and thus:

$$f_k(x,\theta,z) = \sum_{m=-N_\theta}^{N_\theta} f_k^m(x,z)e^{im\theta} \tag{5.2.14}$$

We inject this form for each of the quantity of interest in the equations, and apply the Galerkin method by projecting on the same polynomial space. This provides a matrix representation of the equations, with $N_\theta+1$ uncoupled equations. Therefore, for a considered m order, we have, for the first step of the algorithm:

$$\begin{aligned}
\tilde{u}_{k+1}^m &= \frac{4}{3}u_k^m - \frac{1}{3}u_{k-1}^m - \frac{2\delta_t}{3}\left({}^m_uA\right)^{-1}\left({}^m_uMp_k^m + 2\,{}_uN_k^m - {}_uN_{k-1}^m\right)\\
\tilde{v}_{k+1}^m &= \frac{4}{3}v_k^m - \frac{1}{3}v_{k-1}^m - \frac{2\delta_t}{3}\left({}^m_vA\right)^{-1}\left({}^m_vMp_k^m + 2\,{}_vN_k^m - {}_vN_{k-1}^m\right)\\
\tilde{w}_{k+1}^m &= \frac{4}{3}w_k^m - \frac{1}{3}w_{k-1}^m - \frac{2\delta_t}{3}\left({}^m_wA\right)^{-1}\left({}^m_wMp_k^m + 2\,{}_wN_k^m - {}_wN_{k-1}^m\right)
\end{aligned} \tag{5.2.15}$$

The notation used here for the matrices is as follows: the upper left subscript refers to the azimuthal order, the lower left to the component of the velocity. The non linear terms follow a similar notation, but with the azimuthal order subscript on the right, similar to the notation of the velocity components.

Then for the second step:

$$p_{k+1}^m = p_k^m + \frac{3}{2\delta_t}\left({}^mL\right)^{-1}\left({}^m_uD\tilde{u}_k^m + {}^m_vD\tilde{v}_k^m + {}^m_wD\tilde{w}_k^m\right) \tag{5.2.16}$$

And for the third step:

$$
\begin{aligned}
u_{k+1}^{m} &= \tilde{u}_{k+1}^{m} - \frac{2\delta_t}{3}\left({}_u^m A\right)^{-1}\left({}_u^m M(p_{k+1}^m - p_k^m)\right) \\
v_{k+1}^{m} &= \tilde{v}_{k+1}^{m} - \frac{2\delta_t}{3}\left({}_v^m A\right)^{-1}\left({}_v^m M(p_{k+1}^m - p_k^m)\right) \\
w_{k+1}^{m} &= \tilde{w}_{k+1}^{m} - \frac{2\delta_t}{3}\left({}_w^m A\right)^{-1}\left({}_w^m M(p_{k+1}^m - p_k^m)\right)
\end{aligned} \tag{5.2.17}
$$

The coefficients are put in column vectors, while the various terms of the matrices are defined by:

$$
{}_u A_{n,k}^{j,h} = \int_{-1}^{1} \psi_n(x)\psi_k(x)(x+1)dx \cdot \int_{-1}^{1} \phi_j(z)\phi_h(z)dz \tag{5.2.18}
$$

Thanks to the independence of the r and z coordinates in the approximation, the matrices can be seen to be formed by Kronecker products from smaller matrices, as follows:

$$
{}_u A = {}_u A_r \otimes {}_u A_z \tag{5.2.19}
$$

where $\otimes$ is the Kronecker product, ${}_u A_r$ and ${}_u A_z$ are square matrices defined by:

$$
{}_u A_r^{n,k} = \int_{-1}^{1} \psi_n(x)\psi_k(x)(x+1)dx \tag{5.2.20}
$$

$$
{}_u A_z^{j,l} = \int_{-1}^{1} \phi_j(z)\phi_l(z)dz \tag{5.2.21}
$$

Here the upper right subscripts of the matrices refer to the coefficients of the polynomials involved in the scalar products, while the lower right subscript refers to the type of scalar product considered, r representing the integration along the radial axis, and z along the axial axis.

The other matrices used are described as follows:

$$
{}_u^m A_r^{n,k} = \int_{-1}^{1} {}_u\psi_n(x)\,{}_u\psi_k(x)(x+1)dx \tag{5.2.22}
$$

$$
{}_v^m A_r^{n,k} = \int_{-1}^{1} {}_v\psi_n(x)\,{}_v\psi_k(x)(x+1)dx \tag{5.2.23}
$$

$$
{}_w^m A_r^{n,k} = \int_{-1}^{1} {}_w\psi_n(x)\,{}_w\psi_k(x)(x+1)dx \tag{5.2.24}
$$

$$ {}^{m}_{u}M^{n,k}_{r} = \int_{-1}^{1} \frac{\partial\, {}_{p}\psi_n}{\partial x}(x) {}_{u}\psi_k(x)(x+1)dx \tag{5.2.25} $$

$$ {}^{m}_{v}M^{n,k}_{r} = \int_{-1}^{1} {}_{p}\psi_n(x) {}_{v}\psi_k(x)dx \tag{5.2.26} $$

$$ {}^{m}_{w}M^{n,k}_{r} = \int_{-1}^{1} {}_{p}\psi_n(x) {}_{w}\psi_k(x)(x+1)dx \tag{5.2.27} $$

$$ {}^{m}_{u}D^{n,k}_{r} = \int_{-1}^{1} \left((x+1)\frac{\partial\, {}_{u}\psi_n}{\partial x}(x) + {}_{u}\psi_n \right) {}_{p}\psi_k(x)dx \tag{5.2.28} $$

$$ {}^{m}_{v}D^{n,k}_{r} = \int_{-1}^{1} {}_{v}\psi_n(x) {}_{p}\psi_k(x)dx \tag{5.2.29} $$

$$ {}^{m}_{w}D^{n,k}_{r} = \int_{-1}^{1} {}_{w}\psi_n(x) {}_{p}\psi_k(x)(x+1)dx \tag{5.2.30} $$

$$ {}^{m}_{u}L^{n,k}_{r} = -\int_{-1}^{1} \frac{\partial\, {}_{p}\psi_n}{\partial x}(x) \frac{\partial\, {}_{p}\psi_k}{\partial x}(x)(x+1)dx \tag{5.2.31} $$

$$ {}^{m}_{v}L^{n,k}_{r} = \int_{-1}^{1} \frac{1}{x+1} {}_{p}\psi_n(x) {}_{p}\psi_k(x)dx \tag{5.2.32} $$

$$ {}^{m}_{w}L^{n,k}_{r} = \int_{-1}^{1} {}_{p}\psi_n(x) {}_{p}\psi_k(x)(x+1)dx \tag{5.2.33} $$

$$ {}^{m}_{u}A^{j,l}_{z} = \int_{-1}^{1} {}_{u}\phi_j(z) {}_{u}\phi_l(z)dz \tag{5.2.34} $$

$$ {}^{m}_{v}A^{j,l}_{z} = \int_{-1}^{1} {}_{v}\phi_j(z) {}_{v}\phi_l(z)dz \tag{5.2.35} $$

$$ {}^{m}_{w}A^{j,l}_{z} = \int_{-1}^{1} {}_{w}\phi_j(z) {}_{w}\phi_l(z)dz \tag{5.2.36} $$

$$ {}_{u}^{m}M_z^{j,l} = \int_{-1}^{1} {}_{p}\phi_j(z)\,{}_{u}\phi_l(z)dz \tag{5.2.37} $$

$$ {}_{v}^{m}M_z^{j,l} = \int_{-1}^{1} {}_{p}\phi_j(z)\,{}_{v}\phi_l(z)dz \tag{5.2.38} $$

$$ {}_{w}^{m}M_z^{j,l} = \int_{-1}^{1} \frac{\partial\,{}_{p}\phi_j}{\partial z}(z)\,{}_{w}\phi_l(z)dz \tag{5.2.39} $$

$$ {}_{u}^{m}D_z^{j,l} = \int_{-1}^{1} {}_{u}\phi_j(z)\,{}_{p}\phi_l(z)dz \tag{5.2.40} $$

$$ {}_{v}^{m}D_z^{j,l} = \int_{-1}^{1} {}_{v}\phi_j(z)\,{}_{p}\phi_l(z)dz \tag{5.2.41} $$

$$ {}_{w}^{m}D_z^{j,l} = \int_{-1}^{1} \frac{\partial\,{}_{w}\phi_j}{\partial z}(z)\,{}_{p}\phi_l(z)dz \tag{5.2.42} $$

$$ {}_{u}^{m}L_z^{j,l} = \int_{-1}^{1} {}_{p}\phi_j(z)\,{}_{p}\phi_l(z)dz \tag{5.2.43} $$

$$ {}_{v}^{m}L_z^{j,l} = \int_{-1}^{1} {}_{p}\phi_j(z)\,{}_{p}\phi_l(z)dz \tag{5.2.44} $$

$$ {}_{w}^{m}L_z^{j,l} = -\int_{-1}^{1} \frac{\partial\,{}_{p}\phi_j}{\partial z}(z)\frac{\partial\,{}_{p}\phi_l}{\partial z}(z)dz \tag{5.2.45} $$

where the ${}_{u}\psi, {}_{v}\psi, {}_{w}\psi, {}_{p}\psi$ designate the r polynomials for the induced velocity components u, v, w, and for the pressure p respectively, while the ${}_{u}\phi, {}_{v}\phi, {}_{w}\phi, {}_{p}\phi$ are for the z part. It is to be noted that the m dependency of the polynomials is dropped here for clarity.

This form of the matrices can be further exploited thanks to the properties of the Kronecker product, in order to reduce the size of the matrices involved, and thus improve the efficiency of the method. For example, the product ${}_{u}A \cdot u_k^m$ can be written as ${}_{u}A_z \cdot U_k^m \cdot {}_{u}A_r^T$, where U_k^m is a matrix containing the same coefficients as u_k^m. We will refer to this form of the equations as the 'mini-matrices' form, for lack of a better name.

This form of the equations might however be a problem when solving the equations. The inversion is indeed less straightforward for some cases. In the case of the second step of the algorithm, the matrix decomposition method has to be employed. This method is described in section 5.2.5.

5.2.5 Matrix Decomposition Method

There are two ways of solving the obtained equations depending on how they are expressed. The first is the more direct one, but is much less efficient. Indeed, if one considers the states in vectors, and use Kronecker product to express the matrices, one simply has to inverse the matrices to solve for the coefficients. However the size of the matrices are larger than they need to be ($N_r N_z \cdot N_r N_z$) and manipulating such huge objects is not efficient. It makes the computation of inverse and of products needlessly long. Using the properties of the Kronecker product allows to reduce the size of the matrices considered. This significantly improves performance for the multiplication, however, the process needed to solve the equation becomes more involved than a simple inversion. It is referred in the literature under several name such as the Matrix Decomposition Method [42, 43], the Matrix Diagonalization Method or the Tensor Product Method.

We will present this method here in the case that is of interest, i.e. solving for the pressure in the second equation of the algorithm. Using the mini-matrices form, and with D representing the divergence terms, the equation can be presented under the following form (the m indices have been dropped off the matrices for conciseness):

$$({}_{u,v}L_z)^T \, p_k^m \left({}_uL_r - m^2{}_vL_r\right) + ({}_wL_z)^T \, p_k^m \, {}_wL_r = D \tag{5.2.46}$$

One can see here that a simple inversion is not possible in order to solve for p_k^m because of the impossibility to factor the matrices. We however use to our advantage the fact that ${}_uL_z$ and ${}_vL_z$ are identical to slightly simplify the problem (${}_{u,v}L_z$ thus represents both ${}_uL_z$ and ${}_vL_z$).

The idea behind the Matrix Diagonalization Method is to look for the eigenvalues of $\left(L_r^r - m^2 L_\theta^r\right)(L_z^r)^{-1}$. Indeed:

$$\Big(\left({}_uL_r - m^2{}_vL_r\right)({}_wL_r)^{-1} - \lambda I\Big)\, g = 0 \tag{5.2.47}$$

with g an eigenvector and λ an eigenvalue, is equivalent to:

$$\left({}_uL_r - m^2{}_vL_r\right) g = \lambda g_w L_r \tag{5.2.48}$$

Solving this eigenvalue problem allows to put the previous equation under the following form, with $\left({}_uL_r - m^2{}_vL_r\right)({}_wL_r)^{-1} = E\Lambda E^{-1}$:

$$E^{-1}\left({}_uL_r - m^2{}_vL_r\right) = \Lambda E^{-1}{}_wL_r \tag{5.2.49}$$

where E is the eigenvector matrix and Λ the diagonal eigenvalue matrix.

This allows, if one write $p_k^m = YE^{-1}$, to write the pressure equation as:

$$({}_{u,v}L_z)^T \, Y\Lambda E^{-1}{}_wL_z + ({}_wL_z)^T \, YE^{-1}{}_wL_r = D \tag{5.2.50}$$

which gives:

$$({}_{u,v}L_z)^T \, Y\Lambda + ({}_wL_z)^T \, Y = D\,({}_wL_r)^{-1} E \tag{5.2.51}$$

Thanks to the fact that Λ is diagonal, one can solve for the columns of Y separately, i.e. $\forall \lambda_j$:

$$(({}_{u,v}L_z)^T \lambda_j + ({}_wL_z)^T) Y_j = (D\,({}_wL_r)^{-1} E)_j \tag{5.2.52}$$

Then multiplying Y by E^{-1} on the right, one can find the value of p_k^m.

This method, coupled with the reduced size matrices, allows to improve the performance of the algorithm significantly, and makes it less than quadratic in N_r and N_z.

5.2.6 The non linear terms

The non linear terms are all the terms coming from the $\overrightarrow{\overrightarrow{\text{grad}}}(\overrightarrow{v}) \cdot \overrightarrow{v}$, which are for the three components of the velocity:

$$\begin{aligned}
{}_u\text{NL} &= u\frac{\partial u}{\partial r} && +\frac{v}{r}\left(\frac{\partial u}{\partial \theta} - v\right) && +w\frac{\partial u}{\partial z} \\
{}_v\text{NL} &= u\frac{\partial v}{\partial r} && +\frac{v}{r}\left(\frac{\partial v}{\partial \theta} + u\right) && +w\frac{\partial v}{\partial z} \\
{}_w\text{NL} &= u\frac{\partial w}{\partial r} && +\frac{v}{r}\frac{\partial w}{\partial \theta} && +w\frac{\partial w}{\partial z}
\end{aligned} \tag{5.2.53}$$

They therefore become, with the complex transforms $u_+ = u + iv$ and $u_- = u - iv$:

$$\begin{aligned}
\text{NL}_+ &= \frac{u_+ + u_-}{2}\frac{\partial u_+}{\partial r} && +\frac{u_+ - u_-}{2ir}\left(\frac{\partial u_+}{\partial \theta} + iu_+\right) && +w\frac{\partial u_+}{\partial z} \\
\text{NL}_- &= \frac{u_+ + u_-}{2}\frac{\partial u_-}{\partial r} && +\frac{u_+ - u_-}{2ir}\left(\frac{\partial u_-}{\partial \theta} - iu_-\right) && +w\frac{\partial u_-}{\partial z} \\
\text{NL}_z &= \frac{u_+ + u_-}{2}\frac{\partial w}{\partial r} && +\frac{u_+ - u_-}{2ir}\frac{\partial w}{\partial \theta} && +w\frac{\partial w}{\partial z}
\end{aligned} \tag{5.2.54}$$

The terms they will generate, once projected on another basis with the Galerkin method, will be products of three polynomials in the best case (i.e. when the derivative can be expressed with a single polynomial). In order to avoid the struggle of finding an analytical value for all the various combinations of polynomials, we will use a Legendre-Gauss-Lobato quadrature to evaluate the relevant integrals. This will be less efficient than using analytical values, but will also avoid storing the values of the numerous combinations of integrals.

A Gauss quadrature is a way to approximate the value of an integral by evaluating the function to integrate at a given number of well chosen of points, with well chosen weights:

$$\int_{-1}^{1} f(x)dx \approx \sum_{n=0}^{N} f(x_n)w_n \tag{5.2.55}$$

The more points are added, the more precise the method is, and it happens to be exact for polynomials up to a given order, depending on the quadrature type.

The Legendre-Gauss-Lobato quadrature is a special case of the Gauss quadrature where the points and weights chosen depend on the derivatives of a Legendre polynomial and include the border points. It is exact for all polynomials of order up to $2N-3$, where N is the number of points used in the quadrature, which is particularly useful in our case, where all terms can be expressed as polynomials. We describe the specifics of this quadrature in appendix B.

Accounting for the non linear terms has the advantage to better represent the phenomenon, while avoiding to introduce parameters to describe the flow which lack any dynamics. However, they also introduce some difficulties. The two main difficulties we have faced are the fact that they introduce high order terms which are not dampen and tend to destabilise the algorithm. The second point is due to the loss of regularity due to some non linear terms (e.g., the azimuthal derivation introduces a division by r). This loss of regularity needs to be handle properly to avoid introducing spurious terms, and in order to maintain the accuracy of the approximation on the axis.

5.2.7 Accounting for the free stream velocity

In order to account for the free stream velocity, we decide to describe it as in the Huang and Peters model, with a modulus V_∞ and an angle χ. The free stream velocity is therefore described as:

$$\overrightarrow{V_\infty} = \begin{pmatrix} V_\infty \sin(\chi) \\ 0 \\ V_\infty \cos(\chi) \end{pmatrix}_{(x,y,z)} = \begin{pmatrix} V_\infty \sin(\chi)\cos(\theta) \\ V_\infty \sin(\chi)\sin(\theta) \\ V_\infty \cos(\chi) \end{pmatrix}_{(r,\theta,z)} \tag{5.2.56}$$

The free stream velocity can then be added to the equations, where it only appears in the non linear terms, and will be treated similarly, as follows:

$$\overrightarrow{\text{grad}}(\overrightarrow{V_\infty} + \overrightarrow{v}) \cdot \left(\overrightarrow{V_\infty} + \overrightarrow{v}\right) = \overrightarrow{\text{grad}}(\overrightarrow{v}) \cdot \left(\overrightarrow{V_\infty}\right) + \overrightarrow{\text{grad}}(\overrightarrow{v}) \cdot (\overrightarrow{v}) \tag{5.2.57}$$

It is to be noted that these parameters are inputs of the model, and are not dynamic parameters, since they define the rotor velocity in the air which is uncorrelated from the induced velocity it generates in the simulations. However, nothing prevents the change of these parameters from one time step to another, while keeping the accurate description of the wake. Thus, although they are described similarly to the Huang and Peters model, they do not hinder the accuracy of the model.

5.2.8 No need for negative m values

In the azimuthal description, one would need to consider both negative and positive values of m values. However the fact that all our quantities have in fact real values imposes additional information, thus removing the need for the computation of the negative values of m. Indeed, e.g., for the component w of the velocity, the fact that its values are real imposes $w_{-m} = \overline{w_m}$. One can refer to [4] for a more thorough explanation.

5.3 Boundary conditions

One of the strengths of the Galerkin method is to avoid adding equations to the system in order to respect the boundary conditions by making every element in the solution's description respect them. In this section, we will look at various possibilities for our boundary conditions, and the choice of elements they impose for the Galerkin method.

5.3.1 Open boundary condition

For our application, we require the boundaries of our cylinder (side, top and bottom) to be open boundary conditions, i.e. to let the airflow pass the limit of the domain as if there was no boundary. Two options were tested in this case. The first one is a simple combination of Neumann boundary conditions both on the pressure and on the velocity:

$$\begin{cases} \dfrac{\partial \overrightarrow{v}}{\partial \overrightarrow{n}} = 0 \\ \dfrac{\partial p}{\partial \overrightarrow{n}} = 0 \end{cases} \tag{5.3.1}$$

Here, $\overrightarrow{n}$ represents the normal vector to the boundary. The boundary condition on the velocity translates the fact that the boundary is far enough of the rotor disc, while the pressure boundary condition implies a null pressure gradient in the normal direction of the boundary, which is coherent with the velocity boundary condition. This combination of boundary conditions gives satisfying results, but might sometimes generate non physical phenomena at the boundary, which might end up making the algorithm divergent. It is however easily implemented with the Galerkin method.

A second, more robust, open boundary condition was designed by Dong in [60]. It also consists in a Neumann boundary condition on the velocity, but coupled this time to an adaptable Dirichlet boundary condition on the pressure:

$$\begin{cases} \dfrac{\partial \overrightarrow{v}}{\partial \overrightarrow{n}} = 0 \\ \quad p = -\dfrac{1}{2}|\overrightarrow{v}|^2 S_0(\overrightarrow{v} \cdot \overrightarrow{n}) \end{cases} \tag{5.3.2}$$

Here, S_0 is a smoothed step function designed to be null if some velocity is leaving the domain and is 1 otherwise. (See [60] for more information).

The aim of the pressure boundary condition is here to balance the energy equation when airflow is leaving or entering the domain. It is especially efficient in the case of vortices leaving the computational domain. This boundary condition is however not that easy to implement in the case of a spectral method. Indeed it implies a value that is not only variable at the boundary, but also that depends on the current value of the velocity. Its implementation can however be done, by using the lifting method described by Shen in [43], while using quadrature (ideally the same as the one used for the non linear elements) to evaluate the velocity at the boundary. Indeed, once the velocity is known at the boundary, one can compute the required pressure Dirichlet boundary condition and impose it with the lifting method.

5.3.2 Axis boundary conditions

On top of the previously mentioned boundary conditions, one needs to specify conditions on the axis, which is singular in cylindrical coordinates. These conditions impose some regularity to the velocity and pressure and are referred to as pole conditions. As described in [4], two main types of pole conditions are considered: the essential pole conditions, that are the minimal requirements for the velocity to be regular, and the natural pole conditions that ensure analyticity[2] of the velocity in the Cartesian coordinate system. In the current version of the model, only the essential pole conditions are accounted for. We will see in section 5.11 that one can benefit from accounting for the natural pole conditions.

These essential pole conditions consist in adding Dirichlet or Neumann boundary condition at $x = -1$ for all the quantities considered, depending on their azimuthal order m:

For $m = 0$:

$$\left.\frac{\partial u^+_{k,0}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad u^+_{k,0}(x=-1) = 0 \tag{5.3.3}$$

$$\left.\frac{\partial w_{k,0}}{\partial x}\right|_{x=-1} = 0 \tag{5.3.4}$$

$$\left.\frac{\partial p_{k,0}}{\partial x}\right|_{x=-1} = 0 \tag{5.3.5}$$

[2] Analyticity defines the property of a function to be described locally by a convergent power serie.

For $m = 1$:

$$\left.\frac{\partial u^+_{k,1}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad u^+_{k,1}(x=-1) = 0 \tag{5.3.6}$$

$$\left.\frac{\partial u^-_{k,1}}{\partial x}\right|_{x=-1} = 0 \tag{5.3.7}$$

$$\left.\frac{\partial w_{k,1}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad w_{k,1}(x=-1) = 0 \tag{5.3.8}$$

$$\left.\frac{\partial p_{k,1}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad p_{k,1}(x=-1) = 0 \tag{5.3.9}$$

$$\tag{5.3.10}$$

For $m > 1$:

$$\left.\frac{\partial u^+_{k,m}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad u^+_{k,m}(x=-1) = 0 \tag{5.3.11}$$

$$\left.\frac{\partial u^-_{k,m}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad u^-_{k,m}(x=-1) = 0 \tag{5.3.12}$$

$$\left.\frac{\partial w_{k,m}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad w_{k,m}(x=-1) = 0 \tag{5.3.13}$$

$$\left.\frac{\partial p_{k,m}}{\partial x}\right|_{x=-1} = 0 \quad \text{and} \quad p_{k,m}(x=-1) = 0 \tag{5.3.14}$$

$$\tag{5.3.15}$$

The summary of the boundary conditions is presented in Fig 5.3.1. It presents the observation plane as described by Fig 5.2.1, and precises the boundary conditions for this plane.

5.3.3 Choice of polynomials

We now have a complete set of boundary conditions. This will allow to choose sets of polynomials respecting these boundary conditions. We will use combinations of Legendre polynomials L_n, that should ensure a good conditioning of the matrices.

We basically need four different sets of polynomials to cover all our boundary conditions. The first one must have null Neumann boundary conditions at both boundaries:

$$\gamma_n(x) = L_n(x) - \frac{n(n+1)}{(n+2)(n+3)} L_{n+2}(x) \tag{5.3.16}$$

The second one respect the same boundary conditions plus a null Dirichlet boundary condition at $x = -1$:

$$\begin{aligned}\kappa_n(x) = &-\frac{n^2+5n+6}{n+1} L_n(x) - \frac{2n^3+17n^2+45n+36}{(2n+5)(n+1)} L_{n+1}(x) \\ &+ nL_{n+2}(x) + \frac{2n^2+7n+6}{2n+5} L_{n+3}\end{aligned} \tag{5.3.17}$$

The third one is used on the pressure in the case where the Dong boundary condition is used. It has a null Dirichlet boundary condition at $x = 1$, and a null Neumann boundary condition at $x = -1$:

$$\chi_n(x) = L_n(x) - \frac{2n+3}{(n+2)^2} L_{n+1}(x) + \left(\frac{2n+3}{(n+2)^2} - 1\right) L_{n+2}(x) \tag{5.3.18}$$

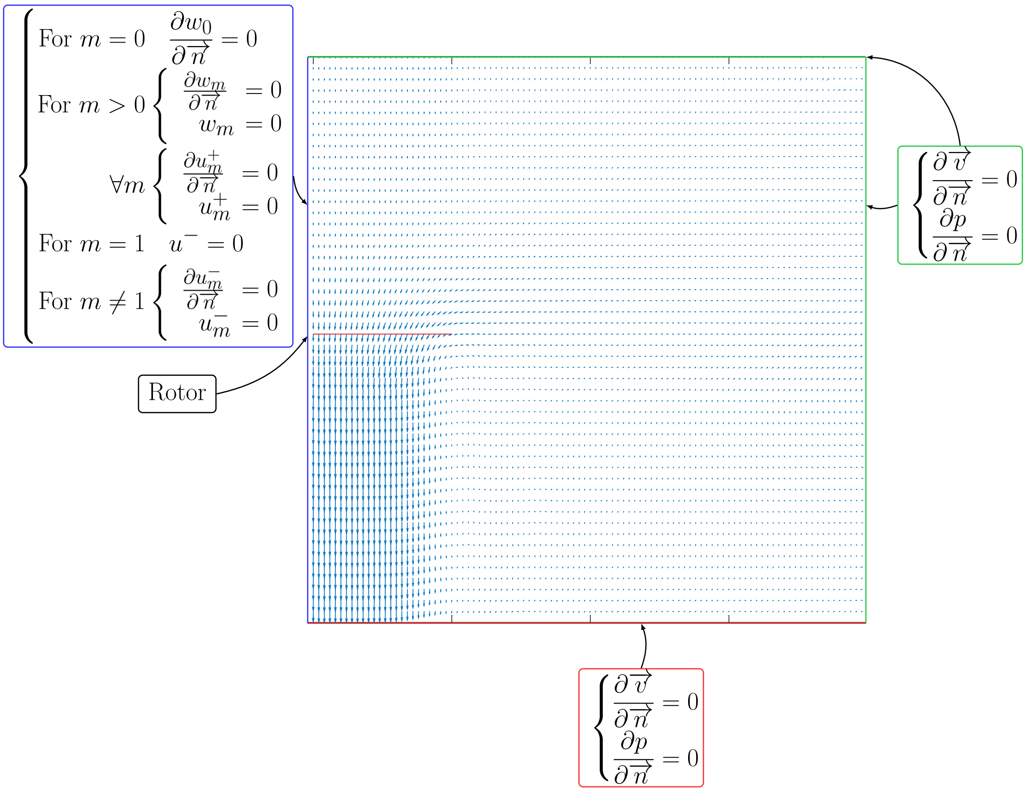


Figure 5.3.1: Boundary conditions represented in the observation plane.

Finally, the last polynomial is used for the z component of the pressure in the case of the Dong boundary condition, and respect two null Dirichlet boundary conditions at the boundaries:

$$\Lambda_n(x) = L_n(x) - L_{n+2}(x) \tag{5.3.19}$$

5.3.3.1 Note on the ground effect case

If one were to apply this method to a rotor near the ground, the boundary conditions would have to be highly modified. Two ways can be seen to deal with this case. First, considering a case entirely in ground effect, one could modify the bottom boundary condition, and thus the polynomials used. The second way would apply more generally, and allow to model the transition between no ground effect and ground effect. It would consist in adding a new term in a similar way as the one used for the Dong pressure boundary condition, but this time on the velocity, to model the right boundary condition.

5.4 Rotor definition

In the above section a time domain method allowing to conserve the divergence-free property of the velocity is described. However, including the rotor as a pressure discontinuity in the domain is not adapted to our method of treating the equations. Indeed, in the Huang and Peters model, the discontinuity was embedded in the coordinate system, but considering a discontinuity in our model would introduce unknown boundary conditions in velocity at this discontinuity, which is precisely the rotor disc. By taking another point of view on the action of the rotor on the fluid, it is possible to avoid this. The change made is to consider the disc as a collection of source terms rather than as a pressure discontinuity. More precisely, the effect of the rotor will be directly the transcription of the blade forces on the air (in reaction to the aerodynamic forces on the blades). This method is used in various actuator disc methods such as described by [6, 68]. On top of removing the need for a discontinuity in the domain, it is also a good way to describe the effect of the blades in a more direct and complete manner. Indeed, in the Peters model, only the blade lift forces are considered to calculate the pressure discontinuity on the rotor disc. Furthermore, since the Galerkin method does not rely on a grid to impose these forces, one could theoretically position the forces at the true position of the blades, therefore accounting for flap and lead-lag angles.

The way to see the action of the blades on the fluid can be illustrated in the case of a Blade Element Model. In this case each element produces forces (i.e., lift and drag), that can be seen as acting on the air at a given location. The challenge is now to represent adequately the forces with spectral elements in our equations. This is done in two steps. First, our equations will only be able to see force density, so one has to divide the forces given by the blade element model by the size of the volume where they will be applied. Then one has to project the force density on the test functions to obtain the spectral elements representing the forces.

The first step might be a problem, since the volume of the blade element might not be represented accurately by the approximation, depending on its precision. Therefore one should use the volume of the representation rather than the true volume of the blade element model. In our applications the axial size of the rotor (i.e., the thickness of the profile) is represented thanks to a Gaussian function which thickness depends on the number of axial coefficients used in the approximation. The azimuthal description of the blade is simplified to a straight line, while the radial description is described accurately. Other azimuthal descriptions could have been chosen, but in this first implementation this simple description seemed to give satisfying results, and is consistent with the lifting line blade element model used for rotorcraft flight dynamics.

The second step is simply the application of the Galerkin method to the spatial description of the forces in order to obtain spectral elements compatible with the model. Once the projection is made, the input terms can be added to the right hand side of the incompressible Euler equation.

5.5 Stability

Hyperbolic equations are at the centre of our model once the negligible viscous terms have been removed. The main problem presented by these equations is that they tend to be unstable when represented by spectral methods, and to develop discontinuities such as shocks. It is well known that these kinds of behaviours are not easily represented by spectral method because of the emergence of Gibbs phenomenon, detrimental to the accuracy of the method, that might even occur in the case of high gradients. Furthermore, in our application, the input of our model (i.e. the blade forces) will be highly localised, which will be associated to high gradients.

Many options exist in the literature in order to maintain the stability and the accuracy of the method.

Our aim in this section is first to understand the origin of the instability, and then to mitigate its effects. To do so we first study the evolution of the norm of the velocity in the case of the chosen algorithm, and then propose to use filters in order to stabilise the algorithm. We will also address the problem of the sharpness, or even discontinuity of the inputs. We then demonstrate the effectiveness of this approach.

5.5.1 Origin of the instability

In order to simplify the reasoning, we only consider one dimension approximated with N elements. Most of the derivation can be done easily in one dimension, and the non linear terms will be far more readable with only one. The generalisation to the three dimension case is straightforward, with the exception of the azimuthal component which relies on the Fourier approximation, and which has thus different properties. The conclusions nevertheless remain the same. Compacting the algorithm to remove the estimated velocity gives the following equation:

$$\frac{\partial \overrightarrow{\nu_k}}{\partial t} = -\overrightarrow{\text{grad}} p_k - 2\overrightarrow{N}_{k-1} + \overrightarrow{N}_{k-2} + \overrightarrow{f_k} \tag{5.5.1}$$

We here ignore the continuity equation and the way we obtain p_k to focus on the evolution of the norm $\|\overrightarrow{\nu_k}\|$ with time. This then gives, by taking the scalar product of the previous equation with $\overrightarrow{\nu_k}$:

$$\underbrace{\langle \frac{\partial \overrightarrow{\nu_k}}{\partial t} \mid \overrightarrow{\nu_k} \rangle}_{=\frac{\partial \|\overrightarrow{\nu_k}\|^2}{\partial t}} = \underbrace{\langle -\overrightarrow{\text{grad}} p_k \mid \overrightarrow{\nu_k} \rangle}_{:=I_P^k} - \underbrace{\langle 2\overrightarrow{N}_{k-1} + \overrightarrow{N}_{k-2} \mid \overrightarrow{\nu_k} \rangle}_{:=I_{NL}^k} + \underbrace{\langle \overrightarrow{f_k} \mid \overrightarrow{\nu_k} \rangle}_{:=I_f^k} \tag{5.5.2}$$

This approach is close to an evaluation of the variation of energy of the system, which is similar to the approaches found in [45, 53]. This allows to measure the stability of the method by assessing if the energy can diverge or not. With only the upper bound, one can say that the energy can diverge, if it is unbounded, or that the energy will converge.

By definition of the scalar product:

$$\frac{\partial \|\overrightarrow{\nu_k}\|^2}{\partial t} = -\int\limits_{\partial D_c} p_k \overrightarrow{\nu_k} \cdot \overrightarrow{n}\, dS - \int\limits_{D_c} \left(2\overrightarrow{N}_{k-1} + \overrightarrow{N}_{k-2}\right) \cdot \overrightarrow{\nu_k} dD + \int\limits_{D_c} \overrightarrow{f_k} \cdot \overrightarrow{\nu_k} dD \tag{5.5.3}$$

We can then reintroduce the actual value of the non linear term to obtain the following expression:

$$\begin{aligned}\frac{\partial \|\overrightarrow{\nu_k}\|^2}{\partial t} &= -\int\limits_{\partial D_c} p_k \overrightarrow{\nu_k} \cdot \overrightarrow{n}\, dS - \int\limits_{\partial D_c} \frac{\mid \overrightarrow{\nu_k} \mid^2}{2} \overrightarrow{\nu_k} \cdot \overrightarrow{n}\, dS \\ &+ \left[\int\limits_{\partial D_c} \frac{\mid \overrightarrow{\nu_k} \mid^2}{2} \overrightarrow{\nu_k} \cdot \overrightarrow{n} - \int\limits_{D_c} \left(2\overrightarrow{N}_{k-1} + \overrightarrow{N}_{k-2}\right) \cdot \overrightarrow{\nu_k} dD\right] + \int\limits_{D_c} \overrightarrow{f_k} \cdot \overrightarrow{\nu_k} dD\end{aligned} \tag{5.5.4}$$

We now have to find acceptable bounds for the integrals I_P^k, I_{NL}^k and I_f^k in order to find the cause(s) of divergence in the algorithm, and how to solve it. It is to be noted that the main argument of the following upper bounds originate from the spectral accuracy property. This property is summarised by equation 5.6.1.

5.5.1.1 The input forces terms

We start by using a more transparent version of the integral we have to evaluate by using the interpolation operator P_N, since all the algorithm will ever see will always be an interpolated version of the inputs:

$$\begin{aligned} I_f^k &= \langle P_N \overrightarrow{f_k} \mid P_N \overrightarrow{\nu_k} \rangle \\ &= \langle P_N \overrightarrow{f_k} - \overrightarrow{f_k} \mid P_N \overrightarrow{v_k} \rangle + \langle \overrightarrow{f_k} \mid P_N \overrightarrow{\nu_k} \rangle \end{aligned} \tag{5.5.5}$$

Then one can use the Cauchy-Schwartz inequality and equation 5.6.1 to bound the different terms of the previous equation:

$$\begin{aligned} |I_f^k| &\leq \|P_N \overrightarrow{f_k} - \overrightarrow{f_k}\| \|P_N \overrightarrow{\nu_k}\| + \|\overrightarrow{f_k}\| \|P_N \overrightarrow{v_k}\| \\ &\leq \left(C_f N^{-\sigma_f} \|\overrightarrow{f_k}\|_{\sigma_f} + \|\overrightarrow{f_k}\| \right) \|P_N \overrightarrow{\nu_k}\| \end{aligned} \tag{5.5.6}$$

Applying the same argument of spectral accuracy on the $P_N \overrightarrow{v_k}$ term gives:

$$|I_f^k| \leq \left(C_f N^{-\sigma_f} \|\overrightarrow{f_k}\|_{\sigma_f} + \|\overrightarrow{f_k}\| \right) \left(C_\nu N^{-\sigma_\nu} \|\overrightarrow{\nu_k}\|_{\sigma_\nu} + \|\overrightarrow{\nu_k}\| \right) \tag{5.5.7}$$

This shows that if the input function experience high gradients, or even is discontinuous, it can hinder the stability of the solution. The input will however not be at the origin of the divergence, as long as it is square-integrable, which should be the case of the blade input forces, unless extreme conditions are encountered.

Filtering the input is thus not a necessity, but might improve the stability of the algorithm.

5.5.1.2 The pressure terms

Following the same steps as for the input term, we have:

$$\begin{aligned} I_p^k &= \langle -P_N \overrightarrow{\mathrm{grad}}(P_N p_k) \mid P_N \overrightarrow{\nu_k} \rangle \\ &= \langle -\overrightarrow{\mathrm{grad}} p_k \mid P_N \overrightarrow{\nu_k} \rangle + \underbrace{\langle \overrightarrow{\mathrm{grad}} p_k - P_N \overrightarrow{\mathrm{grad}} p_k \mid P_N \overrightarrow{\nu_k} \rangle}_{:= E_{pI}^k} \\ &\quad + \underbrace{\langle P_N \overrightarrow{\mathrm{grad}} p_k - \overrightarrow{\mathrm{grad}}(P_N p_k) \mid P_N \overrightarrow{\nu_k} \rangle}_{:= E_{pC}^k} + \underbrace{\langle \overrightarrow{\mathrm{grad}}(P_N p_k) - P_N \overrightarrow{\mathrm{grad}}(P_N p_k) \mid P_N \overrightarrow{\nu_k} \rangle}_{:= E_{pII}^k} \end{aligned} \tag{5.5.8}$$

The three last terms of the right hand side of the last equation are all linked to different errors. Respectively they correspond to the approximation error on the gradient of the pressure, E_{pI}^k, to the commutation error between the interpolation operator and the gradient operator, E_{pC}^k, and to the interpolation error of the gradient of the interpolated pressure E_{pII}^k.

While E_{pI}^k and E_{pII}^k are simply bounded by using Cauchy-Schwartz inequality and the regularity argument as follows:

$$\begin{aligned} |E_{pI}^k| &\leq C_{pI} N^{-\sigma_{pI}} \|\overrightarrow{\mathrm{grad}} p_k\|_{\sigma_{pI}} \|P_N \overrightarrow{\nu_k}\| \\ |E_{pII}^k| &\leq C_{pII} N^{-\sigma_{pII}} \|\overrightarrow{\mathrm{grad}}(P_N p_k)\|_{\sigma_{pII}} \|P_N \overrightarrow{\nu_k}\| \end{aligned} \tag{5.5.9}$$

the commutation error E^k_{pC} uses a different argument, found in [41], that gives:

$$\begin{aligned} |E^k_{pC}| &\leq \|\overrightarrow{\text{grad}}(P_N p_k) - P_N \overrightarrow{\text{grad}} p_k\| \|P_N \overrightarrow{\nu_k}\| \\ &\leq C_{CE} N^{\frac{3}{2}-\sigma_{CE}} \|p_k\|_{\sigma_{CE}} \|P_N \overrightarrow{\nu_k}\| \end{aligned} \tag{5.5.10}$$

Thus, contrary to the input terms, these terms might generate instabilities for it is unbounded with N if the pressure is not regular enough. In practical applications for helicopters, the pressure should also be regular. Once again, extreme conditions, such as shocks, would be problematic to represent, and would notably reduce the stability of the method.

5.5.1.3 The non linear terms

This term is the one necessitating the most attention. In order to simplify the present discussion, we introduce two operators: the quadrature interpolation operator Q_N and the non linear operator NL defined as follows:

$$\text{NL}(\overrightarrow{\nu}) = \overrightarrow{\overrightarrow{\text{grad}}}(\overrightarrow{\nu}) \cdot \overrightarrow{\nu} \tag{5.5.11}$$

This leads to the following form of $I^{k,w}_{NL}$, with $N_k = Q_N(\text{NL}(P_N(\overrightarrow{\nu_k})))$:

$$\begin{aligned} I^{k,w}_{NL} &= \langle 2N_{k-1} - N_{k-2} \mid \overrightarrow{\nu_k} \rangle \\ &= \underbrace{\langle 2N_{k-1} - N_{k-2} - N_k \mid \overrightarrow{\nu_k} \rangle}_{:=E^k_{NL,\delta_t}} + \underbrace{\langle Q_N(\text{NL}(P_N(\overrightarrow{\nu_k}))) - \text{NL}(P_N(\overrightarrow{\nu_k})) \mid \overrightarrow{\nu_k} \rangle}_{:=E^k_{NL,Q}} \\ &+ \underbrace{\langle \text{NL}(P_N(\overrightarrow{\nu_k})) - P_N(\text{NL}(\overrightarrow{\nu_k})) \mid \overrightarrow{\nu_k} \rangle}_{:=E^k_{NL,C}} + \underbrace{\langle P_N(\text{NL}(\overrightarrow{\nu_k})) - \text{NL}(\overrightarrow{\nu_k}) \mid \overrightarrow{\nu_k} \rangle}_{:=E^k_{NL,P}} \\ &+ \langle \text{NL}(\overrightarrow{\nu_k}) \mid \overrightarrow{\nu_k} \rangle \end{aligned} \tag{5.5.12}$$

Thus we have divided the integral in the time estimation error E^k_{NL,δ_t}, the quadrature interpolation error $E^k_{NL,Q}$, the commutation between non linear term and projection error $E^k_{NL,C}$ and the projection error $E^k_{NL,P}$.

The error $E^k_{NL,C}$ is particularly difficult to estimate, since it is combining a multiplication commutation error and a derivative commutation error. We thus only consider one dimension for simplicity, but the results will still hold for the other components. This leads to take the following form of this error for the first component of the velocity:

$$\begin{aligned} E^k_{NL,C} = \langle & P_N u_k \frac{\partial P_N u_k}{\partial r} + P_N v_k \frac{\partial P_N u_k}{r \partial \theta} + P_N w_k \frac{\partial P_N u_k}{\partial z} \\ & - P_N u_k P_N \frac{\partial u_k}{\partial r} + P_N v_k P_N \frac{\partial u_k}{r \partial \theta} + P_N w_k P_N \frac{\partial u_k}{\partial z} \mid u_k \rangle \\ & + \langle P_N u_k P_N \frac{\partial u_k}{\partial r} + P_N v_k P_N \frac{\partial u_k}{r \partial \theta} + P_N w_k P_N \frac{\partial u_k}{\partial z} \\ & - P_N \left(u_k \frac{\partial u_k}{\partial r} + v_k \frac{\partial u_k}{r \partial \theta} + w_k \frac{\partial u_k}{\partial z} \right) \mid u_k \rangle \end{aligned} \tag{5.5.13}$$

One can here see these two scalar products as errors: a commutation error between derivation and interpolation, and a commutation error between multiplication and interpolation. The first term, the commutation error between derivation and interpolation, can be bounded with the same method as for the pressure commutation error with:

$$\begin{aligned}\langle P_N u_k \frac{\partial P_N u_k}{\partial r} + P_N v_k \frac{\partial P_N u_k}{r\partial\theta} + P_N w_k \frac{\partial P_N u_k}{\partial z} \\ - P_N u_k P_N \frac{\partial u_k}{\partial r} + P_N v_k P_N \frac{\partial u_k}{r\partial\theta} + P_N w_k P_N \frac{\partial u_k}{\partial z} \mid u_k\rangle \quad \leq C N^{\frac{3}{2}-\sigma_{NL}} \|\overrightarrow{\nu_k}\|_{\sigma_{NL}}\end{aligned} \tag{5.5.14}$$

While the second term demands slightly more work. No estimation of the error of this term was found in the literature, but from the result obtained, it may simply be trivial to bound by some other method. We nevertheless develop the line of reasoning for this bound, once again only in one dimension to simplify the discussion, the generalisation to higher dimensions being straightforward.

We begin by showing the link between the coefficients of the product of two functions, and the coefficient of each function:

$$\begin{aligned}\forall f, g \in H_{[-1,1]} \\ fg &= \sum_{k=0}^{\infty}\sum_{j=0}^{\infty} \hat{f}_k \hat{g}_j \phi_k \phi_j \\ &= \sum_{k=0}^{\infty}\sum_{j=0}^{\infty} \hat{f}_k \hat{g}_j \sum_{l=0}^{k+j} d_l^{k+j} \phi_l \\ &= \sum_{l=0}^{\infty} \underbrace{\left(\sum_{k=0}^{\infty}\sum_{j=l-k}^{\infty} \hat{f}_k\, \hat{g}_j\, d_l^{k+j}\right)}_{=\hat{fg}_l} \phi_l\end{aligned} \tag{5.5.15}$$

Then we can compute the product of the two interpolated functions:

$$\begin{aligned}\forall f, g \in H_{[-1,1]} \\ P_N f P_N g &= \sum_{k=0}^{N}\sum_{j=0}^{N} \hat{f}_k \hat{g}_j \phi_k \phi_j \\ &= \sum_{k=0}^{N}\sum_{j=0}^{N} \hat{f}_k \hat{g}_j \sum_{l=0}^{k+j} d_l^{k+j} \phi_l \\ &= \sum_{l=0}^{2N} \left(\sum_{\substack{k=l-N \\ k\geq 0}}^{N}\sum_{\substack{j=l-k \\ j\geq 0}}^{N} \hat{f}_k\, \hat{g}_j\, d_l^{k+j}\right) \phi_l\end{aligned} \tag{5.5.16}$$

We drop the notation of the non negativity of k and j in the following relations for the sake of clarity.

This leads us to the following expression for the commutation error:

$$
\begin{aligned}
P_N(fg) - P_N f P_N g = &\sum_{l=0}^{N} \left(\sum_{k=N+1}^{\infty} \sum_{j=l-k}^{\infty} \hat{f}_k \, \hat{g}_j \, d_l^{k+j} \right) \phi_l \\
&- \sum_{l=N+1}^{2N} \left(\sum_{k=l-N}^{N} \sum_{j=l-k}^{N} \hat{f}_k \, \hat{g}_j \, d_l^{k+j} \right) \phi_l
\end{aligned}
\tag{5.5.17}
$$

The norm of the second term can be bounded by the product of the interpolation error of f and of g:

$$
\| \sum_{l=N+1}^{2N} \left(\sum_{k=l-N}^{N} \sum_{j=l-k}^{N} \hat{f}_k \, \hat{g}_j \, d_l^{k+j} \right) \phi_l \|^2 \leq \|f - P_N f\|^2 \|g - P_N g\|^2 \tag{5.5.18}
$$

For the first term, we need to evaluate the evolution of d_l^{k+j}, which can be found in [69], for Legendre polynomials:

$$
\begin{aligned}
d_l^{k+j} &= \int_{-1}^{1} \phi_k(x)\phi_j(x)\phi_l(x)dx \\
&= \begin{cases} 0, & \text{if } l+k+j \text{ odd} \\ (-1)^s \sqrt{\frac{(2s-2k)!(2s-2j)!(2s-2l)!}{(2s+1)!}} \frac{s!}{(s-k)!(s-j)!(s-l)!} & \text{if } l+k+j \text{ even, with } s = \frac{k+j+l}{2} \end{cases}
\end{aligned}
$$

Using this in the evaluation of the norm of the first term, we found:

$$
\| \sum_{l=0}^{N} \left(\sum_{k=N+1}^{\infty} \sum_{j=l-k}^{\infty} \hat{f}_k \, \hat{g}_j \, d_l^{k+j} \right) \phi_l \|^2 \leq \sum_{l=0}^{N} \left(\sum_{k=N+1}^{\infty} \sum_{j=l-k}^{\infty} \hat{f}_k^2 \, \hat{g}_j^2 \, \left(\frac{1}{N!} \right)^2 \right) \|\phi_l\|^2 \tag{5.5.19}
$$

Leading to the final estimation of the commutation error:

$$
\|P_N(fg) - P_N f P_N g\| \leq \frac{C}{N!} \|f\| \, \|g\| + C N^{-\sigma_1 - \sigma_2} \|f\|_{\sigma_1} \|g\|_{\sigma_2} \tag{5.5.20}
$$

Which shows this commutation error is unlikely to be unstable for regular functions.

The use of a second order approximation in time for the non linear term gives for E^k_{NL,δ_t}:

$$
|E^k_{NL,\delta_t}| \leq C_{\delta_t} \delta_t^2 \|\overrightarrow{\nu_k}\| \tag{5.5.21}
$$

The error of the quadrature $E^k_{NL,Q}$ can be taken to be null if one choose a high enough order, since we simply approximate interpolated function, thus product of derivatives.

And finally the regularity of the velocity gives:

$$
|E^k_{NL,P}| \leq \|\mathrm{NL}(\overrightarrow{\nu_k}) - P_N(\mathrm{NL}(\overrightarrow{\nu_k}))\| \|\overrightarrow{\nu_k}\| \leq C N^{-\sigma_{NL}} \|\mathrm{NL}(\overrightarrow{\nu_k})\|_{\sigma_{NL}} \|\overrightarrow{\nu_k}\| \tag{5.5.22}
$$

By definition of the non linear operator, one can thus conclude:

$$
|E^k_{NL,P}| \leq C N^{-\sigma_{NL}} \|\overrightarrow{\nu_k}\|_{(\sigma_{NL}-1)} \, \|\overrightarrow{\nu_k}\|_{\sigma_{NL}} \, \|\overrightarrow{\nu_k}\| \tag{5.5.23}
$$

Finally, we obtain the following bound on non linear term:

$$
I^{k,w}_{NL} \leq C N^{\frac{3}{2} - \sigma_{NL}} \|\overrightarrow{\nu_k}\|_{\sigma_{NL}} + C_{\delta_t} \delta_t^2 \|\overrightarrow{\nu_k}\| + C N^{\sigma_{NL}} \|\overrightarrow{\nu_k}\|_{\sigma_{NL}} \|\overrightarrow{\nu_k}\|_{\sigma_{NL}-1} \|\overrightarrow{\nu_k}\| \tag{5.5.24}
$$

Which leads to the conclusion that the non linear terms can also be a cause of divergence. Indeed, $E^k_{NL,C}$ is unbounded with respect to N, mainly because of the commutation error between derivation and interpolation. Furthermore, one can also see the dependence on the time step, which is satisfyingly of second order, underlying that the error introduced by the explicit scheme used to simplify the integration of the non linear terms is acceptable.

5.5.1.4 Conclusion on the norm estimations

To conclude with these norm estimations, it has been shown that the instability stems from the non linear and gradient terms of the momentum conservation equation. Furthermore, the regularity of the velocity, as well as of the inputs can highly influence the rate of convergence. This highlights the fact that it is required to modify the algorithm in order to maintain its stability, while maintaining sufficient accuracy to represent the phenomena of our application.

5.5.2 Stabilising the algorithm

In order to maintain the stability of spectral methods treating hyperbolic equations during the whole length of a simulation, many different tools have been developed. Filters are a case of particular interest, since they are easy to implement in the case of a spectral method, and act directly on the coefficients of the approximation. Furthermore, their application can be done efficiently. Some methods are more promising and precise, but are mainly tailored for post treatment (see [51]), or are far too computationally heavy (see [50]).

In the next section, we will present what filters are, how to use them and their impact on the results of the approximation.

5.5.2.1 Filters

Filters of order p are defined by Vandeven in [52] as a function σ of the order of the approximation coefficient, having the following properties:

$$\begin{cases} \sigma(0) = 1 \\ \quad \forall l, 1 \leq l \leq p-1,\ \sigma^{(l)}(0) = 0 \\ \quad \forall l, 1 \leq l \leq p-1,\ \sigma^{(l)}(1) = 0 \end{cases} \tag{5.5.25}$$

And being applied as follows to the spectral approximation of a function u, with the filtered interpolation operator $\mathfrak{F}_N$:

$$\mathfrak{F}_N(u)(x) = \sum_{n=0}^{N} \sigma\left(\frac{n}{N}\right) \hat{u_n} \phi_n(x) \tag{5.5.26}$$

The main effect of a filter is to improve the convergence rate of the coefficients of the approximation, which can be seen as introducing some viscosity or damping into the equations (see [53] for a detailed explanation). It makes sense in terms of maintaining the stability of the algorithm to introduce a term dissipating the surplus of energy.

An interesting point of view on filters, is to see them as modification of the kernel (see appendix E.1).

The choice of the filter will therefore be made in order to maintain a decent accuracy, while ensuring stability.

5.5.2.2 Effect of filters on the commutation error

As shown during the estimation of the evolution of the norm of the velocity, the main hindrance to the rate of convergence of the algorithm lies in the commutation error between interpolation and derivation. In this section we will evaluate this error when filters are used, and measure their effects.

We first introduce the filtered approximation of a function g:

$$\mathfrak{F}_N(g)(x) = \sum_{n=0}^{N} \sigma\left(\frac{n}{N}\right) \hat{g_n}\phi_n(x) \tag{5.5.27}$$

with σ a filter.

The error of interest is therefore:

$$I_{\partial\mathfrak{F}_N}(g)(x) = \mathfrak{F}_N\left(\frac{\partial g}{\partial x}\right)(x) - \frac{\partial}{\partial x}\left(\mathfrak{F}_N(g)(x)\right) \tag{5.5.28}$$

Using the fact that:

$$\frac{\partial g}{\partial x}(x) = \sum_{n=0}^{\infty} \hat{\partial g}_n\phi_n(x) = \sum_{n=0}^{\infty} \hat{g}_n \frac{\partial \phi_n}{\partial x}(x) \tag{5.5.29}$$

and:

$$\frac{\partial \phi_n}{\partial x}(x) = \sum_{l=0}^{n-1} d_l^{\phi,n}\phi_l(x) \tag{5.5.30}$$

In the case of normalised Legendre polynomials:

$$d_l^{L,n} = \begin{cases} 2\dfrac{\sqrt{2l+1}}{\sqrt{2n+1}} & \text{for } n+l \text{ odd}, l \leq n \\ 0 & \text{otherwise} \end{cases} \tag{5.5.31}$$

we have, by inversion of the sums:

$$\hat{\partial g}_l = \sum_{k=l+1}^{\infty} \hat{g}_k d_l^{\phi,k} \tag{5.5.32}$$

Then, one can expand the terms of $I_{\partial\mathfrak{F}_N}(g)$ to obtain:

$$\begin{aligned} I_{\partial\mathfrak{F}_N}(g)(x) &= \sum_{l=0}^{N}\sum_{k=l+1}^{\infty} \sigma\left(\frac{l}{N}\right)\hat{g}_k d_l^{\phi,k}\phi_l(x) - \sum_{l=0}^{N-1}\sum_{k=l+1}^{\infty} \sigma\left(\frac{k}{N}\right)\hat{g}_k d_l^{\phi,k}\phi_l(x) \\ &= \sum_{l=0}^{N-1}\left[\sum_{k=l+1}^{N}\left(\sigma\left(\frac{l}{N}\right) - \sigma\left(\frac{k}{N}\right)\right)\hat{g}_k d_l^{\phi,k}\phi_l(x)\right] \\ &\quad + \sum_{l=0}^{N}\sum_{k=N+1}^{\infty} \sigma\left(\frac{l}{N}\right)\hat{g}_k d_l^{\phi,k}\phi_l(x) \end{aligned} \tag{5.5.33}$$

One can see that the usual term of the interpolation error is found, but is filtered, which does show that the filter works as intended. However an additional term, that is usually null, is added, which seems here to compare the filter at different points. This term can however be dealt with, by using the Taylor formula on σ and the fact that the first p derivatives of σ are null at 0. This gives:

$$\begin{aligned}\sigma\left(\frac{l}{N}\right)-\sigma\left(\frac{k}{N}\right) &= \int_0^{\frac{l}{N}} \frac{\sigma^{(p+1)}(x)}{(p+1)!}x^{p+1}dx - \int_0^{\frac{k}{N}} \frac{\sigma^{(p+1)}(x)}{(p+1)!}x^{p+1}dx \\ &= \frac{1}{N^{p+1}}\int_k^l \frac{\sigma^{(p+1)}(\frac{t}{N})}{(p+1)!}t^{p+1}dt\end{aligned} \tag{5.5.34}$$

This allows to minimise the norm of the first term and to see that its accuracy is limited by the order p of the filter. It however shows that filtering might introduce unwanted errors if the order of the filter is too low.

The filtered commutation error remains to be bounded:

$$E_F = \sum_{l=0}^{N}\sum_{k=N+1}^{\infty} \sigma\left(\frac{l}{N}\right)\hat{g}_k d_l^{\phi,k}\phi_l(x) \tag{5.5.35}$$

Similarly to the unfiltered case, one can bound this part as follows, first by assuming that $d_l^{\phi,k} = d_l^{\phi,0}d_0^{\phi,k}$, then by using the spectral accuracy bound on the derivative of g:

$$\begin{aligned}\|E_F\|^2 &\le \left\|\sum_{l=0}^{N}\sigma\left(\frac{l}{N}\right)d_l^{\phi,0}\phi_l(x)\right\|^2 \frac{1}{N}\left\|\sum_{k=N+1}^{\infty}\hat{g}_k d_0^{\phi,k}\right\|^2 \\ &\le CN^{2(1-s)-1}\|g\|_s^2\sum_{l=0}^{N}\sigma\left(\frac{l}{N}\right)^2\frac{2l+1}{2}\end{aligned} \tag{5.5.36}$$

The last line of the previous equation assumes the case of Legendre polynomials. Without accounting for the filtering, one can find the usual $N^{\frac{3}{2}-s}$ order for the Legendre case.

The filter does have a positive influence on the commutation error. However, in order to truly remove the negative effects of this error, one would have to adapt the order of the filter with the value of N. This prevents to truly conclude on the theoretical effectiveness of the filters. Nevertheless, application results are much more encouraging.

5.5.2.3 Effect of filters

Fig 5.5.1 shows the evolution of the states of azimuthal order $m = 0$ for the axial velocity, in a unfiltered case. The darker the colour of the line, the higher the order of the coefficient is. One can here see the development of the high order coefficients and their impact on the stability of the algorithm.

The literature contains numerous different types of filters for our purpose, such as:

$$\sigma_1(\eta) = \exp\left(-\alpha\eta^{2p}\right) \tag{5.5.37}$$

or

$$\sigma_2(\eta) = \exp\left(\left(\frac{\eta^2}{\eta^2-1}\right)^{2p-1}\right) \tag{5.5.38}$$

Where p is an integer defining the order of the filter and $\alpha = -log(\epsilon)$ (where ϵ is the machine zero). These exponential filters have the advantage of being adaptable to our

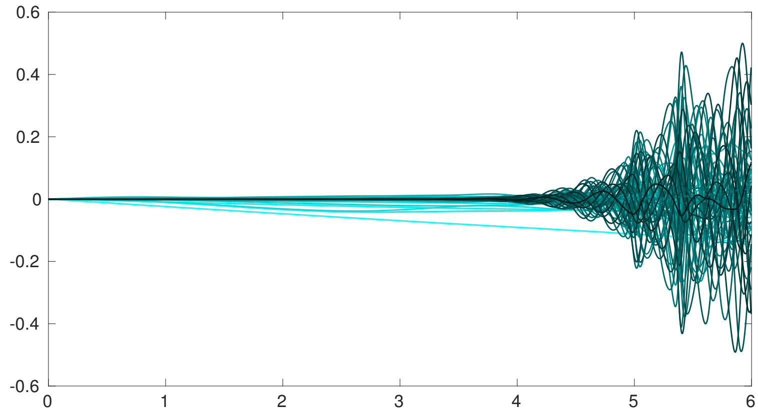


Figure 5.5.1: Evolution with time of the coefficients of the axial velocity without any filter, for 6000 time steps, with $\delta_t = 1E-3$.

needs, with a parameter p and to be simple to compute and to apply. The evolution of the filter σ_1 with p, that we will refer to hereafter as the exponential filter, is presented on Fig 5.5.2. It can be seen that the main function of the exponential filter is to filter mainly the high order terms, and that its order p selects the amount of element being filtered. This is coherent with the observation made above on the study of the evolution of the norm of the velocity and on the observation made in Fig 5.5.1.

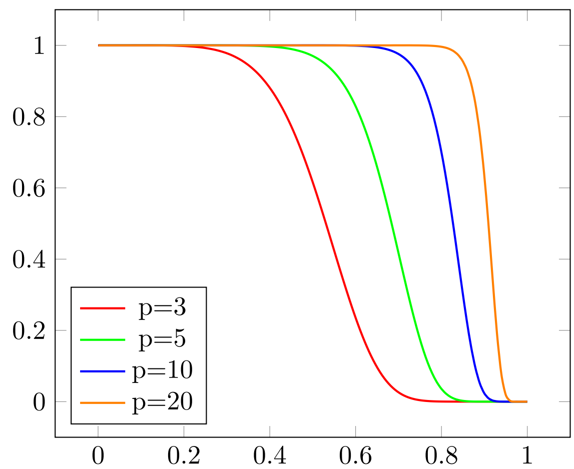


Figure 5.5.2: Impact of the order p on the shape of the exponential filter.

To compare the improvement provided by the filter, we plotted on Fig 5.5.3 the evolution of a low order coefficient of the axial velocity $w^0_{0,0}$, for different values of p[3]. With decreasing order of the filter (and thus stronger filtering), one can see that the value of this coefficient converges to the same value. This shows that the impact of the filter does not hinder the evolution of the low order coefficients.

However for a high order coefficient $w^0_{19,29}$, as is shown on Fig 5.5.4, the impact of the filter is important. For low and high values of the order p, the coefficient is completely filtered. Only the order $p = 20$ is represented here, and it remains null. Higher order of filtering allows it to be expressed but if not enough filtering is used, the coefficient will

[3]Here, $p = \infty$ corresponds to a case without filter, by extension of the exponential formula.

diverge.

Finally for a coefficient of medium order, such as $w^0_{9,14}$ presented on Fig 5.5.5, the time evolution is impacted by the presence of filter in a good manner, avoiding diverging oscillations. However one can see that the impact of a low order of filtering is too important and makes the value of the coefficient irrelevant.

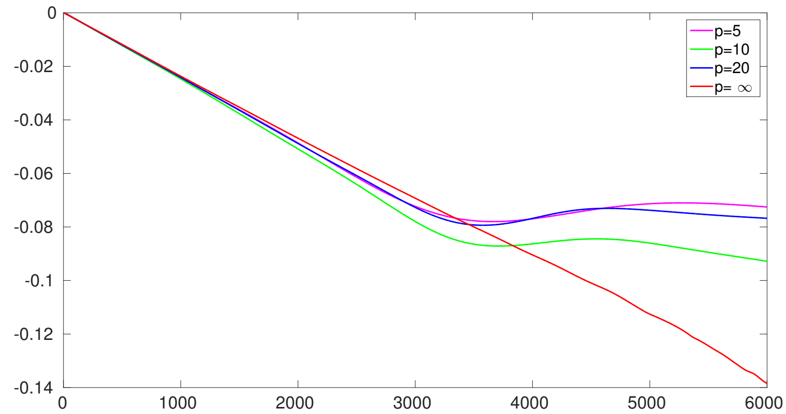


Figure 5.5.3: Evolution with time of $w^0_{0,0}$ for different values of the order p of an exponential filter.

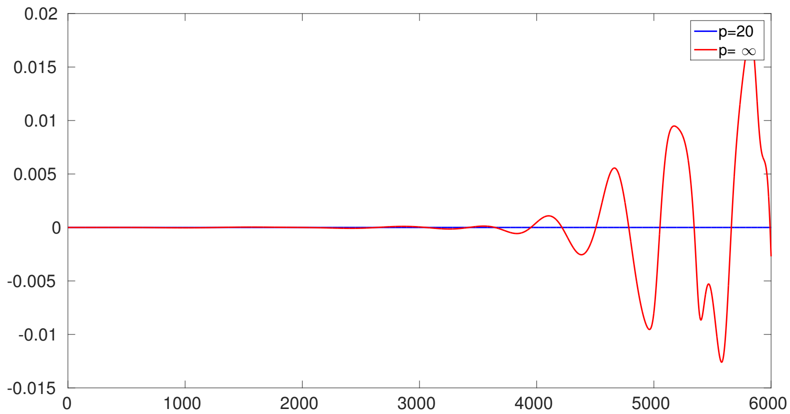


Figure 5.5.4: Evolution with time of $w^0_{19,29}$ for different values of the order p of an exponential filter.

5.5.2.4 Conclusion on filters

The use of filters is indispensable in the current version of the algorithm. Although it seems hard to ensure that they will solve the problem for any situations, they show very good results, at a low computational cost, and for a wide range of cases.

There are some downsides to this approach, however. For example, it tends to introduce some damping in the equations, as shown in [53], which might lead to undesirable effects, and to smooth some details of the induced flow. Another concern, raised in [54], is the fact that the application of the same filter at each time step will eventually lead to an effect much more important than the anticipated effect of the simple filter. The reasoning is that

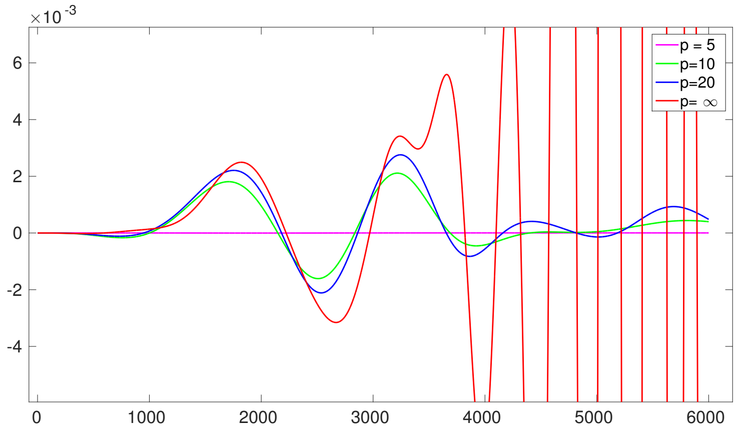


Figure 5.5.5: Evolution with time of $w^0_{9,14}$ for different values of the order p of an exponential filter.

the cumulative effect of the filter over N time step is similar to the application of a filter of the form σ^N.

5.6 Accuracy

In a spectral method, the accuracy of the solution is ensured by the so called spectral accuracy. This is a property of the spectral approximation that ensure that for regular enough function to approximate, the approximation will converge faster than any power of the number of elements in the approximation. In the case of the Legendre approximation, with P_N the Legendre interpolation operator of order N, [41] shows that:

$$\|u - P_N u\| \leqslant C N^{-\sigma} \|u\|_\sigma \tag{5.6.1}$$

Where C is a constant, $\|.\|$ is the norm of the $L^2_{[-1,1]}$ space, $\|.\|_\sigma$ is the norm of the Hilbert space $H^\sigma_{[-1,1]}$, and u is the function to be approximated (see appendix C for the functional spaces definitions and notations).

As one can see on equation 5.6.1, the precision of the solution converges faster for more regular approximated function, and in the extreme case of discontinuous functions, convergence can be really slow. The main example of this slow convergence is found in the famous Gibbs phenomenon. Many methods have been developed to counter the oscillations generated by this phenomenon and to improve the convergence rate of the representation in discontinuous cases. In our case, the main source of discontinuity emerges from the inputs of the algorithm. Indeed, the ideal thin disc, and the sharpness of the blades are detrimental to the accuracy of the spectral method. This also impacts the convergence of the algorithm, since the oscillations tend to introduce unwanted behaviours. In order to ease the convergence of the algorithm and to improve its accuracy in the case of sharp blade forces inputs, a solution proposed by the literature is the use of filters.

To solve the problem of the Gibbs phenomenon, the filters proposed in the literature are numerous. A good review can be found in [52]. In our case we mainly use two filters.

The Fejèr filter, which is defined as follows:

$$\sigma_F(\eta) = 1 - \eta \tag{5.6.2}$$

And the Lanczos filter, defined as:

$$\sigma_L(\eta) = \frac{\sin(\pi\eta)}{\pi\eta} \tag{5.6.3}$$

Their behaviour is plotted on Fig 5.6.1. One can notice that they are much stronger than the filters presented in the previous section.

They aim at smoothing the inputs in order to avoid the problematic consequences of the Gibbs phenomenon. Their effect is demonstrated on Fig 5.6.2. This figure presents the approximation of a Gaussian function and the effect of the filters. It was made however with only 20 terms in the approximation, and thus one can see that the removal of the Gibbs oscillations is made at the cost of the accuracy of the representation. However, Fig 5.6.3 shows the same function being approximated, but with 60 terms in the representation. Once again the Gibbs phenomenon is obliterated, but this time the accuracy of the filtered approximation is much better.

Two conclusions can be drawn of these examples. First, the best way to improve the accuracy of the method is still to increase the number of terms in the approximation. However, the filtering has a smaller impact on the accuracy with greater number of elements, while still removing the spurious oscillations.

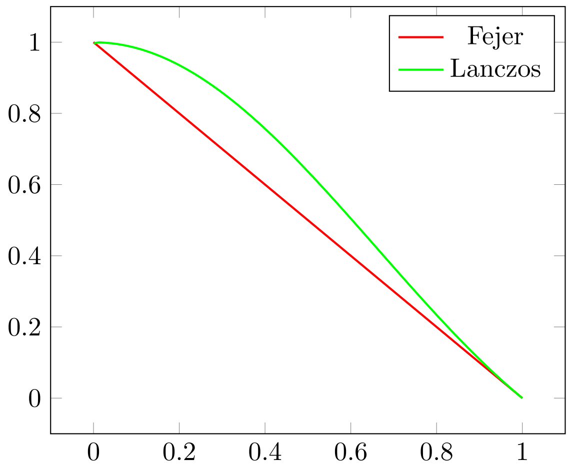


Figure 5.6.1: Representation of the Fejèr and Lanczos filters.

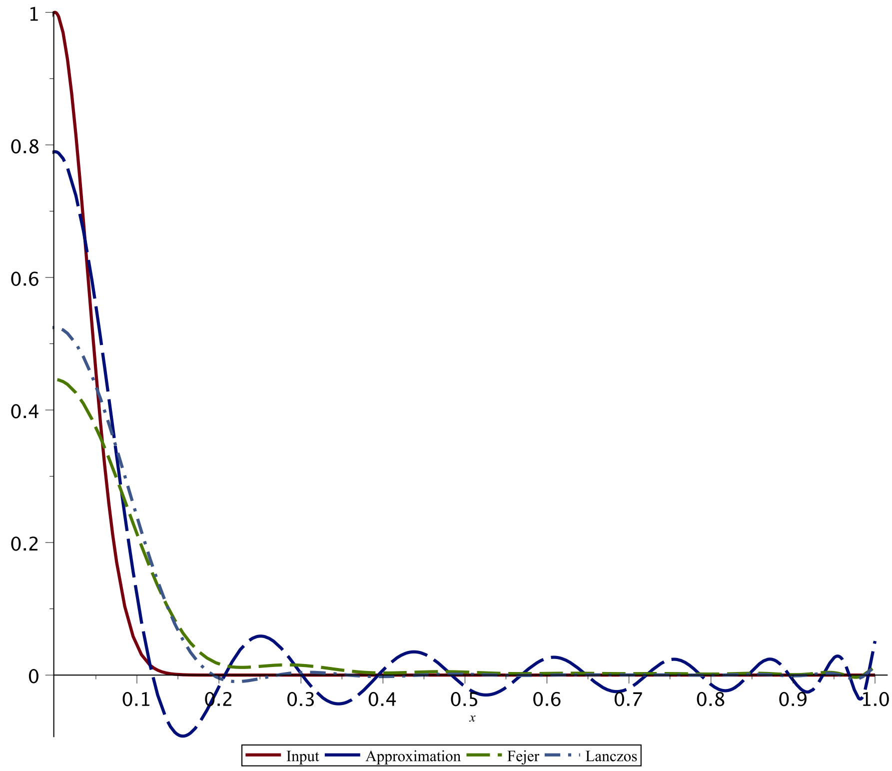


Figure 5.6.2: Approximation of a Gaussian function with 20 elements and different input filters.

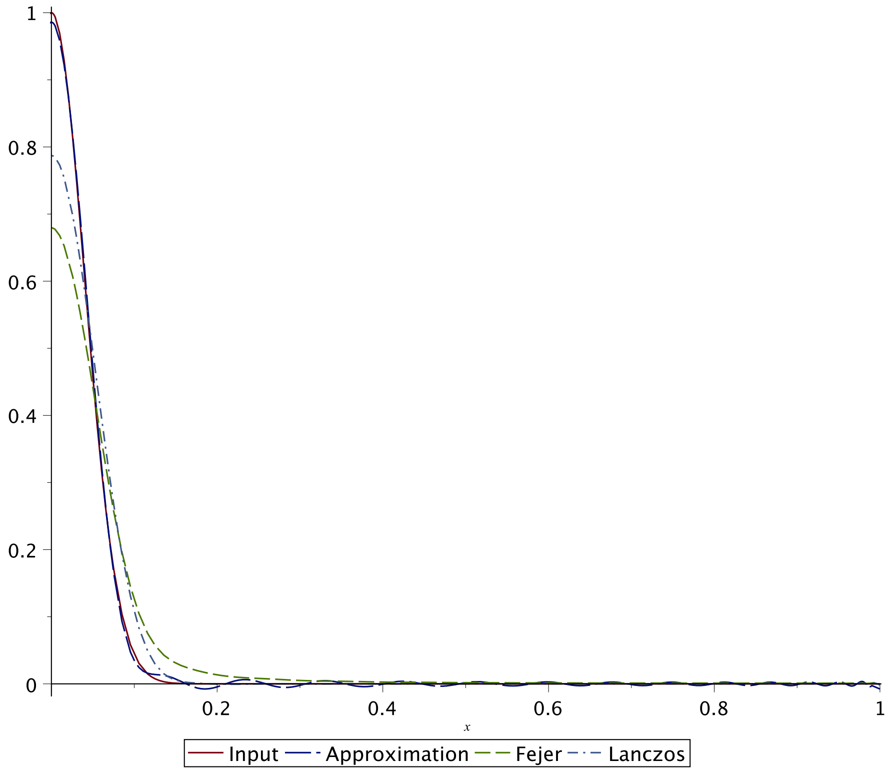


Figure 5.6.3: Approximation of a Gaussian function with 60 elements and different input filters.

Huang and Peters Model		New Model	
Inputs as pressure discontinuity	τ_0^1	Force density sources	$\sqrt{1-(x+1)^2}$ for $x \in [-1, 0]$
Number of elements	$m_{odd} = 20$, $m_{even} = 10$, Total: 157	Number of elements	$N_r = 50$, $N_j = 50$, $N_\theta = 20$[4], Total: 50000
Conditions	$\omega = 0$, $\chi = 0$	Conditions	$V_\infty = 1.0$, $\Lambda = 5$, $\delta_t = 2E-3$
		Filter order	$p = 20$

Table 5.7.1: Input data for comparison with Huang and Peters model

5.7 Comparison with the Huang and Peters model

In this section we compare our new model with Huang and Peters model. We first explain the principles of the comparison, then present the results and discuss them.

5.7.1 Comparing what is comparable

The main problem when trying to compare both models comes from the difference between the two models, one being in the frequency domain and the other in the time domain, one taking its inputs as pressure gradient, the other as forces.

It was therefore decided to choose a constant input, so as to compare a $\omega = 0$ case, and simply to wait for the convergence of the time domain model. Then the shape of the axial input force of the new model was chosen to match the τ_0^1 input of Huang and Peters model. Finally, in order to represent the fact that the Huang and Peters model considers the V_∞ to be large compared to the induced velocity, a non dimensional freestream velocity was chosen with a value of 1 in the new model.

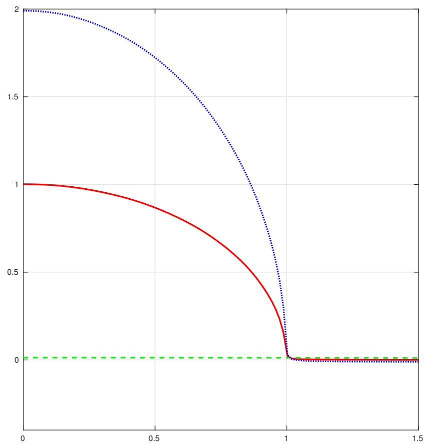
Table 5.7.1 presents the conditions in which both models where taken to make the comparisons.

5.7.2 Results and discussion

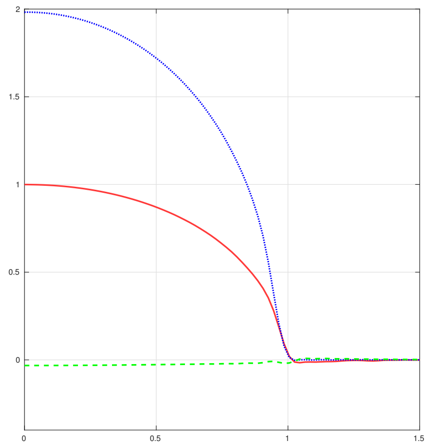
Figure 5.7.1 and 5.7.2 present the comparison for $\chi = 0°$ of the axial and radial velocities respectively. Since the case is axisymmetric, only half the plane is represented. One can see the comparison of the results given by both models on the rotor disc, far upstream and far downstream.

One can see good agreement between the two models for the axial velocity, which is what would be expected, since we are in the validity domain of the Huang and Peters model. The use of the ellipsoidal coordinates however allows a much sharper transition between the on disc and off disc regions. The contraction of the wake is almost not visible in both cases because of the high value of the free stream velocity. Although the tendencies are also similar for the radial velocity, the discrepancies appearing are to be expected because of the huge difference in the form of the inputs. While the axial inputs are tailored to give a comparable description, the radial inputs are left untouched. This is due to the different vision of the inputs. Huang and Peters model considers a discontinuity of pressure as input, which is deduced from the lift distribution, but that impact the radial component directly,

[4]For the axisymmetric case we took $N_\theta = 1$



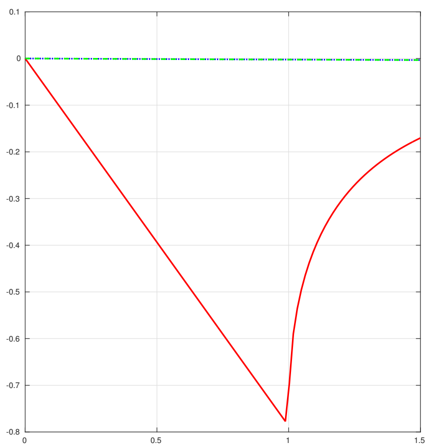
(a) Axial velocity given by Peters and Huang's model



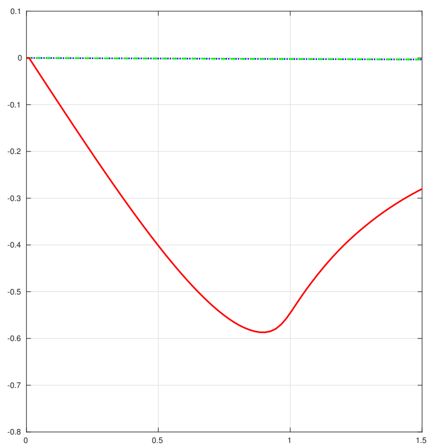
(b) Axial velocity given by the proposed new model

Figure 5.7.1: Comparison of the axial velocity between Peters and Huang's model and the proposed new model for a τ_0^1 load, as presented in [2]. The green dashed line is taken $2R$ above the rotor, the red continuous one on the disc, and the blue dotted one $2R$ below the disc

while the radial component of the new model is here only generated by the coupling of the equations, since the input forces only act on the axial component of the velocity.



(a) Radial velocity given by Peters and Huang's model



(b) Radial velocity given by the proposed new model

Figure 5.7.2: Comparison of the radial velocity between Peters and Huang's model and the proposed new model for a τ_0^1 load, as presented in [2]. The green dashed line is taken $2R$ above the rotor, the red continuous one on the disc, and the blue dotted one $2R$ below the disc

Figure 5.7.3 and 5.7.4 present the comparison for $\chi = 45°$ on the disc, for axial and radial velocities respectively. The same observation as previously can be made: that is to say that the sharp features given by the ellipsoidal coordinates are smoothed in the new model. Furthermore a discrepancy appears in the description of the velocity out of the disc. This may be linked to the differences in the boundary conditions, and domain descriptions between models.

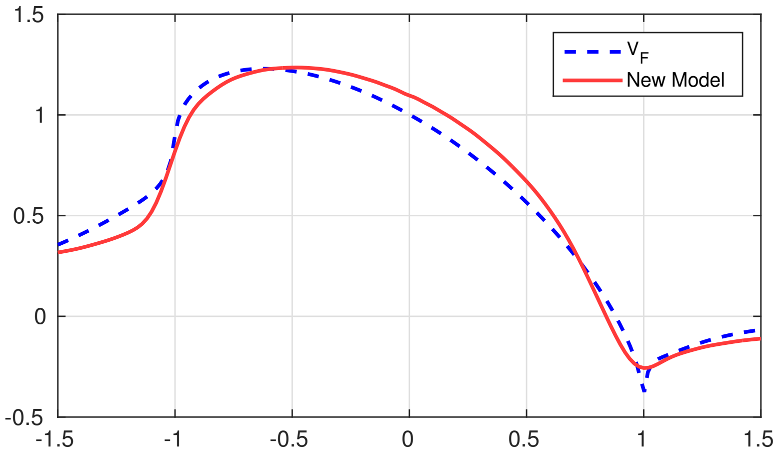


Figure 5.7.3: Comparison of the axial induced velocity on the disc for $\chi = 45°$. The dashed blue line is given by Peters and Huang model, and the red line is given by the proposed new model.

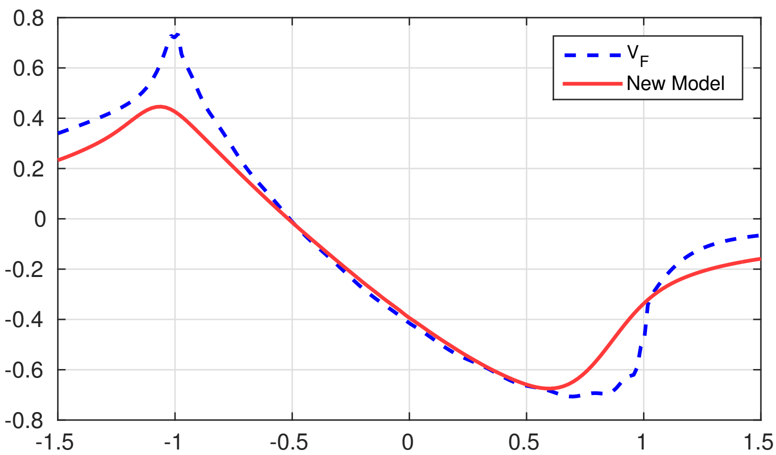


Figure 5.7.4: Comparison of the radial induced velocity on the disc for $\chi = 45°$. The dashed blue line is given by Peters and Huang model, and the red line is given by the proposed new model.

This part of the comparison shows the capacity of the new model to at least give coherent results with the Huang and Peters model in its validity domain. We will further explore its capacities in the following sections to demonstrate its interest.

5.8 Demonstration of capabilities

In this section we test our model on academical cases that would be out of the validity domain of the Huang and Peters model, even with non linear extensions. This aims to demonstrate the capabilities of the new model, and its relevance in the domain of dynamic inflow modelling. In these cases, the loading of the disc is constant and uniform. Only the flight conditions are changed.

5.8.1 Descending flight

For this case, we consider the free stream velocity to have a wake skew angle of 180°. This case is relevant in comparison with the Huang and Peters model, since the wake angle of the model is limited to the $0° - 90°$ range.

Fig 5.8.1 presents the flow field obtained around the rotor once the algorithm is converged. One can see a satisfying representation of the velocity, and in blue lines, the representation of the streamlines emanating from the rotor disc plane. It is to be notice that on this figure the induced velocity and the free stream velocity are plotted at the same time (by addition).

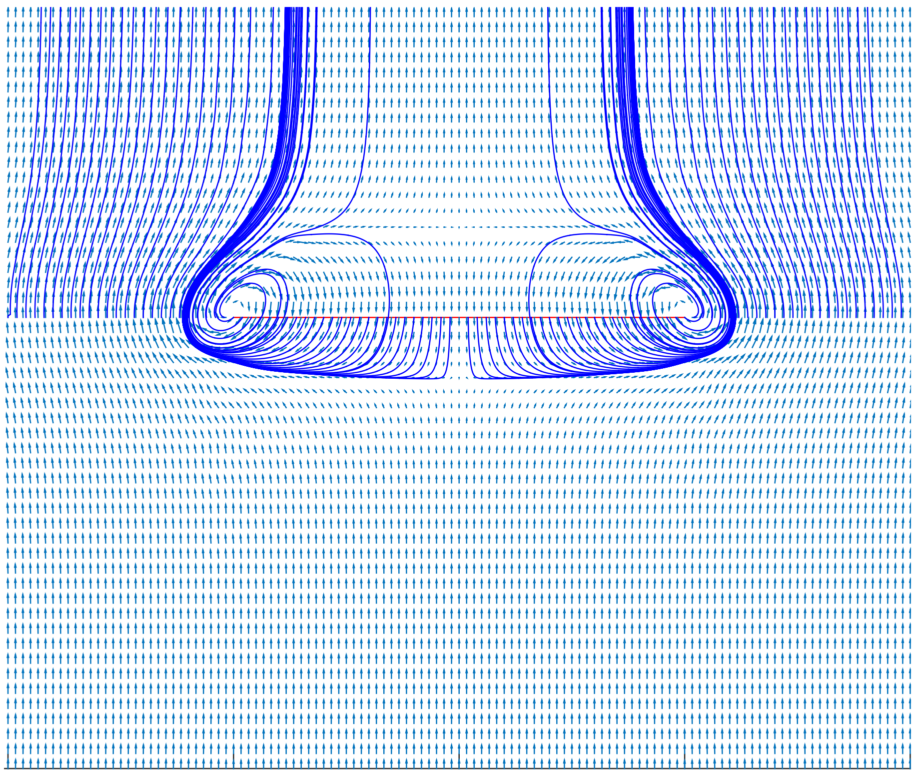


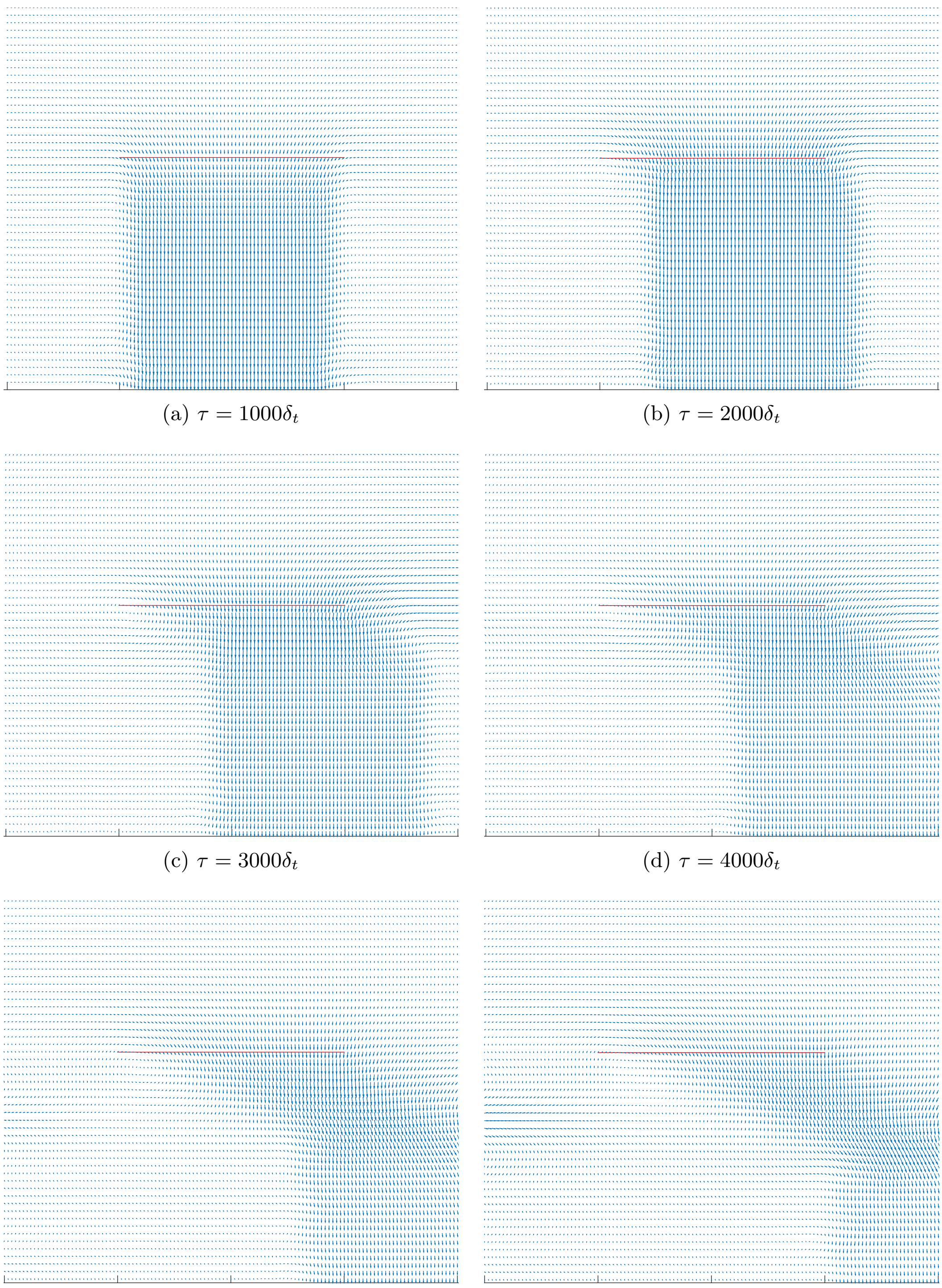
Figure 5.8.1: Induced velocity plus free stream velocity in the case of a descending rotor. The red line represents the rotor, and the blue lines represent the streamlines emanating from the rotor disc plane.

5.8.2 Transition flight

One of the downside of the Huang and Peters model is the rigidity of the representation of the wake, due to the straight rectilinear streamline assumption. Although some effects of wake deformation can be accounted for on the disc thanks to a non linear extension (see section 3.3.4.3), no deformation of the wake is applied in Huang and Peters model.

In this case we have let the wake for an axial flight converged, and then made the rotor instantaneously transition to a forward flight condition. Practically, this was made by modifying the angle of the free stream velocity from $0°$ to $90°$, without changing its norm. The resulting evolution of the wake is presented on Fig 5.8.2. On this figure one can see that the wake deforms as would be expected, while the previous hover part of the wake is convected downstream towards the boundary of the domain.

On the final snapshot of Fig 5.8.2 one can notice some spurious velocities on the left boundary of the domain. These velocities are generated by a Gibbs phenomenon in the azimuthal description of the wake, which underlines the facts that not enough azimuthal elements were accounted for, and that a stronger azimuthal filtering should be used.



(a) $\tau = 1000\delta_t$

(b) $\tau = 2000\delta_t$

(c) $\tau = 3000\delta_t$

(d) $\tau = 4000\delta_t$

(e) $\tau = 5000\delta_t$

(f) $\tau = 6000\delta_t$

Figure 5.8.2: Evolution of the induced velocities in the $y = 0$ plane, with $N_r = 30$, $N_j = 30$ and $N_\theta = 20$.

5.9 Comparison to a prescribed and free wake model

In this section we compare the new model to a prescribed wake and to a free wake model. To do so we couple the model to AMB, which already possesses the required models and rotor descriptions. All the comparisons presented below will be made on the 7A and 7AD rotors which are well documented ONERA wind tunnel rotors.

5.9.1 Integration of the model to AMB

As for the integration of the Huang and Peters model, the model was coded as an independent Python module in order to validate it against our Matlab implementation in the time domain (from which the above results are generated), then integrated as a daughter class of the *disque_rotor* class (see section 4.1.4.1 for AMB description).

The architecture of the new model is presented through the UML Class diagram of Fig 5.9.1[5]. In this diagram, one can see that the implementation of the new model has been slightly changed in comparison with the implementation of Huang and Peters temporal version. Most of the classes are in fact storing classes. The *FileInit* class reads the input data from a file, and stores it, while computing the required filters, depending on the choice of the user. From the input information the *States* class, which stores the coefficients of all the approximated quantities, can be initialised, as well as the *Quad* class, which stores all information related to the quadrature. The polynomials are then initialised, and their values at the quadrature points are stored. They are indeed required for the quadrature of the non linear terms. This allows to compute all the matrices required in the *Matrices* class.

Finally all these data are aggregated in the *NPGIVM* class, which possesses the three main functions of the method. The *time_step* method implements the time marching algorithm presented above. The *lgl_quad* deals with the computation of the non linear terms thanks to the Legendre-Gauss-Lobatto quadrature. Finally, the *velocity_point* method allows to compute the value of the velocity at any given point in space.

[5]The majority of the matrices name of the *Matrices* class have been removed for clarity

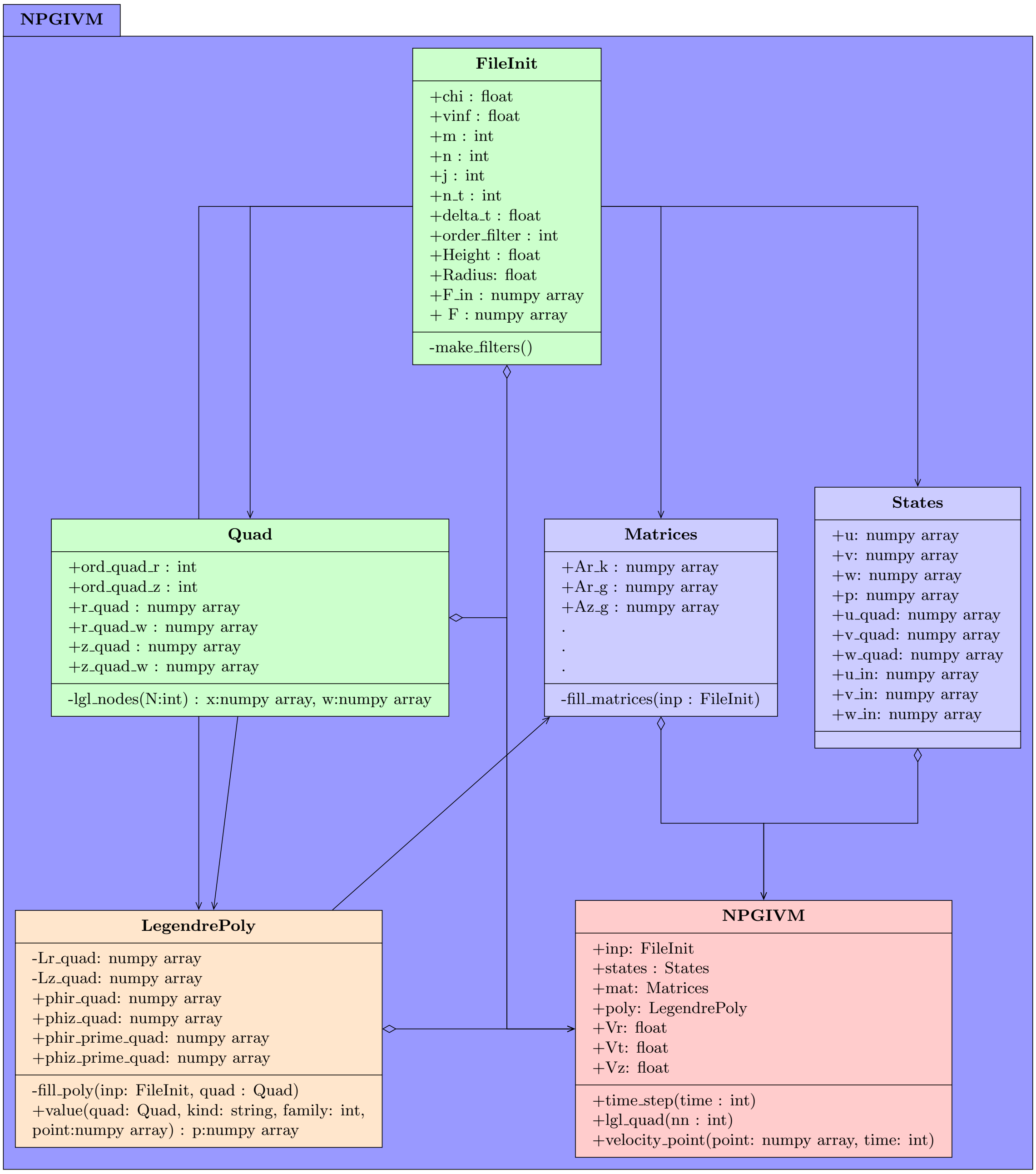


Figure 5.9.1: UML Class Diagram of the new model.

5.9.2 Computing the inputs of the model

Similarly to the case of Huang and Peters model, one needs to translate the raw blade aerodynamic forces input of the blade element model. In the specific case of the AMB coupling, the forces are localised at the centre of each blade element, on the quarter line chord. Since these information are highly localised, they would be badly represented by our method. We thus decide to apply a force density to a volume of the computational domain tailored to the blade volume. The main issue with this point of view is the axial dimension of the blade which is very thin when compared with the size of the domain. It is therefore represented thanks to a Gaussian function. However depending on the number of elements used in the axial direction, even a smooth Gaussian function can be too sharp for the spectral method. Therefore the shape of the Gaussian function is tailored to the number of axial element rather than to the real thickness of the blade. Thus the function used is:

$$f(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{z}{\sigma}\right)^2} \tag{5.9.1}$$

where $\sigma = \frac{4}{N_z}$. Note that the function is normalised so that the integral over the domain is one. Then the quadrature implemented for the non linear terms is used to compute the values of the projection of this Gaussian on the polynomial subspace.

For the radial distribution of the forces, two cases can be considered. In the case of a straight blade with no flap and lead lag angle, one can interpolate the values of the force at the quadrature points in order to evaluate its participation. In the other cases, i.e. with a flap angle, or non straight blades, each blade element force needs to be associated with a radial domain to apply the force accordingly.

In the model presented here, both the lift and drag generated by each blade element is accounted for, in order to represent the interaction between the blade and the air as precisely as possible.

5.9.3 Results

Several cases are presented here, with the main objective of underlining the effect of the trade trade off between accuracy and rapidity of the description. In order to do so we observe two different flight conditions (ascending and descending axial flights), with two different approximations (a very precise one, and a fast one), for the 7A and 7AD rotors. The main difference between these two rotors is that the 7AD as tapered blade tips. It makes the induced velocity distribution easier to represent with a spectral method. Indeed, the induced velocity at the tip of the blade is less sharp, and thus easier to represent. The 7AD representation in AMB is presented on Fig 5.9.2, and its characteristics are listed in table 5.9.1

The fast configuration was computed with $N_r = 80$, $N_z = 70$ and $N_\theta = 10$. The mean time for an iteration run on one CPU was of $3.9s$. The precise configuration was run with $N_r = 200$, $N_z = 150$ and $N_\theta = 16$, and the mean time of the iteration was around 3.2 minutes. Both methods were filtered with a Lanczos input filter and an exponential filter of order $p = 20$. However the time step of the fast method was set to $1°$ of rotor rotation, while the precise case used a $0.5°$ of rotor rotation. Table 5.9.2 compares the computation time of the different methods tested for one iteration. It is to be noticed that the heart of the prescribed and free wake models are coded in Fortran, which makes them significantly faster, and that the time for each iteration tends to grow as the number of vortices to account for increases.

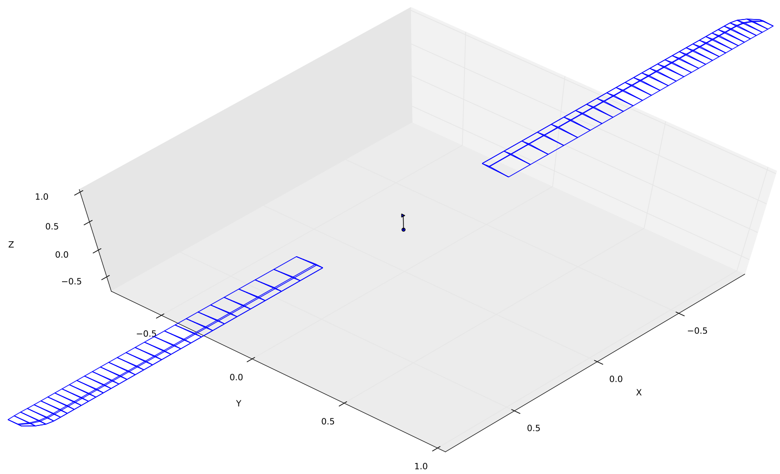


Figure 5.9.2: Representation of the 7AD rotor in AMB.

5.9.3.1 Ascending axial flight condition

For this flight condition the free stream is set to 10m/s in the axial direction. Which is a case similar to the one used for the comparison with Huang and Peters model made in chapter 4.

In the case of the 7A rotor, we compare on Fig 5.9.4, 5.9.5, 5.9.6 the axial, radial and azimuthal induced velocities on the blades respectively, for the free wake model, prescribed wake model and the new model with the fast configurations. Fig 5.9.3 presents the forces on the blade for the same case. The agreement between the models is quite good for the axial and the radial velocities. The azimuthal velocity is quite different for all cases, but the tendencies are at least coherent for all the models. The main differences can be seen at the blade tip. Indeed, in the area where the lift varies strongly, the behaviour of the induced velocity is quite difficult to represent with a spectral method.

Radius R	2.1 m
Blade chord along the span	
0.425m	0.14m
1.575m	0.14m
1.89m	0.14m
1.988m	0.14m
2.002m	0.1385m
2.018m	0.133m
2.025m	0.1298m
2.037m	0.122m
2.048m	0.113m
2.063m	0.0987m
2.078m	0.079m
2.1m	0.046m
Rotational Speed Ω	1020 rpm
Blade root radius	0.425
Airfoils used along the span	
0.425m $\mapsto$ 1.575m	OA213
1.575m $\mapsto$ 2.1m	OA209
Linear twist angle	
0.425m	$0°$
1.575m	$-4.545°$
1.89m	$-3.490°$
1.988m	$-3.877°$
2.002m	$-3.93°$
2.018m	$-3.996°$
2.025m	$-4.024°$
2.037m	$-4.0707°$
2.048m	$-4.114°$
2.063m	$-4.174°$
2.078m	$-4.233°$
2.1m	$-4.320°$

Table 5.9.1: Characteristics of the 7AD rotor.

Method	*Computation time (min)*	*Language*
Prescribed Wake	0.01	Fortran
Free wake	2.5	Fortran
New model, fast configuration	0.125	Python
New model, precise configuration	3.24	Python

Table 5.9.2: Mean time taken by the different models for one time iteration.

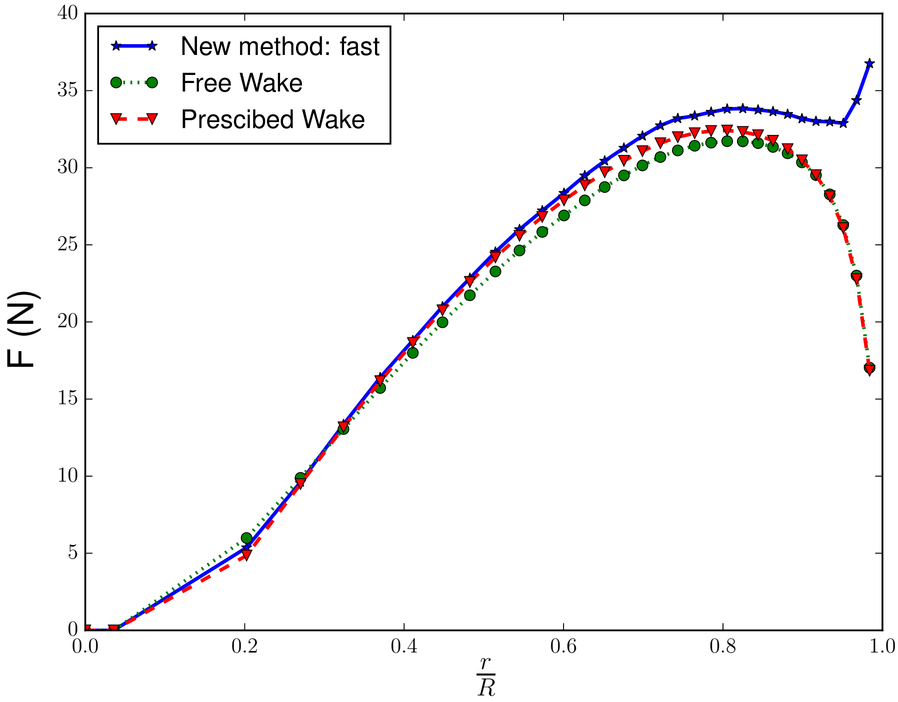


Figure 5.9.3: Comparisons on the blades of the lift distribution given by new model for the 7A rotor, with prescribed and free wake models, after 5 rotations.

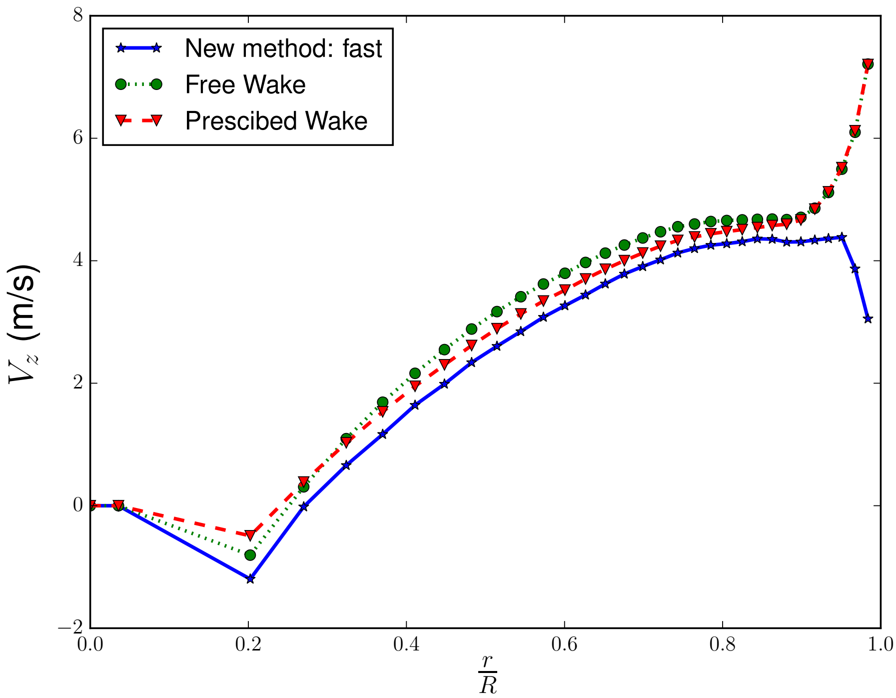


Figure 5.9.4: Comparisons on the blades of the axial velocity given by new model for the 7A rotor, with prescribed and free wake models, after 5 rotations.

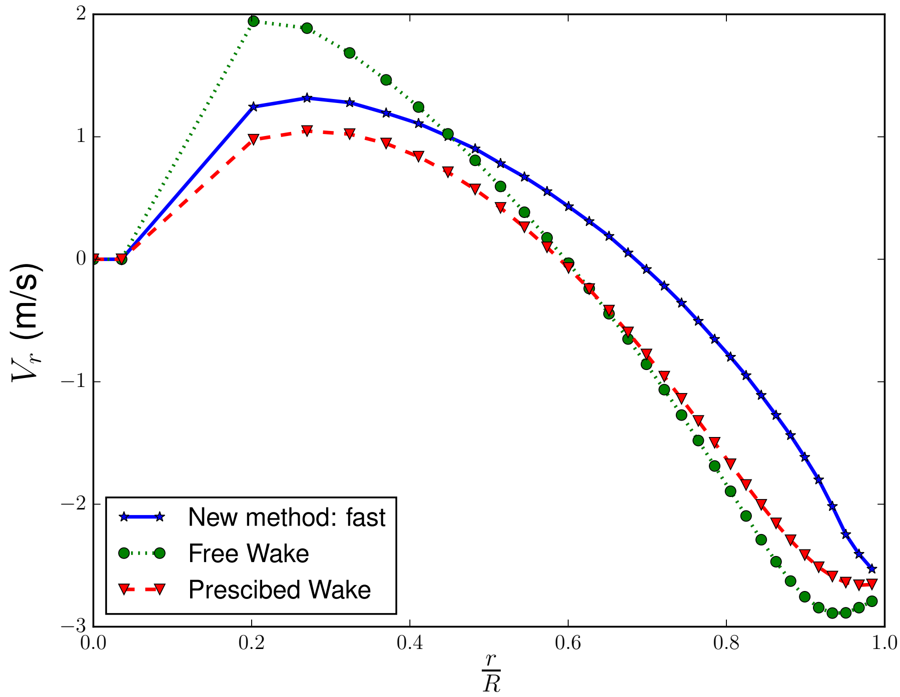


Figure 5.9.5: Comparisons on the blades of the radial velocity given by new model for the 7A rotor, with prescribed and free wake models, after 5 rotations.

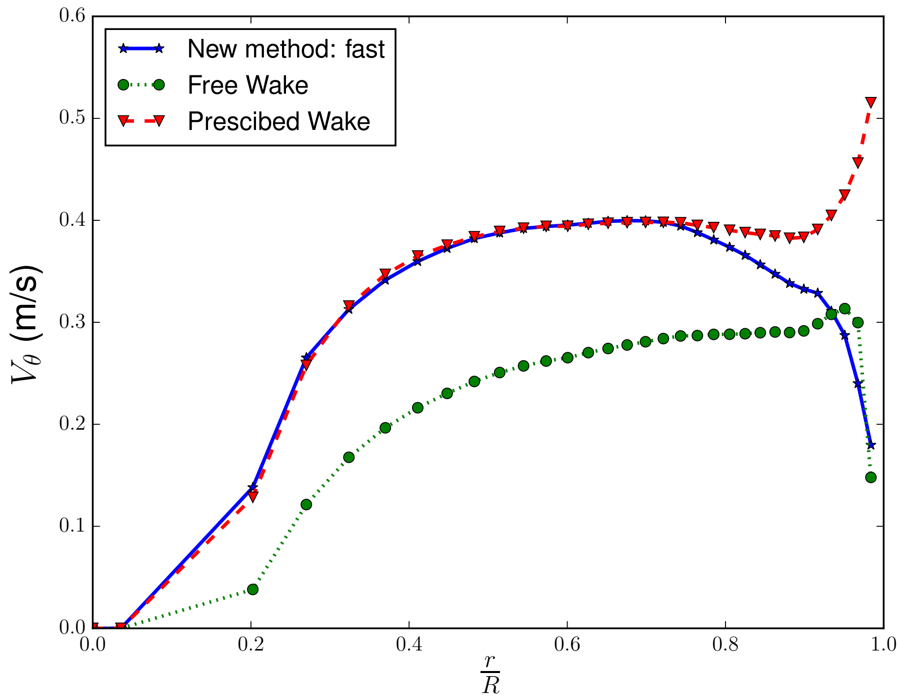


Figure 5.9.6: Comparisons on the blades of the orthoradial velocity given by new model for the 7A rotor, with prescribed and free wake models, after 5 rotations.

For the 7AD rotor, we compare on Fig 5.9.8, 5.9.9, 5.9.10 the axial, radial and azimuthal induced velocities on the blades respectively, for the free wake model, prescribed wake model and the new model with the fast configurations. Fig 5.9.7 presents the forces on the blade for the same case. The same remarks as previously can be made on the comparisons of the various models. However, in the case of the 7AD rotor, the agreement between the model at the tip of the blade is better, because of the shape of the induced velocity that is more easily represented by the spectral method.

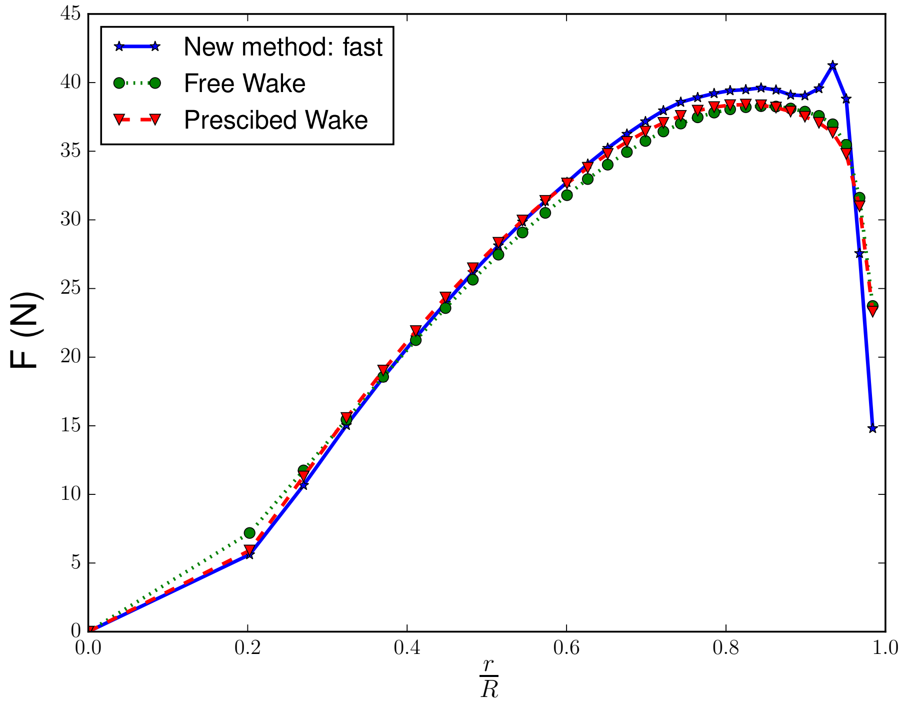


Figure 5.9.7: Comparisons on the blades of the lift distribution given by new model for the 7AD rotor, with prescribed and free wake models, after 5 rotations.

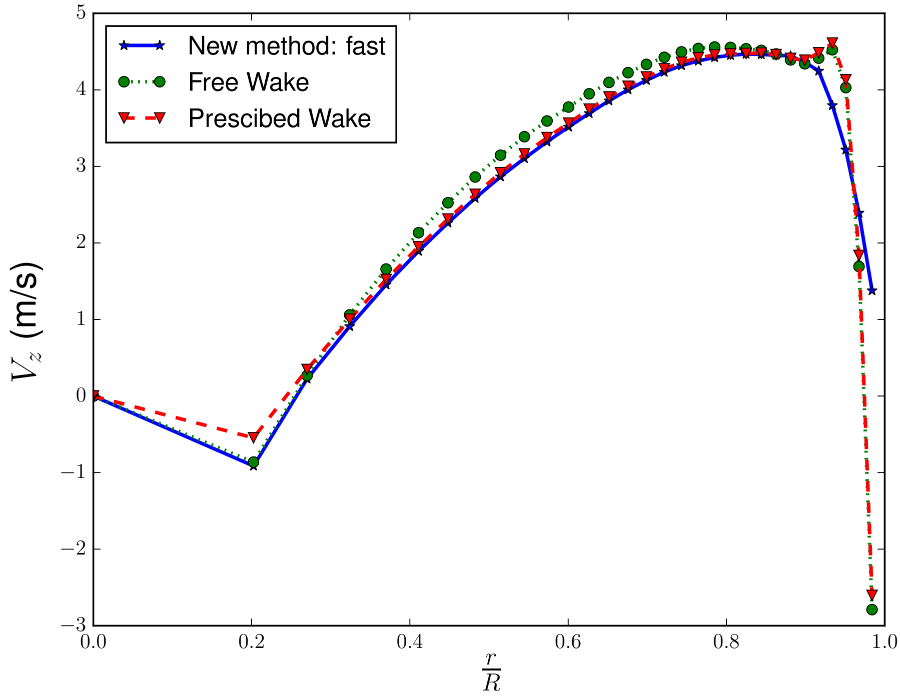


Figure 5.9.8: Comparisons on the blades of the axial velocity given by new model for the 7AD rotor, with prescribed and free wake models, after 5 rotations.

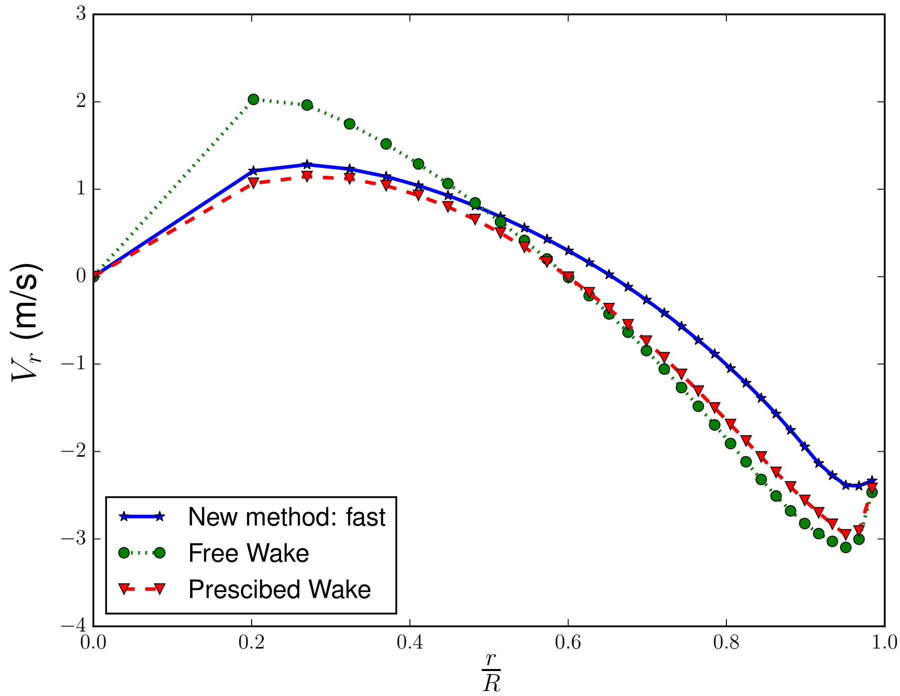


Figure 5.9.9: Comparisons on the blades of the radial velocity given by new model for the 7AD rotor, with prescribed and free wake models, after 5 rotations.

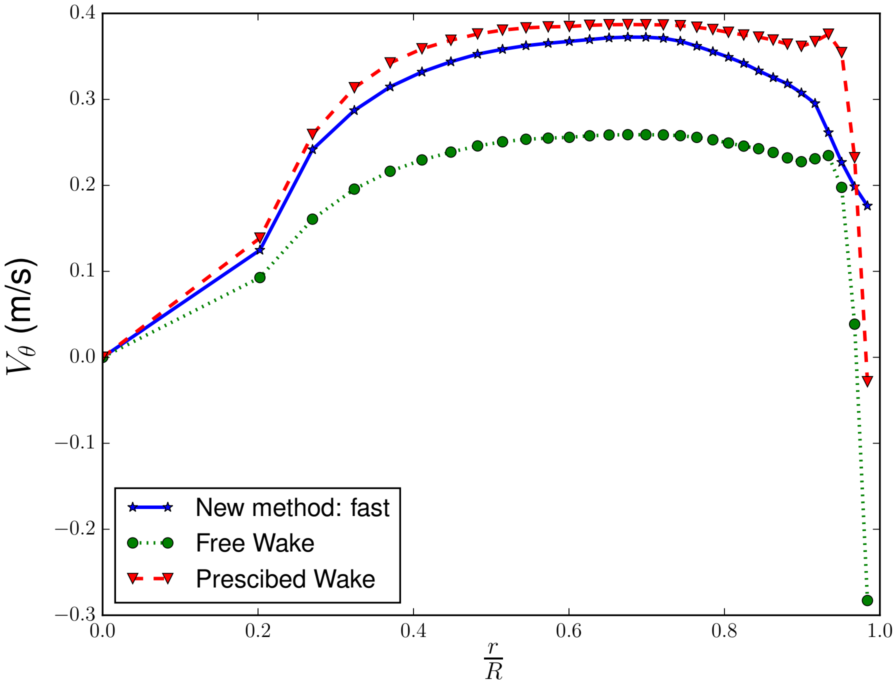


Figure 5.9.10: Comparisons on the blades of the orthoradial velocity given by new model for the 7AD rotor, with prescribed and free wake models, after 5 rotations.

Fig 5.9.11 presents the wake given by the models after 5 rotor rotations for the fast configuration. One can see that the fast model gives a good description, but that most of the effects of the blades are smoothed in the wake, as can be seen on the wake generated by the free wake model on Fig 5.9.12.

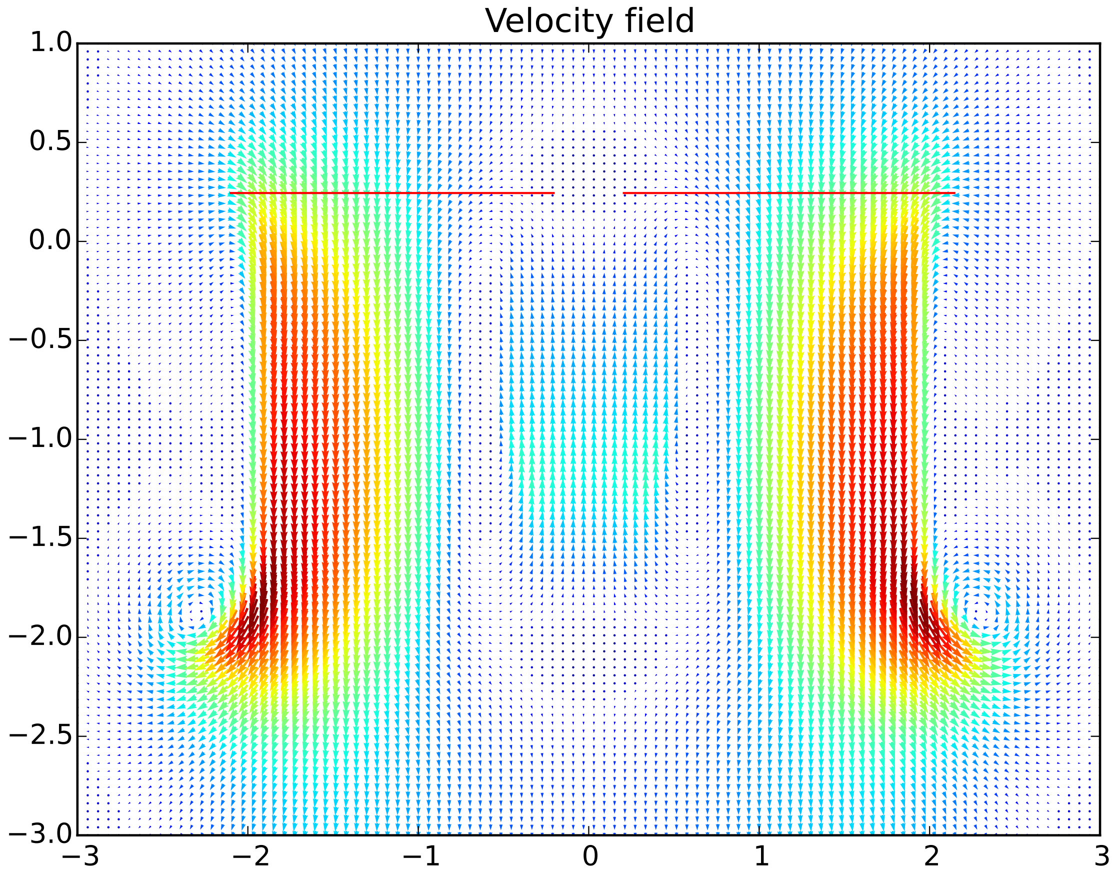


Figure 5.9.11: Wake of the 7AD rotor given by the presented model, for the fast configuration, after 5 rotations.

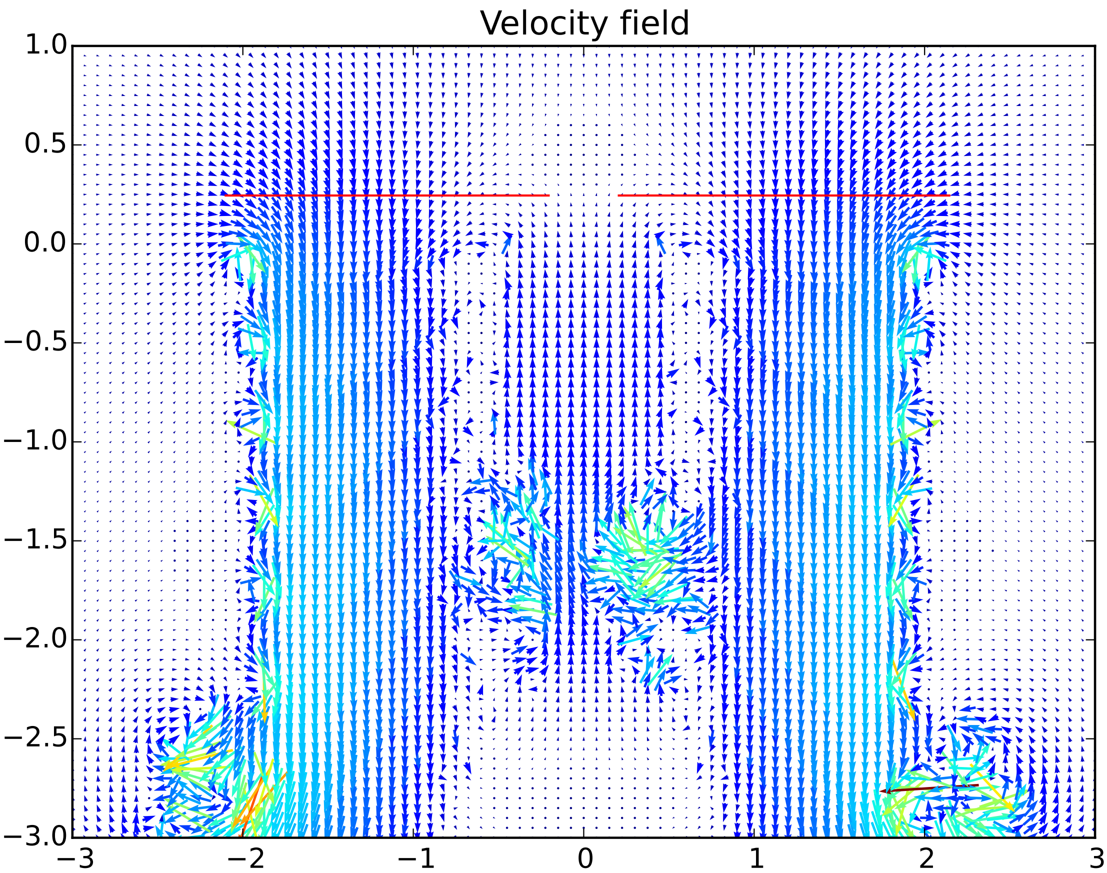


Figure 5.9.12: Wake of the 7AD rotor given by the free wake model, after 5 rotations.

5.9.3.2 Descending axial flight condition

We present here the results obtained on the 7AD rotor for a case of descending axial flight. The free stream velocity is set to $1m/s$. Only the results given by the fast configuration are presented here.

Fig 5.9.14, 5.9.15, 5.9.16 present the axial, radial and orthoradial induced velocities on the blades after 5 rotations of the rotor during the descending case, while 5.9.13 presents the lift distribution. It can be said that the free wake and the new model are in good agreement for the axial and radial velocities and the lift distribution, while the prescribed wake model seems to have reach one of its limits. Indeed, the free stream velocity being smaller than in the previous cases, the impact of the induced velocities on themselves, which is not captured by the prescribed wake model, is not negligible. However, there are some discrepancies between the representation given by the free wake model and the new model. The main one concerns the tangential velocity that is completely different for the two models. Another difference is visible between the two models mostly between 0.4 and 0.5 blade length where the two models have different tendencies. A higher order approximation would probably improve the results given by the new model by making it able to capture the phenomena at the root of these discrepancies.

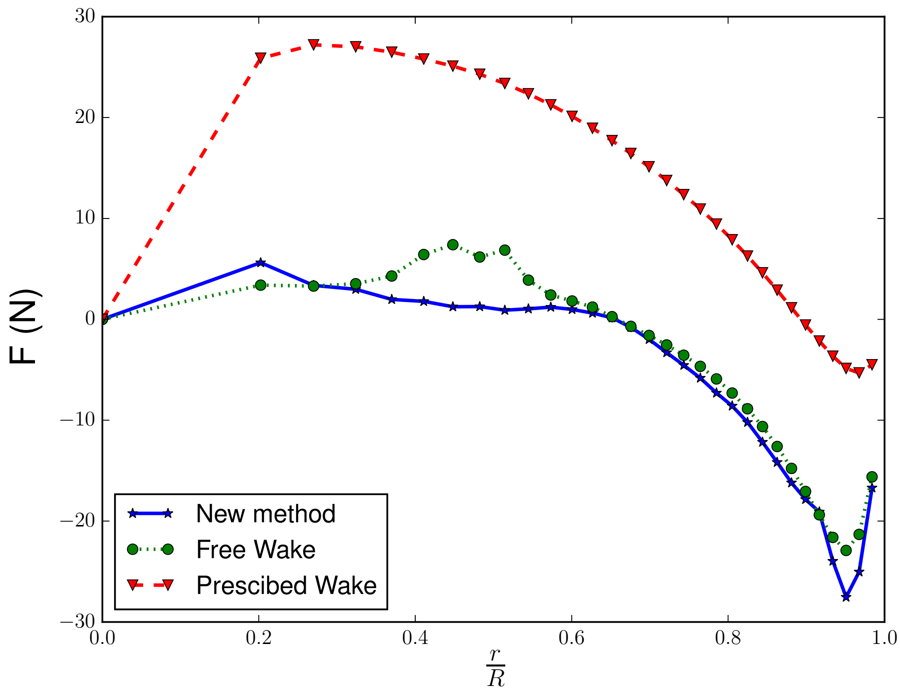


Figure 5.9.13: Comparisons on the blades of the lift distribution given by new model, with prescribed and free wake models, after 5 rotations.

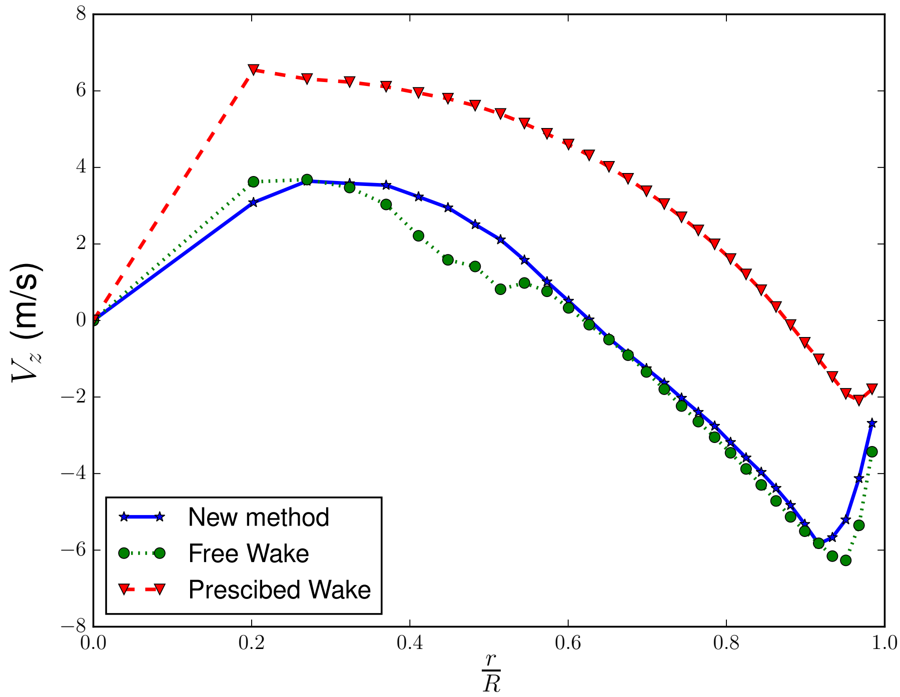


Figure 5.9.14: Comparisons on the blades of the axial velocity given by new model, with prescribed and free wake models, after 5 rotations.

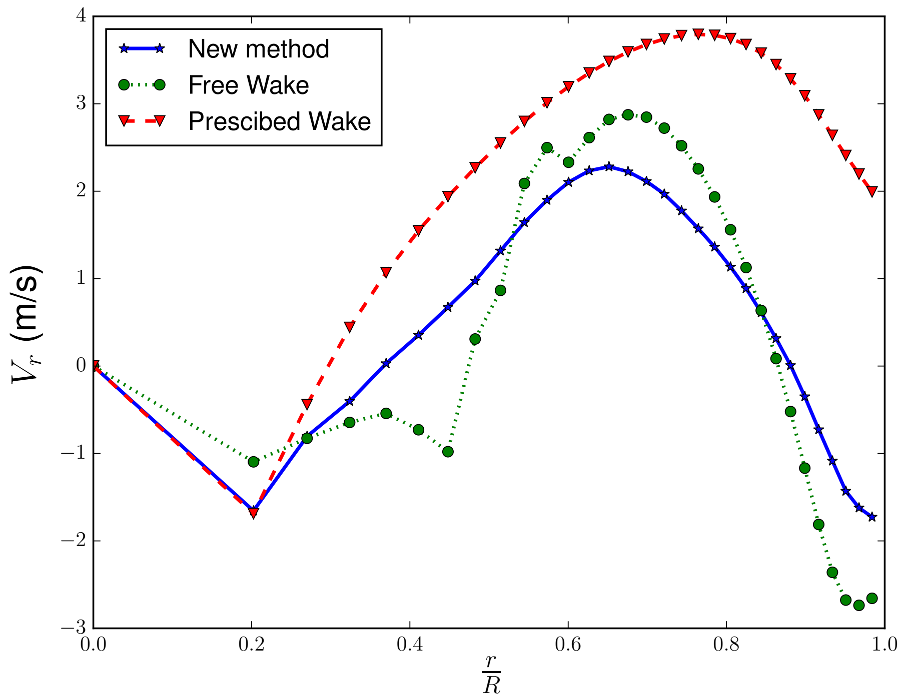


Figure 5.9.15: Comparisons on the blades of the radial velocity given by new model, with prescribed and free wake models, after 5 rotations.

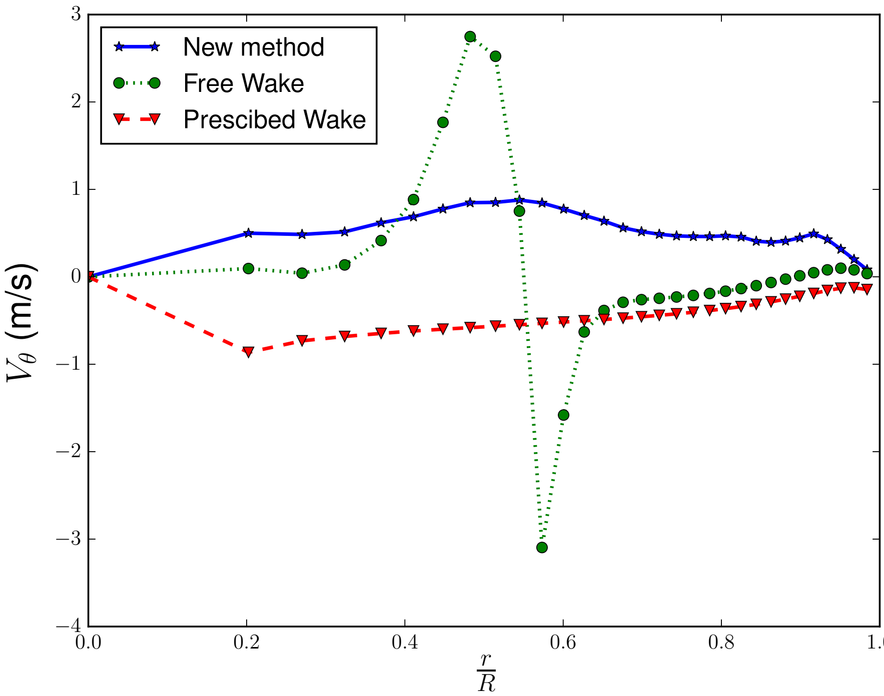


Figure 5.9.16: Comparisons on the blades of the orthoradial velocity given by new model, with prescribed and free wake models, after 5 rotations.

Fig 5.9.17 and Fig 5.9.18 present the wake after 5 rotations, as computed by the free wake and the new model respectively. One can see that the development of the wake is coherent between both models, although the fast configuration does not capture all the details of the vortices generated by the blade. Furthermore the wake is not convected as much as it is in the case of the free wake model. However, there is a good description of the inversion of the induced velocities on the blades, and of the low induced velocity value zone above the middle of the blades. Finally, one can see on Fig 5.9.18 the emergence of spurious velocities on the axis. This problem will be addressed in section 5.11, and is due to the bad treatment of the cylindrical singularity on the axis.

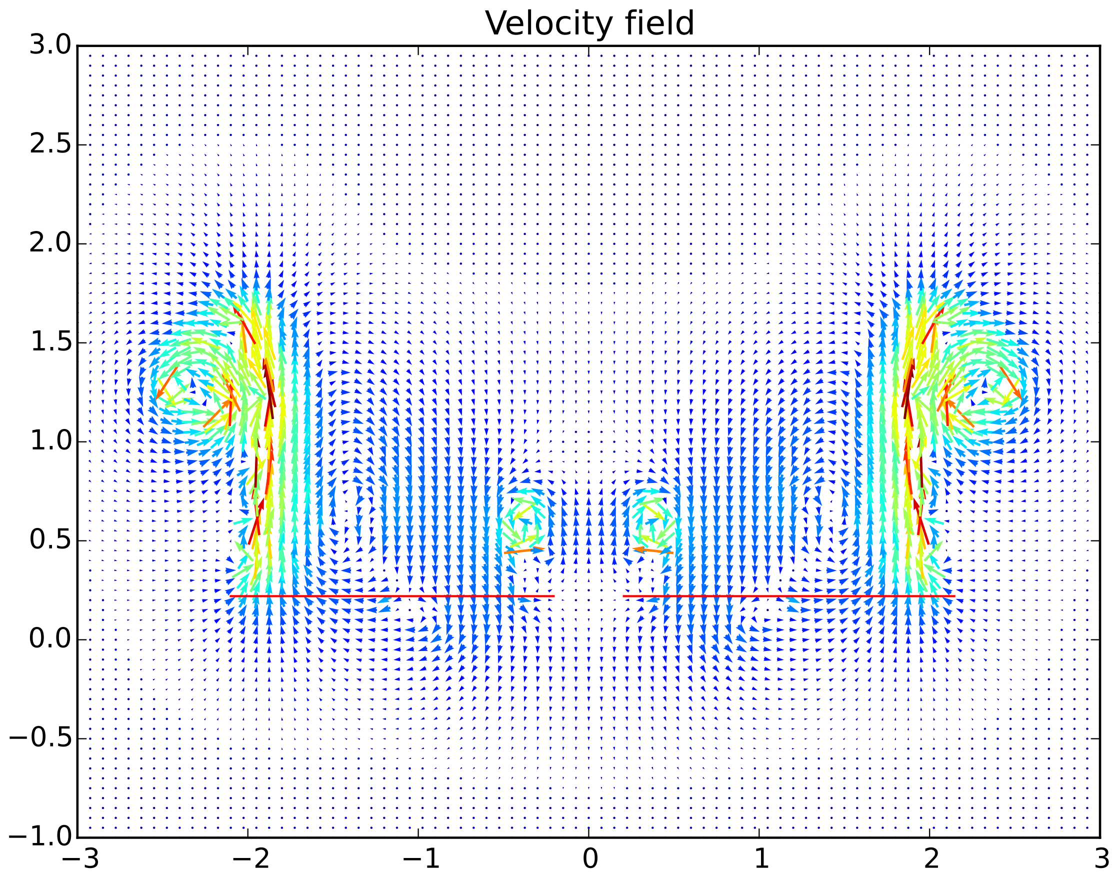


Figure 5.9.17: Wake given by the free wake model, after 5 rotations.

5.9.4 Discussion

The results given by the new model are in good agreements with the free wake model. Furthermore the main features of the induced velocities are satisfyingly represented with few coefficients, and thus a fast computation, while the more time consuming methods allow a better description of all the features of the wake. It was furthermore demonstrated that these remarks can be made for different flight conditions.

However the price to pay for precision is tremendous. The main cause of this cost is the quadrature of the non linear terms, that is quadratic with N_θ. Parallel computing and the use of more than one CPU could therefore greatly improve the performances of this algorithm, but it is difficult to say if it would be competitive with the other models. Indeed, in the computation of the new values of the velocity, the N_θ equations are uncoupled and

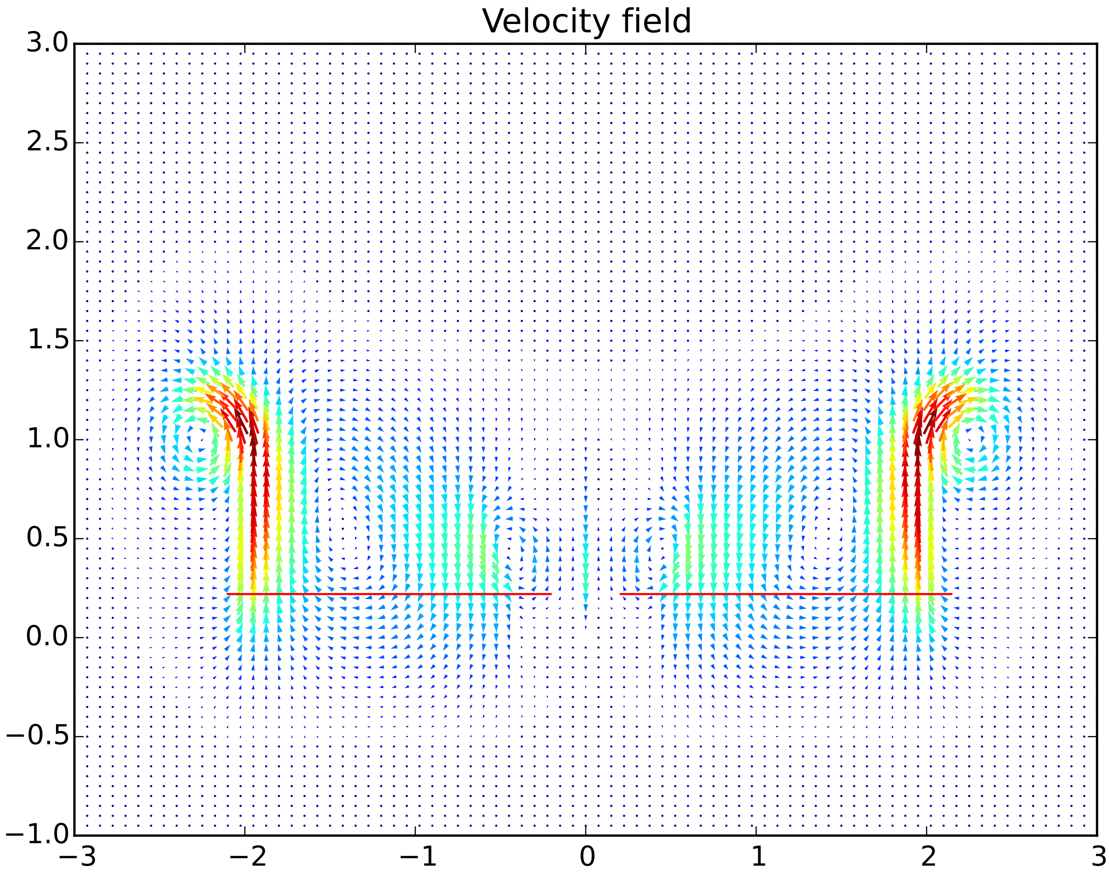


Figure 5.9.18: Wake given by the presented model, for the fast configuration.

could be solved in parallel. Furthermore, the computation of the non linear terms only depends on values that are already known and could therefore also benefit of the parallel computing.

5.10 Re-mapping

One of the reasons for the use of a filter on the blade forces inputs of the model is to avoid the emergence of Gibbs phenomena due to the sharpness of the inputs. Another way to mitigate this behaviour, which can be complementary to the use of filters, is to remap the domain in such a way that one has a better accuracy on the disc. In this section we investigate a way to apply this idea, and underline its advantages and downsides.

5.10.1 What is a remapping?

What will be called hereafter a remapping is a function allowing to pass from one vision of the space to another smoothly. In our case, the aim is to expand space at the point of sharpness, in order to reduce the norm of the derivatives of the function to be approximated. It can be seen as a way to improve convergence of the interpolation operator without increasing the number of elements considered, by acting on the other part of the norm inequality, i.e. the norm of the function.

5.10.2 How to choose a remapping?

First of all, the remapping needs to be a bijection from $[-1, 1]$ into $[-1, 1]$ in order to be able to describe the wanted space with Legendre polynomials. We propose this following form:

$$b_\beta(x) = \frac{x + \beta x^3}{1 + \beta} \tag{5.10.1}$$

where β is a parameter controlling the slope at 0 of the function b_β. For practical purposes we set $b_\beta^{-1} = q_\beta$. This choice of remapping is motivated by its simple form, a polynomial, and by the fact it is in fact quite versatile thanks to its parameter β. This parameter will allow space to be more or less expanded around 0, e.g. around the rotor disc in the z coordinate, as can be seen on Fig. 5.10.1. This will have the main effect that sharp functions will be smoothed, thus limiting the Gibbs phenomenon around the point of high gradients. On Fig. 5.10.2 we have plotted a sharp gaussian function and its approximation by the first 20 Legendre polynomials, but in various dilated spaces, using different values of β. It visibly shows the diminution of the Gibbs phenomenon.

5.10.3 How to apply a remapping?

We need to distinguish the representation space, which is the space in which we wish to compute the velocity, and the computational space, which is a dilated space where the inputs take a more acceptable form (i.e. a less sharp form). Thus, in one dimension, one can set:

$$\underbrace{u(x)}_{\text{Input space}} = \sum_{k=0}^{\infty} \hat{u}_k \phi(x) = \sum_{k=0}^{\infty} \hat{u}_k \psi\left(q_\beta(x)\right) = \sum_{k=0}^{\infty} \hat{u}_k \psi(\xi) = \underbrace{v(\xi)}_{\text{Dilated space}} \tag{5.10.2}$$

where we introduce the coordinate of the computational space $\xi = q_\beta(x)$. This remapping will dilate the neighbourhood of the point $x = 0$. Another way to see this, is that we no longer use a combination of Legendre's polynomials as projection space, but rather a combination of Legendre's polynomials composed with the function q_β.

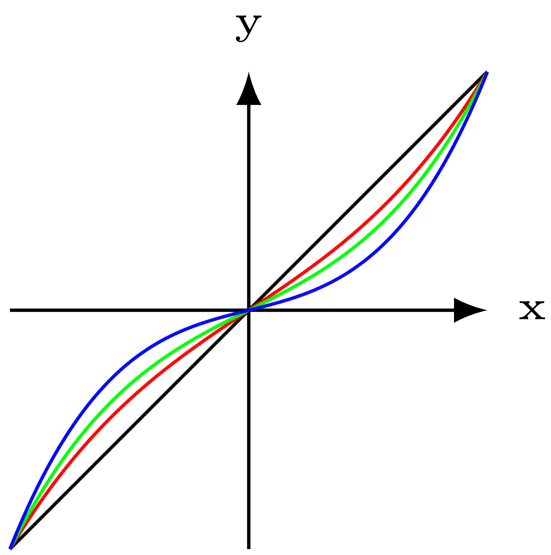


Figure 5.10.1: Evolution of the function b_β with various values of β. In black, $\beta = 0$, in red $\beta = 0.5$, in green $\beta = 1$, and in blue $\beta = 3$.

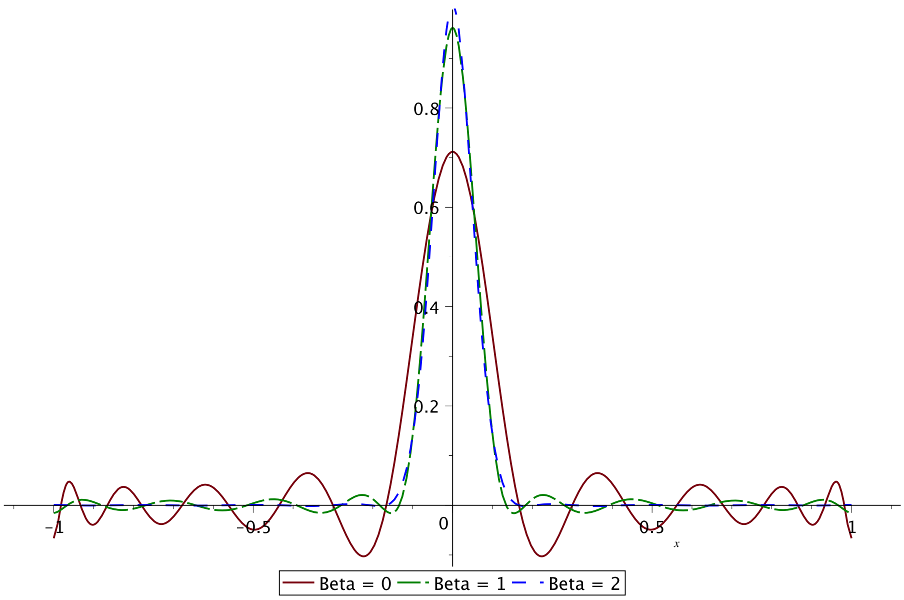


Figure 5.10.2: Approximation of a gaussian function with various values of β. In red, $\beta = 0$, in green $\beta = 1$, and in blue $\beta = 2$.

Using this, one can also show that:

$$u'(x) = \frac{\partial q_\beta}{\partial x} v'(\xi) \tag{5.10.3}$$

Here one can see, that the flatter b_β is at 0 (i.e. the steeper q_β is), the smoother v will be. However, only the neighbourhood of $x = 0$ will be improved, if a sharp behaviour appears elsewhere, there will be no remedy but a higher order, and/or filtering. One could introduce another remapping, but it requires to know the location of the sharp behaviour beforehand.

This leads to the following integration when projecting the velocity on a test function ϕ_k, with a weight function w that might depend on β:

$$\int_{-1}^{1} u'(x)\phi_k(x)w(x)dx = \int_{-1}^{1} q'_\beta\left(b_\beta(\xi)\right)\, v'(\xi)\psi_k(\xi)\, w\left(b_\beta(\xi)\right) b'_\beta(\xi)d\xi \tag{5.10.4}$$

which is obtained by the change of variable $x = b_\beta(\xi)$. Since b_β is a bijection, one have: $q'_\beta\left(b_\beta(\xi)\right) = \frac{1}{b'_\beta(\xi)}$. Furthermore, one can choose the weight function, w, as wanted. Here, one can choose $w(x) = 1$, allowing to keep a scalar product, and simplifying the presented integral to:

$$\int_{-1}^{1} u'(x)\phi_k(x)w(x)dx = \int_{-1}^{1} v'(\xi)\psi_k(\xi)d\xi \tag{5.10.5}$$

But in a case where one wish to project u and not its derivative, one would have to choose the weight $w\left(b_\beta(\xi)\right) = \frac{1}{b'_\beta(\xi)}$. In our case, we need to use both integrals, hence the incentive to use a polynomial form of the remapping in order to easily express the integrals.

In the case of the pressure equations, we choose a weight function of the form $w(\xi) = (1 + b_\beta(\xi))b'_\beta(\xi)^2$.

Thus with a good choice of weight function, and for a given β, one can compute the required matrices to run a simulation.

5.10.4 Results and comparisons

The first thing to notice for the remapping is its efficiency to help the representation of the blade forces input. We can see on Fig 5.10.3 and Fig 5.10.4 the effect of β on the representation of the input forces, localised on the rotor disc. All the figures have been made with the same number of elements in the approximations: 30 for the axial description and 30 for the radial description, on an axisymmetric case. The higher the β, the sharper the description of the inputs are. On Fig 5.10.3 one can see that the highest value of β allows to represent sharp functions as input, while having almost no Gibbs phenomenon. On Fig 5.10.4, the improvement brought by the remapping can be seen in its clear description of the rotor, and the swift transition between on disc and off disc.

It is to be noted however that too high values for β ($\beta > 8$) tend to increase the condition number of the matrices, leading to instabilities.

This method has not been applied to a rotor case, but, from the results presented here, it seems to be a possible improvement to represent cases such as the 7A rotor, which presents an especially sharp induced velocities shapes at its blade tip.

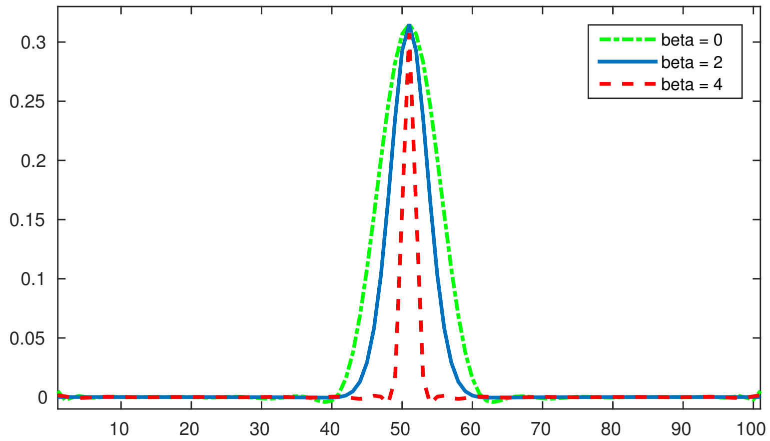


Figure 5.10.3: Approximation of a gaussian function in z with various values of β. In green, $\beta = 0$, in blue $\beta = 2$, and in red $\beta = 4$. For the case $\beta = 4$, the sharpness of the gaussian has been increased.

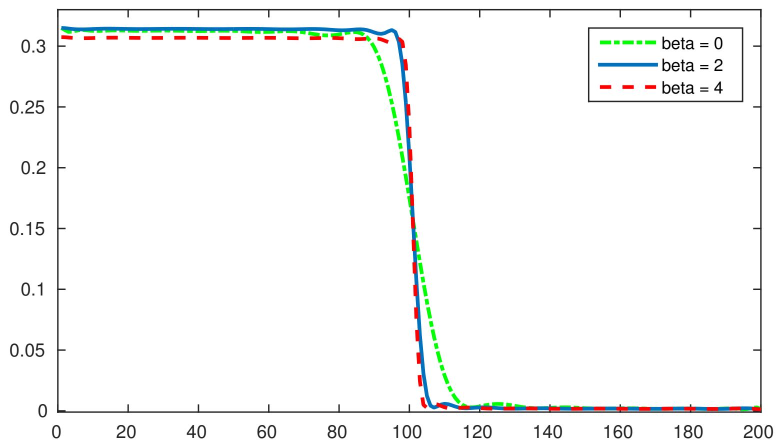


Figure 5.10.4: Approximation of a step function in r with various values of β. In green, $\beta = 0$, in blue $\beta = 2$, and in red $\beta = 4$.

5.11 Regularity on the axis

As mentioned before, the axis is singular in cylindrical coordinates. The first method employed to solve the consequences of this singularity was to impose the essential pole conditions, as described in [4], as boundary conditions on the axis. This works fine for axisymmetric cases and for most flight conditions.

However for higher and higher advance ratio conditions (or high wake angle) the algorithm experiences unstable behaviour on the axis, first for the axial velocity, which then leads to the divergence of the algorithm. The origin of this unstable behaviour could be traced back to the non linear terms of the axial momentum conservation equation mainly. They are not the only irregular terms, but are the most significant in the applications studied. The most problematic of these terms, $u^-_{m_1}\frac{\partial w_{m_2}}{r\partial\theta}$, happens to be irregular on the axis when computing it with $m_1 = 1$. Indeed, the orthoradial derivative has by definition less regularity on the axis (because of the division by r), and no further regularity can be retrieved from the $u^-_{m_1}$ component of the velocity for $m_1 = 1$, since its value is not null for this azimuthal order. For example, the case with $m_2 = 1$ should have the same regularity on the axis as w_m, where $m = m_1 + m_2 = 2$. It is in fact not the case. Although the non linear term is not singular, it imposes a non zero value on the axis, when the required pole condition should be null for this azimuthal order.

It is to be noted that this problem can be solved by the use of stronger exponential filters. However, the aim of this filter is to remove high order spurious terms, and not to smooth the representation of the velocities on the axis. Using it with a too large value of the filter order has a non negligible impact on the representation of the sharp details of the wake, and it is highly preferable to solve this problem by another mean that would not impact the accuracy of the results.

This prompted the author to either add more regularity into the w_{m_2} component, or to find a way to ignore the axis region in the non linear terms. The second solution was investigated, but lead to no satisfying solution since it implied to deal with Gibbs phenomenon, or to ignore part of the computational domain, which was just moving to another place the problem of the boundary conditions.

Furthermore, adding a constant regularity for all values of m_2 was in fact ineffective, because either too restrictive on the representation of the velocity, or leaving room for some irregularities on the axis. One answer to the problem was in fact contained in the natural pole conditions [4]. Indeed when including these boundary conditions into the non linear terms, the radial derivatives and the orthoradial derivatives were much more regular for all values of m_2. Yet, it did not solve all the irregularities.

In fact most of the problems for this method come from the fact that the required regularity of a solution is not matched by the other terms in the equations. This kind of discrepancies in a spectral method is problematic. Indeed, part of the regularity on the axis is dictated by the chosen boundary conditions (and thus in our case, by the approximation basis). Imposing an effect that is less regular on a not adapted basis creates oscillations, which are in fact assimilable to Gibbs phenomena. Indeed, the simplest example would be a non null function at $r = 0$, represented by an approximation basis, null at $r = 0$, which would try to represent it, creating oscillations, similar to Gibbs phenomena.

In the remaining of this section, we will present the way we have chosen to account for these natural pole conditions, following ideas in [70], how we have derived it, how it affects our algorithm. Unfortunately, no satisfying solution is found, although it is thought that the heart of the problem has been exposed.

5.11.1 Natural pole conditions

The natural pole conditions defined in [4] are deduced from the regularity conditions that are required for the velocity field. I.e. if one assumes the Cartesian components of the velocity to be analytic, one can deduce conditions on the regularity of the cylindrical components.

However, one can find these pole conditions by another mean than the one presented in [4]. The way they will be developed here will in fact answer the question: "what boundary conditions should be imposed on the axis to have regular non linear terms?", since it is the heart of our problem. This differs from the previous approach in the methodology, and should be slightly more restrictive on the pole conditions. To solve this question one can assume the form of the axis regularity, as proposed by [70], and tailor them in order to have sufficient regularity.

Dropping the dependency in z and time, and considering only one term, in order to clarify the derivation without loss of generality, it gives:

$$u_m^+(\overline{r},\theta) = \overline{r}^{k_m^+}\hat{u}_m^{+,n}(\overline{r})e^{im\theta} \tag{5.11.1}$$
$$u_m^-(\overline{r},\theta) = \overline{r}^{k_m^-}\hat{u}_m^{-,n}(\overline{r})e^{im\theta} \tag{5.11.2}$$
$$w_m(\overline{r},\theta) = \overline{r}^{k_m^z}\hat{w}_m^{n}(\overline{r})e^{im\theta} \tag{5.11.3}$$
$$p_m(\overline{r},\theta) = \overline{r}^{k_m^p}\hat{p}_m^{n}(\overline{r})e^{im\theta} \tag{5.11.4}$$

Where $k_m^+, k_m^-, k_m^z, k_m^p$ control the regularities of u_m^+, u_m^-, w_m, p_m on the axis, and are to be determined. To do so, we input this form of the velocity components in the non linear terms and verify that the non linear terms are at least as regular as its corresponding term in the equations. Meaning that the non linear terms of azimuthal order m added to the axial part of the velocity should be at least as regular as w_m.

Computing the non linear term associated to the u_m^- component, and only considering the $u^-\partial u^-$ terms, where m_1 and m_2 are positive, we have:

$$\frac{u_{m_1}^-}{2}\frac{\partial u_{m_2}^-}{\partial r} - \frac{u_{m_1}^-}{2}\frac{\partial u_{m_2}^-}{r\partial\theta} = \frac{1}{2}u_{m_1}^{-,n}r^{k_{m_1}^-}\left[u_{m_2}^{-,n}r^{k_{m_2}^- -1}(k_{m_2}^- - m_2 + 1) + r^{k_{m_2}^-}\frac{\partial u_{m_2}^{-,n}}{\partial r}\right] \tag{5.11.5}$$

with $m_1 + m_2 = m$ being imposed by the exponential term. This gives the following possible conditions on k_m^-:

$$k_{m_1}^- + k_{m_2}^- \geqslant k_m^-, \quad \text{if} \quad k_{m_2}^- - m_2 + 1 = 0 \tag{5.11.6}$$
$$k_{m_1}^- + k_{m_2}^- + 1 \geqslant k_m^-, \quad \text{if} \quad k_{m_2}^- - m_2 + 1 = 0 \quad \text{and} \quad \left.\frac{\partial u_{m_2}^{-,n}}{\partial r}\right|_{r=0} = 0 \tag{5.11.7}$$
$$k_{m_1}^- + k_{m_2}^- + 1 \geqslant k_m^-, \quad \text{if} \quad k_{m_2}^- - m_2 + 1 = -1 \tag{5.11.8}$$
$$k_{m_1}^- + k_{m_2}^- - 1 \geqslant k_m^-, \quad \text{otherwise} \tag{5.11.9}$$

With similar derivations on the other non linear terms, a set of conditions can be derived for all parameters. Furthermore, by applying this method to the whole equation, and not only the non linear terms, one can derive the same relations for the pressure. Finally, one can settle on the following rules for the regularities of the velocity in order to solve all the conditions:

$$
\begin{aligned}
k_m^+ &= m+1 \\
k_m^- &= m-1 \\
k_m^z &= m \\
k_m^p &= m \\
\left.\frac{\partial \hat{u}_m^{+,n}}{\partial r}\right|_{r=0} &= 0 \\
\left.\frac{\partial \hat{u}_m^{-,n}}{\partial r}\right|_{r=0} &= 0 \\
\left.\frac{\partial \hat{w}_m^{n}}{\partial r}\right|_{r=0} &= 0 \\
\left.\frac{\partial \hat{p}_m^{n}}{\partial r}\right|_{r=0} &= 0
\end{aligned}
\tag{5.11.10}
$$

One can verify that the boundary conditions on the degree of m imposed by this method is similar to the one obtained in [4]. However, the conditions to impose on the velocity in order to produce regular non linear terms are more constraining than the natural and essential pole conditions. This justify the fact that the natural pole conditions were not sufficient in our case. It is however to be noticed that most of the cases treated in the literature do not impose these conditions, although there seems to be no true consensus on the matter. The need for stronger boundary conditions probably emerged from the lack of viscosity in our equations, which thus tends to amplify the effects of the numerical errors and the high order non linear terms, which are the root of this problem.

5.11.2 Impact on the algorithm

One way to implement the natural pole conditions automatically in our description is to pass by a pseudo basis which adds the relevant regularity to the given coefficient, in the form of a r^m factor. This method presented in [70] allows to respect the pole conditions without impacting too much the algorithm. Indeed, most of the r^m terms will be simplified. However, since the polynomials will be modified, all the matrices will change.

We therefore define the velocity component and the pressure as follows, $\forall m \geqslant 2$:

$$u_{k,m}^{+}(\overline{r},\theta,\overline{z}) = \sum_{n=0}^{N_r}\sum_{j=0}^{N_z} \overline{r}^{m+1}\hat{u}_{k,m}^{+,n,j}\phi_n^{m+1}(\overline{r})\gamma_j(\overline{z})e^{im\theta} \tag{5.11.11}$$

$$u_{k,m}^{-}(\overline{r},\theta,\overline{z}) = \sum_{n=0}^{N_r}\sum_{j=0}^{N_z} \overline{r}^{m-1}\hat{u}_{k,m}^{-,n,j}\phi_n^{m-1}(\overline{r})\gamma_j(\overline{z})e^{im\theta} \tag{5.11.12}$$

$$w_{k,m}(\overline{r},\theta,\overline{z}) = \sum_{n=0}^{N_r}\sum_{j=0}^{N_z} \overline{r}^{m}\hat{w}_{k,m}^{n,j}\phi_n^{m}(\overline{r})\gamma_j(\overline{z})e^{im\theta} \tag{5.11.13}$$

$$p_{k,m}(\overline{r},\theta,\overline{z}) = \sum_{n=0}^{N_r}\sum_{j=0}^{N_z} \overline{r}^{m}\hat{p}_{k,m}^{n,j}\phi_n^{m}(\overline{r})\gamma_j(\overline{z})e^{im\theta} \tag{5.11.14}$$

The first step of the projection method therefore becomes:

$$
\begin{aligned}
\tilde{u}^{+}_{k+1,m} &= \frac{4}{3}u^{+}_{k,m} - \frac{1}{3}u^{+}_{k-1,m} + \frac{2\delta t}{3}\left(-\frac{\partial p_{k,m}}{r\partial r} + r^{-m-1}(2\mathrm{NL}^{u^+}_{k,m} - \mathrm{NL}^{u^+}_{k-1,m})\right)\\
\tilde{u}^{-}_{k+1,m} &= \frac{4}{3}u^{-}_{k,m} - \frac{1}{3}u^{-}_{k-1,m} + \frac{2\delta t}{3}\left(-2mp_{k,m} - r\frac{\partial p_{k,m}}{\partial r} + r^{-m+1}(2\mathrm{NL}^{u^-}_{k,m} - \mathrm{NL}^{u^-}_{k-1,m})\right)\\
\tilde{w}_{k+1,m} &= \frac{4}{3}w_{k,m} - \frac{1}{3}w_{k-1,m} + \frac{2\delta t}{3}\left(-\frac{\partial p_{k,m}}{\partial z} + r^{-m}(2\mathrm{NL}^{w}_{k,m} - \mathrm{NL}^{w}_{k-1,m})\right)
\end{aligned}
\tag{5.11.15}
$$

The pressure correction is then given by:

$$
-\frac{3}{2\delta_t}\mathrm{div}\left(\overrightarrow{\nu}_{k+1,m}\right) = -\Delta(p_{k+1,m} - p_{k,m}) \tag{5.11.16}
$$

with:

$$
\overrightarrow{\mathrm{div}}\left(\overrightarrow{\nu}_{k+1,m}\right) = r^m\left((m+1)\tilde{u}^{+}_{k,m} + \frac{r}{2}\frac{\partial \tilde{u}^{+}_{k,m}}{\partial r} + \frac{1}{2}\frac{\partial \tilde{u}^{-}_{k,m}}{r\partial r} + \frac{\partial \tilde{w}_{k,m}}{\partial z}\right) \tag{5.11.17}
$$

and:

$$
\Delta p_{k,m} = r^m\left(\frac{2m+1}{r}\frac{\partial p_{k,m}}{\partial r} + \frac{\partial^2 p_{k,m}}{\partial r^2} + \frac{\partial^2 p_{k,m}}{\partial z^2}\right) \tag{5.11.18}
$$

where the azimuthal derivatives have been simplified with the corresponding terms from the radial derivatives.

Furthermore the non linear terms can be expressed as:

$$
\begin{aligned}
\mathrm{NL}_{+} &= r^{m+1}\left(\frac{ru^{+}_{m_1}}{2}\frac{\partial u^{+}_{m_2}}{\partial r} + \frac{u^{-}_{m_1}}{2r}\frac{\partial u^{+}_{m_2}}{\partial r} + (m_2+1)u^{+}_{m_1}u^{+}_{m_2} + w_{m_1}\frac{\partial u^{+}_{m_2}}{\partial z}\right)\\
\mathrm{NL}_{-} &= r^{m-1}\left(\frac{r^2u^{+}_{m_1}}{2}\frac{\partial u^{-}_{m_2}}{\partial r} + \frac{u^{-}_{m_1}}{2r}\frac{\partial u^{-}_{m_2}}{\partial r} + (m_2-1)u^{+}_{m_1}u^{-}_{m_2} + w_{m_1}\frac{\partial u^{-}_{m_2}}{\partial z}\right)\\
\mathrm{NL}_{z} &= r^{m}\left(\frac{ru^{+}_{m_1}}{2}\frac{\partial w_{m_2}}{\partial r} + \frac{u^{-}_{m_1}}{2r}\frac{\partial w_{m_2}}{\partial r} + m_2u^{+}_{m_1}w_{m_2} + w_{m_1}\frac{\partial w_{m_2}}{\partial z}\right)
\end{aligned}
\tag{5.11.19}
$$

Thanks to the azimuthal part of the description, we have $m = m_1 + m_2$, which allows to compensate for the terms simplified in the momentum conservation equation.

This shows that by simplifying by the relevant power of r, one can use similar equations while ensuring both the essential and natural pole conditions.

5.11.3 Choice of polynomials

Because of the r^m factor, the boundary conditions that the polynomials have to verify are no longer expressed in the same manner. This implies to redefine all the polynomials, with a new dependency on the azimuthal order m. We can therefore define our new polynomial family as follows:

$$
\begin{aligned}
\chi^m_n(x) = &\frac{2n^2+m+10n+12}{m(2n+3)}L_n(x) + \frac{1}{2n+5}L_{n+1}(x)\\
&- \frac{2n^2+2n+m}{m(2n+3)}L_{n+2}(x) - \frac{1}{2n+5}L_{n+3}(x)
\end{aligned}
\tag{5.11.20}
$$

These polynomials verify the following boundary conditions:

$m = 0$	$u^+_{k,0}$	$u^-_{k,0}$	$w_{k,0}$	$p_{k,0}$
Boundary condition at $x = -1$	$u^+_{k,0}(x) = 0$	$u^-_{k,0}(x) = 0$	$\frac{\partial w_{k,0}}{\partial x}(x) = 0$	$\frac{\partial p_{k,0}}{\partial x}(x) = 0$
Boundary condition at $x = +1$	$\frac{\partial u^+_{k,0}}{\partial x}(x) = 0$	$\frac{\partial u^-_{k,0}}{\partial x}(x) = 0$	$\frac{\partial w_{k,0}}{\partial x}(x) = 0$	$\frac{\partial p_{k,0}}{\partial x}(x) = 0$
Polynomial	ϕ_n	ϕ_n	γ_n	γ_n

Table 5.11.1: Boundary conditions and polynomials used for $m = 0$

$m = 1$	$u^+_{k,1}$	$u^-_{k,1}$	$w_{k,1}$	$p_{k,1}$
Boundary condition at $x = -1$	$u^+_{k,1}(x) = 0$	$\frac{\partial u^-_{k,1}}{\partial x}(x) = 0$	$w_{k,1}(x) = 0$	$p_{k,1}(x) = 0$
Boundary condition at $x = +1$	$\frac{\partial u^+_{k,1}}{\partial x}(x) = 0$	$\frac{\partial u^-_{k,1}}{\partial x}(x) = 0$	$\frac{\partial w_{k,1}}{\partial x}(x) = 0$	$\frac{\partial p_{k,1}}{\partial x}(x) = 0$
Polynomial	ϕ_n	γ_n	ϕ_n	ϕ_n

Table 5.11.2: Boundary conditions and polynomials used for $m = 1$

$$
\begin{aligned}
\frac{\partial}{\partial x}\left(\chi_n^m(x)\right)\Big|_{x=-1} &= 0 \\
\frac{\partial}{\partial x}\left(r^m \chi_n^m(x)\right)\Big|_{x=1} &= 0
\end{aligned}
\tag{5.11.21}
$$

The tables 5.11.1, 5.11.2, 5.11.3 summarise the choice of boundary conditions and of polynomials for the three components of the velocity as well as the pressure, depending on the azimuthal order.

5.11.4 Limitations

It appears, once the solution is implemented and tested, that it does not solve the original problem. This could have been foreseen by the fact that we impose a new boundary

$m \geqslant 2$	$u^+_{k,m}$	$u^-_{k,m}$	$w_{k,m}$	$p_{k,m}$
Boundary condition at $x = -1$	$\frac{\partial^m u^+_{k,m}}{\partial x^m}(x) = 0$	$\frac{\partial^m u^-_{k,m}}{\partial x^m}(x) = 0$	$\frac{\partial^m w_{k,m}}{\partial x^m}(x) = 0$	$\frac{\partial^m p_{k,m}}{\partial x^m}(x) = 0$
Boundary condition at $x = +1$	$\frac{\partial u^+_{k,m}}{\partial x}(x) = 0$	$\frac{\partial u^-_{k,m}}{\partial x}(x) = 0$	$\frac{\partial w_{k,m}}{\partial x}(x) = 0$	$\frac{\partial p_{k,m}}{\partial x}(x) = 0$
Polynomial	$\left(\frac{x+1}{2}\right)^{m+1} \chi_n^{m+1}$	$\left(\frac{x+1}{2}\right)^{m-1} \chi_n^{m-1}$	$\left(\frac{x+1}{2}\right)^{m} \chi_n^{m}$	$\left(\frac{x+1}{2}\right)^{m} \chi_n^{m}$

Table 5.11.3: Boundary conditions and polynomials used for $m > 1$

condition on the derivative of the polynomials that is not ensured by the terms imposed in the equations. In fact, the problem has simply been moved elsewhere. The author could not find another valid way to ensure the regularity on the axis of the non linear and pressure terms of the equations. The problem is seldom considered in the literature, but Blackburn and Sherwin address it in [71]. However their solution does not seem satisfying in our case. They advice to neglect the terms that are not regular enough on the axis. In our case, this would notably hinder the representation of the high advance ratio cases, since the representation of the advection term is problematic.

The origin of the problem is in fact the way the computational domain is seen, and the fact that it is ill design for extreme flight conditions such as perfectly edgewise flow.

6
Conclusion and further work

Résumé en français: Conclusion et perspectives Le travail de cette thèse a permis de développer un nouveau modèle de vitesses induites dépassant un certain nombre de défauts des méthodes à nombre d'états finis qui ont l'avantage d'être adaptables à chaque type d'application en termes de compromis précision vs temps de calcul grâce au choix du nombre d'états. Cependant, ce modèle n'est lui même pas sans défauts. Il semble tout de même que les résultats obtenus grâce à ce modèle démontrent sa pertinence pour la prise en compte de certains phénomènes. Il reste cependant plus coûteux en temps de calcul que le modèle de Peters et Huang.

En suivant la voie explorée dans cette thèse, il peut être pertinent de développer d'autres modèles, en utilisant par exemple des coordonnées ellipsoïdales, en améliorant les conditions de régularité sur l'axe, ou simplement en rendant plus efficace le modèle, en particulier par une implémentation réduisant le temps de calcul..

6.1 Summary

After the analysis of the state of the art in the matter of rotor finite states dynamic inflow modelling, we have concluded for the need of a new model, more general, and able to cover a wider range of flight conditions. The study of the Huang and Peters model allowed to reveal some limitations of the model, first thanks to a theoretical study, and then by implementing it both in the frequency and time domain. The need for a new model was supported by several observations, mainly pointing out as main culprits the assumptions taken by the previous models, that were responsible for the diminution of the domain of validity. Thus, the main developments made in this thesis were achieved by removing the too limiting assumptions, while still applying a Galerkin method to the equations.

We have therefore developed a model with a larger domain of validity. In order to do so, we have used a Galerkin method on the incompressible Euler equations, with no further assumptions, thus removing the linearisation and the velocity potential assumptions. This allows to treat all the domain in a coherent manner without distinguishing between what is above, on and below the rotor. The time discretisation scheme allows to deal with non linear equations, and to respect the continuity equation at each time step. Furthermore, it allows to easily couple the induced velocity model with a blade element model. This raises the question of the integration of a rotor within the method. Here again, the point of view previously used in the Huang and Peters model (and in fact by all the Peters' finite states dynamic rotor inflow models), of a pressure discontinuity, is changed. Indeed this simplified modelling of the rotor as a disc of pressure discontinuity, only accounts for the lift distribution. The inputs of the new model are all the forces generated by the blades. This allows to account for all the contributions of the blades in a more comprehensive way, and not only the lift forces.

This new model is thought to be more coherent in its derivation, and allows to reach a greater domain of validity than the previous models. Most notably, it is able to represent most of the flight conditions encountered by a helicopter, and to represent the dynamic evolution of the wake during the transition between these flight conditions.

This has however been achieved at the cost of some drawbacks, the main one being a lack of performance in terms of computational time in the present implementation. Nevertheless, the results obtained are thought to be a good display of the capacities of the new model, which may lead the way for further developments, with better efficiency and stability. Fig 6.1.1 presents the evolution with time of the various finite states models, and where the presented model hopes to be and shows the step that has been reached in the hypothesis considered with respect to the other models.

On the practical side, the model has been implemented in Matlab, and then has been coupled with a blade element model in Python. The tests of various flight conditions, revealed its capacities to treat all kinds of flight conditions, and to present a good agreement with other models. The axial flight condition has been tested in descending and ascending conditions, and revealed the expected results. The contraction of the wake is similar to the

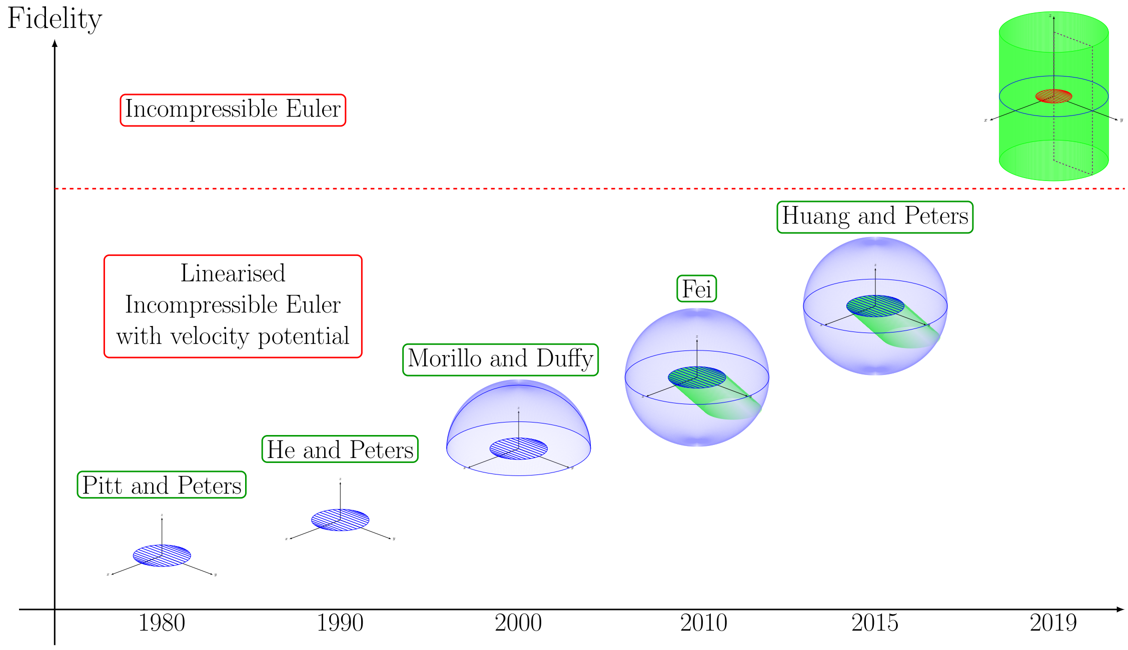


Figure 6.1.1: Evolution of finite states induced velocity models, from Pitt and Peters to the present model

one predicted by a free wake model, and the development of the wake with time is satisfying. Furthermore, the calculated induced velocities on the blades are in good agreement with the one computed with prescribed wake and free wake models.

6.2 Conclusion

The following conclusion can be taken from the work made is this thesis:

1. Existing finite states dynamic models have been studied in depth in chapter 3, giving a better understanding of their strengths and limitations. Most notably, it revealed some issues for the adaptation of the model to the time domain. Furthermore, this study highlighted the origins of the limitations and thus indicated the way to improve the representation of the induced velocities.

2. A new finite states dynamic model of rotor induced velocities has been developed in chapter 5, that expands the domain of validity of the Huang and Peters model. All components of the induced velocity field can be computed, at all points around the rotor and in all flight conditions in a homogeneous way with a unique consistent model. Since the rotor is seen as acting directly on the air with its blade forces, the model gives a more comprehensive way to account for its effects.

3. This new model retains the finite states properties of its predecessors by using a Galerkin method. However, it treats the incompressible Euler equations with no other assumptions, which allows to have a more homogeneous and coherent model for all cases encountered (there is no need for a composition and transition between different models depending where the induced velocity is calculated).

4. This model has been successfully implemented and coupled to the blade element model of a rotorcraft simulation tool in order to demonstrate its capacities, and to apply it on a real rotor, and finally to compare it to other induced velocity models for validation.

5. These comparisons show that the new proposed model is able to reach the accuracy of a free wake model for all the induced velocity components. Yet for the moment, this accuracy is obtained at a computational cost higher than the free wake model. But the free wake model is coded in Fortran whereas the new model is in Python and could be widely parallelised.

6.3 Further Work

A number of aspects remains to be improved for the current state of the model.

1. The choice of representing the domain by a cylinder with a given ratio Λ makes it difficult to compromise between the two extreme flight conditions of a helicopter, which are the hover condition and the high advance ratio flight case. It could therefore be interesting to use the ellipsoidal coordinates that are used by Peters et al. which provides a better framework for representing the rotor wake in all conditions. They were however not considered in the current work because of their singularities, and of the difficulty to derive meaningful pole conditions in this system (which would contain the exact same singularity on the axis, on top of other ones on the disk). However, with sufficient study of the regularity conditions, they seem like the ideal way to model the induced flow by a rotor.

2. Another point of improvement lies in representation of the velocities for a low number of elements in the spectral approximation. The algorithm should be sufficiently

efficient to be used with a large enough number of elements. For some applications (e.g. rotorcraft presizing or real time simulation), where short execution time is of the essence, reducing the number of elements in the representation may be a good way to match the required efficiency. However the representation of the wake might be far less accurate, as well as the behaviour on the blades. A better implementation would allow parallel computing, that would remedy to this drawback.

3. Finally the stability of the algorithm is jeopardised by the axis pole conditions. First, the axis regularity conditions used do not seem to be enough, and no meaningful conditions could be derived in order to maintain the accuracy on the axis. This makes some flight conditions require more filtering than others, because their representation on the axis tends to generate spurious high order terms. A study of the behaviour of the non linear terms in the cylindrical coordinate system seems necessary in order to truly solve this problem.

$\mathcal{A}$ Miscellaneous

A.1 Legendre Polynomials

The Legendre functions of the first and second kind are solutions to the differential equations:

$$\frac{\partial}{\partial x}\left[(1-x^2)\frac{\partial P_n^m}{\partial x}(x)\right] + \left[n(n+1) - \frac{m^2}{1-x^2}\right] P_n^m(x) = 0 \tag{A.1.1}$$

In the case where $m = 0$, one finds back the Legendre's differential equation, which solutions are the orthogonal Legendre polynomials.

For integer values of n and m, one can express them as:

$$P_n^m(x) = (-1)^m (1-x^2)^{\frac{m}{2}} \frac{\partial^m}{\partial x^m} P_n(x) \tag{A.1.2}$$

Where:

$$P_n(x) = \frac{1}{2^n n!} \frac{\partial^n}{\partial x^n} (x^2 - 1)^n \tag{A.1.3}$$

This gives:

$$P_n^m(x) = \frac{(-1)^m}{2^n n!} (1-x^2)^{\frac{m}{2}} \frac{\partial^{n+m}}{\partial x^{n+m}} (x^2-1)^n \tag{A.1.4}$$

They are furthermore linked by numerous relations that can be found in [67] and [72] for the Legendre polynomials and the associated Legendre functions respectively.

A.1.1 Form of the x derivative of the velocity potential

The motivation for the change of variable made by Huang is the stability of the solution when high harmonic numbers are considered.

However it appeared, while manipulating the various expressions, that the x derivative of the Φ_n^m potential functions could in fact be expressed solely with $\frac{P_n^m}{\nu}$ functions on the disk:

$$
\begin{aligned}
\left.\frac{\partial \Phi_n^m}{\partial x}\right|_{\eta=0} &= \frac{\sqrt{1-\nu^2}}{\nu}\frac{\partial \overline{P}_n^m}{\partial \nu}\cos(m\psi)\cos(\psi) - \frac{m}{\sqrt{1-\nu^2}}\overline{P}_n^m \sin(\psi)\sin(m\psi) \\
&= \frac{1}{2}\cos((m+1)\psi)\left(\frac{\sqrt{1-\nu^2}}{\nu}\frac{\partial \overline{P}_n^m}{\partial \nu} + \frac{m}{\sqrt{1-\nu^2}}\overline{P}_n^m\right) \\
&+ \frac{1}{2}\cos((m-1)\psi)\left(\frac{\sqrt{1-\nu^2}}{\nu}\frac{\partial \overline{P}_n^m}{\partial \nu} - \frac{m}{\sqrt{1-\nu^2}}\overline{P}_n^m\right) \\
&= \frac{1}{2}\cos((m+1)\psi)\left(\sqrt{(n+m+1)(n-m)}\frac{\overline{P}_n^{m+1}}{\nu}\right) \\
&- \frac{1}{2}\cos((m-1)\psi)\left(\sqrt{(n+m)(n-m+1)}\frac{\overline{P}_n^{m-1}}{\nu}\right)
\end{aligned}
$$

with:

$$
\left.\frac{\sqrt{1-\nu^2}}{\nu}\frac{\partial \overline{P}_n^m}{\partial \nu}\right|_{\eta=0} = \frac{1}{\sqrt{1-\nu^2}}\left(\sqrt{(n+m+1)(n-m)}\frac{\sqrt{1-\nu^2}\,\overline{P}_n^{m+1}}{\nu} - m\overline{P}_n^m\right) \tag{A.1.5}
$$

and:

$$
\left.\frac{\sqrt{1-\nu^2}}{\nu}\frac{\partial \overline{P}_n^m}{\partial \nu}\right|_{\eta=0} = \frac{1}{\sqrt{1-\nu^2}}\left(-\sqrt{(n+m)(n-m+1)}\frac{\sqrt{1-\nu^2}\,\overline{P}_n^{m-1}}{\nu} + m\overline{P}_n^m\right) \tag{A.1.6}
$$

This allows to derive the values to be used in the $[S]$ matrix.

A.2 Ellipsoidal coordinate system

The ellipsoidal coordinate system can be described in terms of the cartesian coordinates as:

$$\begin{cases} x = -\sqrt{1-\nu^2}\sqrt{1+\eta^2}\cos(\psi) \\ y = \sqrt{1-\nu^2}\sqrt{1+\eta^2}\sin(\psi) \\ z = -\nu\eta \end{cases} \tag{A.2.1}$$

The link between the derivatives in cartesian and ellipsoidal coordinates are useful for the derivation of the matrices:

$$\frac{\partial}{\partial z} = \frac{1}{\nu^2+\eta^2}\left(\eta(1-\nu^2)\frac{\partial}{\partial \nu} + \nu(1+\eta^2)\frac{\partial}{\partial \eta}\right) \tag{A.2.2}$$

$$\frac{\partial}{\partial x} = \frac{\sqrt{(1+\eta^2)(1-\nu^2)}}{\nu^2+\eta^2}\left(\nu\frac{\partial}{\partial \nu} + \eta\frac{\partial}{\partial \eta}\right)\cos\psi + \frac{1}{\sqrt{(1+\eta^2)(1-\nu^2)}}\frac{\partial}{\partial \psi}\sin\psi \tag{A.2.3}$$

$$\frac{\partial}{\partial y} = -\frac{\sqrt{(1+\eta^2)(1-\nu^2)}}{\nu^2+\eta^2}\left(\nu\frac{\partial}{\partial \nu} - \eta\frac{\partial}{\partial \eta}\right)\sin\psi + \frac{1}{\sqrt{(1+\eta^2)(1-\nu^2)}}\frac{\partial}{\partial \psi}\cos\psi \tag{A.2.4}$$

Furthermore, these relations show clearly the singularities appearing due to the ellipsoidal coordinates.

A.3 On the exact solution

We here develop a bit around the exact solution, in order to assess our affirmation that it does not respect the continuity equation. To do so, we simply look at a simplified case, where $\chi = 0$, and we develop the divergence of the real part of the exact solution:

$$\begin{aligned} \mathrm{div}\overrightarrow{u}(x,y,z) = &-\frac{\partial}{\partial x}\int_{-\infty}^{z} \cos(\omega(z-\overline{z}))\frac{\partial P}{\partial x}(x,y,\overline{z}d\overline{z} \\ &-\frac{\partial}{\partial y}\int_{-\infty}^{z} \cos(\omega(z-\overline{z}))\frac{\partial P}{\partial y}(x,y,\overline{z})d\overline{z} \\ &-\frac{\partial}{\partial z}\int_{-\infty}^{z} \cos(\omega(z-\overline{z}))\frac{\partial P}{\partial z}(x,y,\overline{z})d\overline{z} \end{aligned} \tag{A.3.1}$$

This form is justified by the fact that when $\chi = 0$ the streamlines are parallel to the z axis. It then can be expanded, using Leibniz's rule:

$$\begin{aligned} \mathrm{div}\overrightarrow{u}(x,y,z) = &-\int_{-\infty}^{z} \cos(\omega(z-\overline{z}))\Delta P(x,y,\overline{z})d\overline{z} \\ &-\frac{\partial z}{\partial z}\cos(\omega(z-z))\frac{\partial P}{\partial z}(x,y,z) \\ &-\int_{-\infty}^{z} \omega\sin(\omega(z-\overline{z}))\frac{\partial P}{\partial z}(x,y,\overline{z})d\overline{z} \end{aligned} \tag{A.3.2}$$

In the general case, this would not simplify further. But using the fact that the pressure is a potential function as in the case of Huang and Peters model, it gives:

$$\operatorname{div}\overrightarrow{u} = -\frac{\partial P}{\partial z} - \int_{-\infty}^{z} \omega \sin(\omega(z-\overline{z}))\frac{\partial P}{\partial z}(x,y,\overline{z})d\overline{z} \tag{A.3.3}$$

This expression has no reason to be null, which shows that in most cases the exact solution does not respect the continuity equation.

A.4 Downstream velocity: problematic domain definition

The definition of the problematic domain answers the question: "what are the conditions for the adjoint downstream velocity time $\tau + \sigma\sin(\chi)$ to be higher than τ?". The answer is however not as obvious as it seems. Indeed, it is first required to need the adjoint downstream velocity. For this, χ must be strictly positive, and the point to be considered must be below the rotor ($z < 0$), in order to require the computation of the adjoint.

Let's consider a point (x, y, z) below the rotor. Computing the induced velocities at this point will require the values of the adjoint of V_F at time τ at $(-x, -y, -z)$ and at time $\tau - \xi_0$ at $(-x_0, -y_0, 0)$. (The notations used here are all defined in section 3.3.1.1) It is now required to see if these adjoint V_F values will require the computation of the adjoint downstream velocity. Following its definition given by Huang in [1], one can see that the downstream velocity is required for points with $x \leq 0$ and $|x^2 + y^2 + z^2| \geq \rho$, where $\rho = 1$ for the axial velocity and $\rho = \cos(\chi)$ for the other components of the velocity.

It follows that the points requiring the adjoint downstream velocity are points respecting $-x \leq 0$ and $|x^2 + y^2 + z^2| \geq \rho$, hence the definition of the domain D:

$$D = \{(x,y,z)\,|\,x^2+y^2+z^2 \geq \cos(\chi)\,\text{and}\,x \leq 0, z < 0\} \tag{A.4.1}$$

B Gauss-Lobatto quadrature

B.1 Principle

The Gauss-Lobatto quadrature is defined by its choice of quadrature points and weights. The $n+1$ points of the quadrature are defined as the roots of P'_{n-1} plus the two boundary points, where P_n is an orthogonal polynomial. In the case where the chosen orthogonal family is the Legendre polynomials L_n, the method is called a Legendre-Gauss-Lobatto quadrature.

The weights are defined as:

$$w_i = \frac{2}{n(n-1)\left(P_{n-1}(x_i)\right)^2} \tag{B.1.1}$$

This method is exact for polynomials of degree less or equal to $2n-3$, with n the number of integration points.

All this information was found in [73].

B.2 Implementation

As mentioned in section 5.2.6, a quadrature method is used to compute the non linear terms. Since the azimuthal part is dealt with a Fourier approximation, the non linearity in azimuth can be dealt separately without any quadrature.

For the r and z approximation, and for a general non linear term $v_1(r,z)v_2(r,z)$ projected on ϕ_k and ψ_l, a quadrature with n_r points on the radial part and n_z points on the axial part gives the following formulation:

$$\int_{-1}^{1}\int_{-1}^{1} v_1(r,z)v_2(r,z)\phi_k(r)\psi_l(z)rdrdz = \sum_{i=0}^{n_r}\sum_{j=0}^{n_z} r_i v_1(r_i,z_j)v_2(r_i,z_j)\phi_k(r_i)\omega_i^r\psi_l(z_j)\omega_j^z \tag{B.2.1}$$

However, in the case of the velocity, we use its spectral representation, which gives:

$$v_1(r_i,z_j) = \sum_{n=0}^{N_r}\sum_{h=0}^{N_z} v_1^{n,h}\phi_n(r_i)\psi_h(z_j) \tag{B.2.2}$$

If we define matrices Q^ϕ containing values of the polynomials ϕ at the quadrature points as:

$$(Q^\phi)_{n,i} = \phi_n(x_i) \tag{B.2.3}$$

And the matrix Q as:

$$Q = Q^\phi \otimes Q^\psi \tag{B.2.4}$$

Then one can compute the values of the velocity at the collocation points as:

$$v_1(r_i, z_j) = \left(Q\hat{V}_1\right)_{i\,N_r+j} \tag{B.2.5}$$

where $\hat{V}_1$ is the vector of the coefficients of the approximation of v_1.

This then allows to write:

$$\int_{-1}^{1}\int_{-1}^{1} v_1(r,z)v_2(r,z)\phi_k(r)\psi_l(z)rdrdz = \left[Q^T\left(\left(Q\hat{V}_1\right) * \left(Q\hat{V}_2\right) * W\right)\right]_{k\,N_r+l} \tag{B.2.6}$$

Where $*$ is the Hadamar product (element-wise multiplication), and $W = w_r \otimes w_z$ is the vector of weights where $(w_r)_i = r_i\omega_i^r$ and $(w_z)_j = \omega_j^z$

This gives a simple expression for the computation of the quadrature for the non linear terms. However, depending on the non linear term to be expressed, the matrices Q^ϕ and Q^ψ will have to be adapted. It is to be noted that this formulation can be adapted to the mini matrices formulation easily, by using the Kronecker product of the Q matrix.

C

Functional normed spaces

We used several various normed functional spaces in the thesis, that we define properly here.

C.1 L^p_ω spaces

The $L^p_\omega(I)$ spaces designate the functional space of function defined over the domain I equipped with the norm:

$$\|f\|^p_\omega = \left(\int_I |f(x)|^p \omega(x) dx \right)^{\frac{1}{p}} \tag{C.1.1}$$

And is defined by:

$$L^p_\omega(I) = \{ f \mid \|f\|^p_\omega < \infty \} \tag{C.1.2}$$

The $L^2_\omega(I)$ space is of particular interest since its norm derives from a scalar product. It is the space of square integrable functions.

C.2 Sobolev and Hilbert spaces

The Sobolev spaces are also normed functional spaces, but in their case the norm with which they are equipped is built with combination of L^p_ω norms, on the function itself, or on its derivatives, and can be defined as follows:

$$W^{p,m}_\omega(I) = \left\{ f \in L^p_\omega(I) \mid \forall n,\, |n| < m,\, \frac{\partial^n f}{\partial x^n} \in L^p_\omega(I) \right\} \tag{C.2.1}$$

However the definition of the associated norm might vary. In the case where $p = 2$, the space considered is called an Hilbert space and denoted $H^m_\omega(I)$.

D

A method with less inversion

We present in the appendix a little more details about the development of the improvement of the Morillo-Duffy model, presented in chapter 4.2.1.

We therefore have the following equation:

$$\begin{aligned}\mathop{+}\limits_{m,n} \iiint\limits_D \frac{\partial a_n^m}{\partial t}\overrightarrow{\text{grad}}\Psi_n^m.\overrightarrow{\text{grad}}\Lambda_j^r + a_n^m\overrightarrow{\text{grad}}(\overrightarrow{\text{grad}}\Psi_n^m.\overrightarrow{\xi}).\overrightarrow{\text{grad}}\Lambda_j^r dD = \\ \mathop{+}\limits_{m,n} \iiint\limits_D \tau_n^m\overrightarrow{\text{grad}}\Phi_n^m.\overrightarrow{\text{grad}}\Lambda_j^r dD\end{aligned} \tag{D.0.1}$$

Which can be put under the following matrix form:

$$A_1\frac{d}{dt}(\{a_n^m\}) + (\cos(\chi)D_1 - \sin(\chi)S_1)\{a_n^m\} = M_\tau\{\tau_n^m\} \tag{D.0.2}$$

We define:

$$\Theta_n^m = \sigma_n^m\Phi_{n+1}^m + \varsigma_n^m\Phi_{n-1}^m + \sigma_{n+1}^{m+1}\Phi_{n+2}^{m+1} + \varsigma_{n+1}^{m+1}\Phi_n^{m+1} \tag{D.0.3}$$

We then refer to the following table for the choices made about the placement of the z derivatives. The table present the choice of velocity potential and of test function, and the various matrix terms to be computed, depending on the parity of the terms involved. If the derivative must change it is indicated by the word 'Swap'.

		odd/odd	odd/even	even/odd	even/even
Ψ_n^m		Θ_n^m			
Λ_j^r		$\sigma_j^r\Phi_{j+1}^r + \varsigma_j^r\Phi_{j-1}^r$			
A_1	$\iint\limits_S \Psi_n^m\frac{\partial\Lambda_j^r}{\partial z}dS$	even/odd	Swap: odd/odd	odd/odd	odd/even
D_1	$\iint\limits_S \frac{\partial\Psi_n^m}{\partial z}\frac{\partial\Lambda_j^r}{\partial z}dS$	odd/odd	odd/even (SP)	even/odd (SP)	Swap: odd/odd
S_1	$\iint\limits_S \frac{\partial\Psi_n^m}{\partial x}\frac{\partial\Lambda_j^r}{\partial z}dS$	even/odd (SP)	Swap odd/odd	odd/odd	odd/even (SP)
M_τ	$\iint\limits_S \Phi_n^m\frac{\partial\Lambda_j^r}{\partial z}dS$	odd/odd	odd/even	even/odd	Swap: odd/odd

One can then compute the expression of the various matrices. For A_1: If $m+n$ odd and $r+j$ even:

$$
\begin{aligned}
[A_1]_{n,j}^{m,r} &= \iint_S \frac{\partial \Psi_n^m}{\partial z} \Lambda_j^r dS \\
&= \iint_A (\Phi_n^m + \Phi_{n+1}^{m+1})(\sigma_j^r \Phi_{j+1}^r + \varsigma_j^r \Phi_{j-1}^r) dA \\
&= \delta_{m,r} \left(\sigma_j^r \int_0^1 P_n^m \, P_{j+1}^r \nu d\nu + \varsigma_j^r \int_0^1 P_n^m \, P_{j-1}^r \nu d\nu \right) \\
&+ \delta_{m+1,r} \left(\sigma_j^r \int_0^1 P_{n+1}^{m+1} \, P_{j+1}^r \nu d\nu + \varsigma_j^r \int_0^1 P_{n+1}^{m+1} \, P_{j-1}^r \nu d\nu \right)
\end{aligned}
$$

Otherwise:

$$
\begin{aligned}
[A_1]_{n,j}^{m,r} &= \iint_S \Psi_n^m \frac{\partial \Lambda_j^r}{\partial z} dS \\
&= \iint_A \Theta_n^m \Phi_j^r dA \\
&= \delta_{m,r} \left(\sigma_n^m \int_0^1 P_{n+1}^m \, P_j^r \nu d\nu + \varsigma_n^m \int_0^1 P_{n-1}^m \, P_j^r \nu d\nu \right) \\
&+ \delta_{m+1,r} \left(\sigma_{n+1}^{m+1} \int_0^1 P_{n+2}^{m+1} \, P_j^r \nu d\nu + \varsigma_{n+1}^{m+1} \int_0^1 P_n^{m+1} \, P_j^r \nu d\nu \right)
\end{aligned}
$$

For D_1: For $m+n$ even and $r+j$ even:

$$
\begin{aligned}
[D_1]_{n,j}^{m,r} &= \iint_S \frac{\partial^2 \Psi_n^m}{\partial z^2} \Lambda_j^r dS \\
&= \iint_A \frac{\partial}{\partial z} \left(\Phi_n^m + \Phi_{n+1}^{m+1} \right) \left(\sigma_j^r \Phi_{j+1}^r + \varsigma_j^r \Phi_{j-1}^r \right) dA \\
&= \delta_{m,r} \frac{\partial Q_n^m}{\partial \eta}(i0) \left(\sigma_j^r \int_0^1 P_n^m \, P_{j+1}^r d\nu + \varsigma_j^r \int_0^1 P_n^m \, P_{j-1}^r d\nu \right) \\
&+ \delta_{m+1,r} \frac{\partial Q_{n+1}^{m+1}}{\partial \eta}(i0) \left(\sigma_j^r \int_0^1 P_{n+1}^{m+1} \, P_{j+1}^r d\nu + \varsigma_j^r \int_0^1 P_{n+1}^{m+1} \, P_{j-1}^r d\nu \right)
\end{aligned}
$$

Otherwise:

$$
\begin{aligned}
[D_1]^{m,r}_{n,j} &= \iint\limits_S \frac{\partial \Psi^m_n}{\partial z}\frac{\partial \Lambda^r_j}{\partial z} dS \\
&= \iint\limits_A (\Phi^m_n + \Phi^{m+1}_{n+1})\Phi^r_j dA \\
&= \delta_{m,r}\left(\int\limits_0^1 P^m_n\, P^r_j \nu d\nu\right) + \delta_{m+1,r}\left(\int\limits_0^1 P^{m+1}_{n+1}\, P^r_j \nu d\nu\right)
\end{aligned}
$$

For S_1:

We define γ^m_n and κ^m_n variables as follows, in order to simplify the expression of the matrices:

$$
\begin{aligned}
&\forall m > 0, \forall n > m : \\
&\quad \gamma^m_n = \frac{1}{2}\sqrt{(n+m+1)(n-m)} \\
&\quad \kappa^m_n = -\frac{1}{2}\sqrt{(n+m)(n-m+1)} \\
&\text{If } m = 0, \forall n > m : \\
&\quad \gamma^m_n = \sqrt{(n+1)(n)} \\
&\quad \kappa^m_n = 0
\end{aligned}
$$

For $m+n$ odd and $r+j$ even:

$$
\begin{aligned}
[S_1]^{m,r}_{n,j} &= \iint\limits_S \frac{\partial^2 \Psi^m_n}{\partial z \partial x}\Lambda^r_j dS \\
&= \iint\limits_A \frac{\partial}{\partial x}(\Phi^m_n + \Phi^{m+1}_{n+1})(\sigma^r_j \Phi^r_{j+1} + \varsigma^r_j \Phi^r_{j-1}) dA \\
&= \iint\limits_A \left(\gamma^m_n \frac{\Phi^{m+1}_n}{\nu} + \kappa^m_n \frac{\Phi^{m-1}_n}{\nu} + \gamma^{m+1}_{n+1}\frac{\Phi^{m+2}_{n+1}}{\nu} + \kappa^{m+1}_{n+1}\frac{\Phi^m_{n+1}}{\nu}\right)(\sigma^r_j \Phi^r_{j+1} + \varsigma^r_j \Phi^r_{j-1}) dA \\
&= \delta_{m+1,r}\,\gamma^m_n \left(\sigma^r_j \int\limits_0^1 P^{m+1}_n P^r_{j+1} d\nu + \varsigma^r_j \int\limits_0^1 P^{m+1}_n P^r_{j-1} d\nu\right) \\
&+ \delta_{m-1,r}\,\kappa^m_n \left(\sigma^r_j \int\limits_0^1 P^{m-1}_n P^r_{j+1} d\nu + \varsigma^r_j \int\limits_0^1 P^{m-1}_n P^r_{j-1} d\nu\right) \\
&+ \delta_{m+2,r}\,\gamma^{m+1}_{n+1} \left(\sigma^r_j \int\limits_0^1 P^{m+2}_{n+1} P^r_{j+1} d\nu + \varsigma^r_j \int\limits_0^1 P^{m+2}_{n+1} P^r_{j-1} d\nu\right) \\
&+ \delta_{m,r}\,\kappa^{m+1}_{n+1} \left(\sigma^r_j \int\limits_0^1 P^m_{n+1} P^r_{j+1} d\nu + \varsigma^r_j \int\limits_0^1 P^m_{n+1} P^r_{j-1} d\nu\right)
\end{aligned}
$$

Otherwise:

$$
\begin{aligned}
[S_1]_{n,j}^{m,r} &= \iint\limits_S \frac{\partial \Psi_n^m}{\partial x}\frac{\partial \Lambda_j^r}{\partial z} dS \\
&= \iint\limits_A \frac{\partial \Theta_n^m}{\partial x}\Phi_j^r dA \\
&= \iint\limits_A \left(\sigma_n^m\left(\gamma_{n+1}^m\frac{\Phi_{n+1}^{m+1}}{\nu}+\kappa_{n+1}^m\frac{\Phi_{n+1}^{m-1}}{\nu}\right)+\varsigma_n^m\left(\gamma_{n-1}^m\frac{\Phi_{n-1}^{m+1}}{\nu}+\kappa_{n-1}^m\frac{\Phi_{n-1}^{m-1}}{\nu}\right)\right. \\
&+ \left.\sigma_{n+1}^{m+1}\left(\gamma_{n+2}^{m+1}\frac{\Phi_{n+2}^{m+2}}{\nu}+\kappa_{n+2}^{m+1}\frac{\Phi_{n+2}^{m}}{\nu}\right)+\varsigma_{n+1}^{m+1}\left(\gamma_n^{m+1}\frac{\Phi_n^{m+2}}{\nu}+\kappa_n^{m+1}\frac{\Phi_n^m}{\nu}\right)\right)\Phi_j^r dA \\
&= \delta_{r,m-1}\left(\sigma_n^m\kappa_{n+1}^m\int_0^1 P_{n+1}^{m-1}P_j^r d\nu+\varsigma_n^m\kappa_{n-1}^m\int_0^1 P_{n-1}^{m-1}P_j^r d\nu\right) \\
&+ \delta_{r,m}\left(\sigma_{n+1}^{m+1}\kappa_{n+2}^{m+1}\int_0^1 P_{n+2}^{m}P_j^r d\nu+\varsigma_{n+1}^{m+1}\kappa_{n}^{m+1}\int_0^1 P_{n}^{m}P_j^r d\nu\right) \\
&+ \delta_{r,m+1}\left(\sigma_n^m\gamma_{n+1}^m\int_0^1 P_{n+1}^{m+1}P_j^r d\nu+\varsigma_n^m\gamma_{n-1}^m\int_0^1 P_{n-1}^{m+1}P_j^r d\nu\right) \\
&+ \delta_{r,m+2}\left(\sigma_{n+1}^{m+1}\gamma_{n+2}^{m+1}\int_0^1 P_{n+2}^{m+2}P_j^r d\nu+\varsigma_{n+1}^{m+1}\gamma_{n}^{m+1}\int_0^1 P_{n}^{m+2}P_j^r d\nu\right)
\end{aligned}
$$

And finally, the M_τ matrix is fairly similar to the M matrix with the exception of the even/even terms.

It is to be noted that the above equations are all divided by π with the exception of the $m = r = 0$ equations that are divided by 2π. The above formulae are for the general case where no $m = n$ terms appear. In the case were $m = n$ and the case $m = n+1$, some special treatment needs to be done. Some are handled by the definition of the γ_n^m and κ_n^m variables, the others are presented hereafter.

For S_1 in the case $m = n > 0$, the x derivative of the speed potential can be expressed as follow:

$$
\left.\frac{\partial \Phi_m^m}{\partial x}\right|_{\eta=0} = -m\sqrt{\frac{2m+1}{2m}}\Phi_{m-1}^{m-1} \tag{D.0.4}
$$

Giving for the expression of the $m = n > 0$ terms of S_1:

$$
\begin{aligned}
[S_1]_{m,j}^{m,r} &= \iint\limits_A \frac{\partial}{\partial x}(\Phi_m^m + \Phi_{m+1}^{m+1})(\sigma_j^r \Phi_{j+1}^r + \varsigma_j^r \Phi_{j-1}^r) dA \\
&= \iint\limits_A \left((-m)\sqrt{\frac{2m+1}{2m}} \Phi_{m-1}^{m-1} - (m+1)\sqrt{\frac{2m+3}{2m+2}} \Phi_m^m \right) (\sigma_j^r \Phi_{j+1}^r + \varsigma_j^r \Phi_{j-1}^r) dA \\
&= \delta_{m,r}(-m)\sqrt{\frac{2m+1}{2m}} \left(\sigma_j^r \int_0^1 P_{m-1}^{m-1} P_{j+1}^r \nu d\nu + \varsigma_j^r \int_0^1 P_{m-1}^{m-1} P_{j-1}^r \nu d\nu \right) \\
&+ \delta_{m-1,r}(-(m+1))\sqrt{\frac{2m+3}{2m+2}} \left(\sigma_j^r \int_0^1 P_m^m P_{j+1}^r \nu d\nu + \varsigma_j^r \int_0^1 P_m^m P_{j-1}^r \nu d\nu \right)
\end{aligned}
$$

Adding the matrix allowing to pass from one projection to the one used by Morillo: For P_1: For $m + n$ even and $r + j$ even:

$$
\begin{aligned}
[P_1]_{n,j}^{m,r} &= \iint\limits_S \Psi_n^m \frac{\partial \Phi_j^r}{\partial z} dS \\
&= \iint\limits_A \Theta_n^m \frac{\partial \Phi_j^r}{\partial z} dA \\
&= \iint\limits_A \left(\sigma_n^m \Phi_{n+1}^m + \varsigma_n^m \Phi_{n-1}^m + \sigma_{n+1}^{m+1} \Phi_{n+2}^{m+1} + \varsigma_{n+1}^{m+1} \Phi_n^{m+1} \right) \frac{\partial \Phi_j^r}{\partial z} dA \\
&= \delta_{m,r} \frac{\partial Q_j^r}{\partial \eta}(i0) \left(\sigma_n^m \int_0^1 P_{n+1}^m \, P_j^r d\nu + \varsigma_n^m \int_0^1 P_{n-1}^m \, P_j^r d\nu \right) \\
&+ \delta_{m+1,r} \frac{\partial Q_j^r}{\partial \eta}(i0) \left(\sigma_{n+1}^{m+1} \int_0^1 P_{n+2}^{m+1} \, P_j^r d\nu + \varsigma_{n+1}^{m+1} \int_0^1 P_n^{m+1} \, P_j^r d\nu \right)
\end{aligned}
$$

Otherwise:

$$
\begin{aligned}
[P_1]_{n,j}^{m,r} &= \iint\limits_S \frac{\partial \Psi_n^m}{\partial z} \Phi_j^r dS \\
&= \iint\limits_A (\Phi_n^m + \Phi_{n+1}^{m+1}) \Phi_j^r dA \\
&= \delta_{m,r} \left(\int_0^1 P_n^m \, P_j^r \nu d\nu \right) + \delta_{m+1,r} \left(\int_0^1 P_{n+1}^{m+1} \, P_j^r \nu d\nu \right)
\end{aligned}
$$

This method allows for good replication of the Morillo results. Stability still requires some analysis, notably for the S_1 matrix, and many cases of computation are to be tested. The coefficients α_ψ and β_ψ might also require some optimisation.

Concerning the slow convergence of this kind of method, it can be improved through the use of the Cesaro mean, as shown below. Indeed, good convergence is equivalent to having coefficients describing the speed potential that converges to 0 as m and n increase.

Since the Cesaro mean method takes the mean of the partial sum, it tends to improve convergence and the operation can be repeated if necessary.

Thus the computation of the velocity that was taken under the form:

$$\overrightarrow{V} = \overrightarrow{\text{grad}} \sum_{m=0}^{M} \sum_{n=m+1}^{N(M,m)} a_n^m \Psi_n^m \tag{D.0.5}$$

where

$$N(M,m) = m + \left\lfloor \frac{M}{2} \right\rfloor + 1 - \left\lceil \frac{m}{2} \right\rceil \tag{D.0.6}$$

Dealing only with the potentials and taking the mean of the partial sums gives:

$$S_K = \frac{1}{K+1} \sum_{k=0}^{K} \sum_{m=0}^{k} \sum_{n=m+1}^{N(k,m)} a_n^m \Psi_n^m \tag{D.0.7}$$

Which gives when regrouping the identical terms:

$$S_K = \sum_{k=0}^{K} \sum_{n=k+1}^{N(K,k)} \frac{K-n+2}{K+1} a_n^k \Psi_n^k \tag{D.0.8}$$

Other point of view: we are solely interested in the projection of the equation on the subspace described by the Λ_j^r functions. They leave the liberty of the choice of scalar product, allowing to take a more permissive one with regard to the derivations.

For example taking:

$$\langle f, g \rangle = \iint\limits_S f g \nu dS \tag{D.0.9}$$

as scalar product would allow to take as potential function

$\mathcal{E}$

Legendre Kernel and mollification

This appendix is interested in a method of removing or rather mitigating the Gibbs phenomenon, through the use of mollifiers. This idea is primarily developed for Fourier transform, but can be adapted for Legendre polynomials, as will be shown in the second section of this appendix. In order to understand the ideas behind this, we need to introduce the concept of kernels.

E.1 Kernels

The principle of a kernel is to translate the interpolation of the projection by a convolution. For example the fourier transform is linked to the Dirichlet kernel:

$$S_N(f)(x) = \int_{-\pi}^{\pi} f(y) D_N(x-y) dy = (D_N * f)(x) \tag{E.1.1}$$

where:

$$D_N(x) = \frac{1}{2\pi} \sum_{k=-N}^{N} e^{ikx} \tag{E.1.2}$$

In order to see the projection with Legendre polynomials as a convolution (although it will not be strictly speaking a convolution), one need to consider the normed Legendre polynomials, in order to tackle an orthonormal basis.

One can then write:

$$\begin{aligned} I_N(f)(x) &= \sum_{k=0}^{N} \hat{f}_k \phi_k(x) \\ &= \sum_{k=0}^{N} \int_{-1}^{1} f(y) \phi_k(y) dy \phi_k(x) \\ &= \int_{-1}^{1} f(y) \sum_{k=0}^{N} \phi_k(x) \phi_k(y) dy \end{aligned} \tag{E.1.3}$$

which yields the kernel K_N, which can be presented in another form with the Christoffel-Darboux formula:

$$
\begin{aligned}
K_N(x,y) &= \int_{-1}^{1} \sum_{k=0}^{N} \phi_k(x)\phi_k(y)dy \\
&= \int_{-1}^{1} \frac{n+1}{2} \frac{\phi_N(y)\phi_{N+1}(x) - \phi_N(x)\phi_{N+1}(y)}{x-y} dy
\end{aligned}
\tag{E.1.4}
$$

Combining with a new operator $\otimes$:

$$I_N(f)(x) = (K_N \otimes f)(x) \tag{E.1.5}$$

We thus see that one can see the whole projection and interpolation process as one single operation, with all its content represented by the kernel K_N.

Furthermore, this point of view of the kernel allows to cast a new light on the vision one has of the filters. Indeed, the Fejèr filter is for example linked to the Fejèr kernel since:

$$S_N^F(f)(x) = \int_{-\pi}^{\pi} f(y)F_N(x-y)dy = (F_N * f)(x) \tag{E.1.6}$$

with:

$$F_N(x) = \frac{1}{N} \sum_{k=0}^{N-1} D_k(x) \tag{E.1.7}$$

One can here see that the Fejèr kernel is called a positive summability kernel, which represents the property one seeks when using it, that it highly dampens the Gibbs phenomenon, since it will experience little overshoot.

E.2 Mollifier

The main idea behind a mollifier is to use a modified kernel in order to retrieve the spectral information. The most early forms of this can be seen with the Fejèr kernel. This modification of the kernel can be easily translated by a linear filter, as it is the Cesaro mean of the partial sum of the interpolation. We here see the link one can make between mollifiers and filtering.

However the main forms of mollifiers, as described in [51], are in fact highly localised. Furthermore adaptive mollifiers are an active field of research [48, 50, 74], and give impressiv results in the treatment of discontinuities. However, those methods are first quite computationally expensive, and then are only post treatment methods, that only give a better representation of the approximation from Gibbs contaminated coefficients, and not better values for those coefficients. They are therefore not adapted to our application.

Bibliography

[1] Jianzhe Huang. *Potential-flow Inflow Model Including Wake Distortion and Contraction.* Doctor of Science Dissertation, Washington University in St. Louis, 2015.

[2] Jianzhe Huang, David A Peters, and JVR Prasad. "Converged Velocity Field for Rotors by a Blended Potential Flow Method". In: *GSTF Journal on Aviation Technology (JAT)* 1.2 (2015).

[3] JM Lopez and Jie Shen. "An efficient spectral-projection method for the Navier–Stokes equations in cylindrical geometries: I. Axisymmetric cases". In: *Journal of Computational Physics* 139.2 (1998), pp. 308–326.

[4] JM Lopez, F Marques, and Jie Shen. "An efficient spectral-projection method for the Navier–Stokes equations in cylindrical geometries: II. Three-dimensional cases". In: *Journal of Computational Physics* 176.2 (2002), pp. 384–401.

[5] Ed Barocela et al. "The effect of wake distortion on rotor inflow gradients and off-axis coupling". In: *22nd Atmospheric Flight Mechanics Conference*. 1997, p. 3579.

[6] Frederic Le Chuiton. "Actuator disc modelling for helicopter rotors". In: *Aerospace Science and Technology* 8.4 (2004), pp. 285–297.

[7] Seong Yong Wie, Seongkyu Lee, and Duck Joo Lee. "Potential panel and time-marching free-wake-coupling analysis for helicopter rotor". In: *Journal of Aircraft* 46.3 (2009), pp. 1030–1041.

[8] David A Peters and Cheng Jian He. "Finite state induced flow models part II: three-dimensional rotor disk". In: *Journal of Aircraft* 32.2 (1995), pp. 323–333.

[9] Kenneth B Amer. "Theory of helicopter damping in pitch or roll and a comparison with flight measurements". In: (1950).

[10] D. M. Pitt. *Rotor Dynamic Inflow Derivatives and Time Constants from Various Inflow Models.* Tech. rep. ARMY TROOP SUPPORT and AVIATION MATERIEL READINESS COMMAND ST LOUIS MO, 1983.

[11] Dale M Pitt and David A Peters. "Theoretical prediction of dynamic-inflow derivatives". In: *Vertica* 5 (1980), pp. 21–34.

[12] M Joglekar and R Loewy. *An actuator-disc analysis of helicopter wake geometry and the corresponding blade response.* Tech. rep. ROCHESTER UNIV NY, 1970.

[13] David A Peters, David Doug Boyd, and Cheng Jian He. "Finite-State Induced-Flow Model for Rotors in Hover and Forward Flight". In: *Journal of the American Helicopter Society* 34.4 (1989), pp. 5–17.

[14] Ay Su, Kyung M Yoo, and David A Peters. "Extension and validation of an unsteady wake model for rotors". In: *Journal of Aircraft* 29.3 (1992), pp. 374–383.

[15] David A Peters, D Barwey, and A Su. "An integrated airloads-inflow model for use in rotor aeroelasticity and control analysis". In: *Mathematical and computer modelling* 19.3-4 (1994), pp. 109–123.

[16] Pierre-Marie Basset and RA Ormiston. "Comparison and validation of the France/USA finite state rotor dynamic inflow models". In: 36th European Rotorcraft Forum, Paris, France. Sept. 2010.

[17] Bernard Benoit et al. "HOST, a general helicopter simulation tool for Germany and France". In: *Annual Forum Proceedings-American Helicopter Society*. Vol. 56. 2. American Helicopter Society, Inc. 2000, pp. 1110–1131.

[18] Pierre-Marie Basset et al. "Finite state rotor induced flow model for interference and ground effect". In: *Annual Forum Proceedings-American Helicopter Society*. Vol. 57. 1. American Helicopter Society, Inc. 2001.

[19] Jorge A. Morillo. *A Fully Three-Dimensional Unsteady Rotor Inflow Model from a Galerkin Approach*. Master of Science Thesis, Washington University in St. Louis, 2001.

[20] Jorge A Morillo and David A Peters. "Velocity field above a rotor disk by a new dynamic inflow model". In: *Journal of Aircraft* 39.5 (2002), pp. 731–738.

[21] Antonio Hsieh. *A Complete Finite-State Model for Rotors in Axial Flow*. Doctor of Science Dissertation, Washington University in St. Louis, 2006.

[22] David A Peters, Antonio Hsieh, and Cristina Garcia-Duffy. "A complete finite-state inflow theory from the potential flow equations". In: *3rd International Basic Research Conference on Rotorcraft Technology, Nanjing, China*. 2009.

[23] Zhongyang Fei and David A Peters. "Inflow below the rotor disk for skewed flow by the finite-state, adjoint method". In: *the 38th European Rotorcraft Forum*. 2012.

[24] Zhongyang Fei. *A Rigorous Solution for Finite-State Inflow throughout the Flowfield*. Doctor of Science Dissertation, Washington University in St. Louis, 2013.

[25] Paul J Carpenter and Bernard Fridovich. *Effect of a rapid blade-pitch increase on the thrust and induced-velocity response of a full-scale helicopter rotor*. Tech. rep. TN-3044. NACA, Nov. 1953.

[26] Robert A Ormiston and David A Peters. "Hingeless Rotor Response with Nonuniform Induced Flow and Elastic Blade Bending". In: *Journal of Aircraft* 9.10 (1972), pp. 730–736.

[27] David A Peters. "The importance of steady and dynamic inflow on the stability of rotor-body systems". In: ITR Methodology Workshop. NASA-Ames Research Center, June 1988.

[28] David A Peters and Ninh HaQuang. "Dynamic inflow for practical applications". In: *JAHS* 33.4 (Oct. 1988).

[29] Chengjian He, CS Lee, and Weibin Chen. "Finite state induced flow model in vortex ring state". In: *Journal of the American Helicopter Society* 45.4 (2000), pp. 318–320.

[30] David A Peters and Chengjian He. "Modification of Mass-Flow Parameter to Allow Smooth Transition Between Helicopter and Windmill States". In: *Journal of the American Helicopter Society* 51.3 (2006), pp. 275–278.

[31] Y Murakami and SS Houston. "Correction of the definition of mass flow parameter in dynamic inflow modelling". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 223.7 (2009), pp. 1037–1040.

[32] Aviv Rosen and Aharon Isser. "A model of the unsteady aerodynamics of a hovering helicopter rotor that includes variations of the wake geometry". In: *Journal of the American Helicopter Society* 40.3 (1995), pp. 6–16.

[33] Ioannis Goulos. "An Improved Analytical Approach for Modeling the Effect of Rotor Wake Curvature Using Finite-State Induced Flow Models". In: *Journal of the American Helicopter Society* 61.3 (2016), pp. 1–16.

[34] Jeffrey D Keller. "An investigation of helicopter dynamic coupling using an analytical model". In: *Journal of the American Helicopter Society* 41.4 (1996), pp. 322–330.

[35] Edward Barocela. "The effect of wake distortion on dynamic inflow for lifting rotors". PhD thesis. Washington University, 1997.

[36] Pierre-Marie Basset and Frederic Tchen-Fo. "Study of the rotor wake distortion effects on the helicopter pitch". In: *24th European Rotorcraft Forum, Marseilles (France), paper FM06* (Sept. 1998).

[37] Krishnamohan R Krothapalli, JVR Prasad, and David A Peters. "Helicopter rotor dynamic inflow modeling for maneuvering flight". In: *Journal of the American Helicopter Society* 46.2 (2001), pp. 129–139.

[38] Jinggen Zhao, JVR Prasad, and David A Peters. "Rotor dynamic wake distortion model for helicopter maneuvering flight". In: *Journal of the American Helicopter Society* 49.4 (2004), pp. 414–424.

[39] Jinggen Zhao. "Dynamic wake distortion model for helicopter maneuvering flight". PhD thesis. Georgia Institute of Technology, 2005.

[40] John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

[41] Claudio Canuto and Alfio Quarteroni. "Approximation results for orthogonal polynomials in Sobolev spaces". In: *Mathematics of Computation* 38.157 (1982), pp. 67–86.

[42] Jie Shen. "Efficient spectral-Galerkin method I. Direct solvers of second-and fourth-order equations using Legendre polynomials". In: *SIAM Journal on Scientific Computing* 15.6 (1994), pp. 1489–1505.

[43] Jie Shen. "Efficient spectral-Galerkin methods III: Polar and cylindrical geometries". In: *SIAM Journal on Scientific Computing* 18.6 (1997), pp. 1583–1604.

[44] David Gottlieb and Jan S Hesthaven. "Spectral methods for hyperbolic problems". In: *Partial Differential Equations*. Elsevier, 2001, pp. 83–131.

[45] JC Simo and F Armero. "Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations". In: *Computer Methods in Applied Mechanics and Engineering* 111.1-2 (1994), pp. 111–154.

[46] Zhanwen Yang, Jiwei Zhang, and Chengchao Zhao. "Numerical blow-up analysis of linearly implicit Euler method for nonlinear parabolic integro-differential equations". In: *Journal of Computational and Applied Mathematics* (2019).

[47] David Gottlieb and Chi-Wang Shu. "On the Gibbs phenomenon and its resolution". In: *SIAM review* 39.4 (1997), pp. 644–668.

[48] David Gottlieb and Eitan Tadmor. "Recovering pointwise values of discontinuous data within spectral accuracy". In: *Progress and supercomputing in computational fluid dynamics*. Springer. 1985, pp. 357–375.

[49] Yvon Maday, Sidi M Ould Kaber, and Eitan Tadmor. "Legendre pseudospectral viscosity method for nonlinear conservation laws". In: *SIAM Journal on Numerical Analysis* 30.2 (1993), pp. 321–342.

[50] Joanna Piotrowska, Jonah M Miller, and Erik Schnetter. "Spectral Methods in the Presence of Discontinuities". In: *arXiv preprint arXiv:1712.09952* (2017).

[51] Eitan Tadmor. "Filters, mollifiers and the computation of the Gibbs phenomenon". In: *Acta Numerica* 16 (2007), pp. 305–378.

[52] Hervé Vandeven. "Family of spectral filters for discontinuous problems". In: *Journal of Scientific Computing* 6.2 (1991), pp. 159–192.

[53] Jan Hesthaven and Robert Kirby. "Filtering in Legendre spectral methods". In: *Mathematics of Computation* 77.263 (2008), pp. 1425–1452.

[54] Alex Kanevsky, Mark H Carpenter, and Jan S Hesthaven. "Time-consistent filtering in spectral and spectral element methods". In: *J. Comput. Phys., submitted for publication* (2004).

[55] Robert L Sani and Philip M Gresho. "Resume and remarks on the open boundary condition minisymposium". In: *International Journal for Numerical Methods in Fluids* 18.10 (1994), pp. 983–1008.

[56] Michael Philip Kirkpatrick and Steven William Armfield. "Open boundary conditions in numerical simulations of unsteady incompressible flow". In: *ANZIAM Journal* 50 (2008), pp. 760–773.

[57] Robert L Sani et al. "Pressure boundary condition for the time-dependent incompressible Navier–Stokes equations". In: *International Journal for Numerical Methods in Fluids* 50.6 (2006), pp. 673–682.

[58] Jie Liu. "Open and traction boundary conditions for the incompressible Navier–Stokes equations". In: *Journal of Computational Physics* 228.19 (2009), pp. 7250–7267.

[59] Donatella Donatelli, Eduard Feireisl, and Antonin Novotny. "On incompressible limits for the Navier-Stokes system on unbounded domains under slip boundary conditions". In: *Discrete Contin. Dyn. Syst. Ser. B* 13.4 (2010), pp. 783–798.

[60] Suchuan Dong, George E Karniadakis, and C Chryssostomidis. "A robust and accurate outflow boundary condition for incompressible flow simulations on severely-truncated unbounded domains". In: *Journal of Computational Physics* 261 (2014), pp. 83–105.

[61] Suchuan Dong and Jie Shen. "A pressure correction scheme for generalized form of energy-stable open boundary conditions for incompressible flows". In: *Journal of Computational Physics* 291 (2015), pp. 254–278.

[62] J.Luneau and A. Bonnet. *Theories Dynamique des Fluides*. Cepadues Editions, 1989.

[63] Simply Connected Space. *Simply Connected Space — Wikipedia, The Free Encyclopedia*. [Online; accessed 23-October-2019]. 2019. URL: `https://en.wikipedia.org/wiki/Simply_connected_space`.

[64] Cristina Garcia-Duffy. *Applying dynamic wake models to induced power calculations for an optimum rotor*. Tech. rep. WASHINGTON UNIV ST LOUIS MO SCHOOL OF ENGINEERING and APPLIED SCIENCE, 2009.

[65] Pierre-Marie Basset et al. "CREATION: a numerical workshop for rotorcraft concepts generation and evalutation". In: Rotorcraft Virtual Engineering Conference. Liverpool, Nov. 2016.

[66] Jeewoong Kim et al. "Integration of Finite State Dynamic Wake Model forRotor Interference with Comprehensive Rotorcraft Analysis". In: *Vertical Flight Society 75th Annual Forum and Technology Display, Philadelphia, Pennsylvania, May 13–16, 2019*. 2009.

[67] Legendre Polynomials. *Legendre polynomials — Wikipedia, The Free Encyclopedia*. [Online; accessed 23-September-2019]. 2019. URL: `https://en.wikipedia.org/wiki/Legendre_polynomials`.

[68] David O'Brien and Marilyn Smith. "Analysis of rotor-fuselage interactions using various rotor models". In: *43rd AIAA Aerospace Sciences Meeting and Exhibit*. 2005, p. 468.

[69] HA Mavromatis and RS Alassar. "A generalized formula for the integral of three associated Legendre polynomials". In: *Applied mathematics letters* 12.3 (1999), pp. 101–105.

[70] F Domenichini and B Baccani. "A formulation of Navier–Stokes problem in cylindrical coordinates applied to the 3D entry jet in a duct". In: *Journal of Computational Physics* 200.1 (2004), pp. 177–191.

[71] H.M. Blackburn and S.J. Sherwin. "Formulation of a Galerkin spectral element-Fourier method for three-dimensional incompressible flows in cylindrical geometries". In: *Journal of Computational Physics* 197 (2004), pp. 759–778.

[72] Associated Legendre Polynomials. *Associated Legendre Polynomials — Wikipedia, The Free Encyclopedia*. [Online; accessed 23-September-2019]. 2019. URL: `https://en.wikipedia.org/wiki/Associated_Legendre_polynomials`.

[73] Gaussian Quadrature. *Gaussian Quadrature — Wikipedia, The Free Encyclopedia*. [Online; accessed 23-September-2019]. 2019. URL: `https://en.wikipedia.org/wiki/Gaussian_quadrature`.

[74] Eitan Tadmor and Jared Tanner. "Adaptive mollifiers for high resolution recovery of piecewise smooth data from its spectral information". In: *Foundations of Computational Mathematics* 2.2 (2002), pp. 155–189.

Abstract

The rotor wake of a rotorcraft is difficult to model, and shows significant importance for the flight dynamics of the aircraft, vibration, acoustics,... Numerous models exist, but the most successful ones for the application to flight mechanics are the finite states models of Peters et al. They allow to tailor the computational cost and fidelity of the model to each application.

This thesis explores their development and underlines some shortcomings. A few improvements are suggested, but it appears that a more drastic change is required to intrinsically account for some phenomena. Therefore a new method is proposed to develop a more general and homogeneous model, no longer relying on constraining assumptions. This model is however not deprived of flaws. They are therefore treated in order to improve the presented model.

The new model shows great results when a large number of states are used, and the impact of a less precise but faster approximation are underlined.

Résumé en français

Le sillage d'un rotor est difficile à modéliser, et a un impact significatif pour la dynamique du vol des voilures tournantes, leurs vibrations, leurs propriétés acoustiques... De nombreux modèles existent, mais le plus utilisé pour l'application à la mécanique du vol est le modèle à états finis développé par Peters et al. Ce type de modèle permet d'ajuster le prix en temps de calcul et la fidélité de leurs résultats à chaque application. Cette thèse explore le développement de ces modèles et en souligne quelques défauts. Des améliorations sont suggérées mais il semble nécessaire d'appliquer des changements plus drastiques pour dépasser certaines limites. Ainsi, une nouvelle méthode est développée dans un cadre plus générale et homogène, qui ne repose plus sur des hypothèses contraignantes. Cependant ce nouveau modèle n'est pas sans défauts. Ils sont donc analysés et traités afin d'améliorer le modèle. Le nouveau modèle livre donc de bons résultats dans le cas d'un grand nombre d'éléments utilisés, et l'impact d'une approximation plus rapide mais moins précise est souligné.