An Isogeometric Boundary Element Method for Three-Dimensional Steady Lifting Flows

Sotirios Chouliaras
Department of Naval Architecture, Ocean and Marine Engineering
University of Strathclyde, Glasgow

September 2020

Abstract

In this PhD thesis an Isogeometric Boundary Element Method (IGA-BEM) for three-dimensional steady lifting flows based on Morino's [50] formulation is presented. A potential flow assumption is used and the unknown perturbation potential satisfies Laplace's equation. Application of Green's identities leads to a Boundary Integral Equation (BIE) that is enhanced with kinematic and dynamic boundary conditions. Analysis suitable T-splines are used for the representation of all boundary surfaces and the unknown perturbation potential is approximated by the same T-spline basis as the one used for the geometry. The BIE is discretised by enforcing it on the generalised version of Greville points for unstructured T-meshes. A novel numerical application of the so-called Kutta condition is introduced that utilises the advantages of IGA with regard to the smoothness of the trailing edge curve basis functions. This leads to a quadratic system that is solved by a Newton-Raphson iterative scheme. The method is applied for three different test cases and shows good agreement with existing experimental results and superior behaviour when compared to a low order panel method. The effect of the tip singularity on Kutta condition is also investigated for different levels of refinement and positions of the trailing edge collocation points.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my sincere gratitude to the following people, without whom I would not have been able to complete this research.

My supervisor, Dr Panagiotis Kaklis, for offering me the opportunity to pursue a PhD in the first place and his constant support, encouragement and patience throughout this journey.

My colleague, collaborator and, above all, friend Theofanis Katsoulis for his support in my research and life in general.

My collaborators, Dr Constantinos Politis, Dr Konstantinos Kostas and Dr Alexandros Ginnis, for their contribution in various aspects of my research since the very early stages of it.

Dr Ioannis Emiris and Dr Christos Konaxis from ATHENA, the coordinators of ARCADES, for organising and running the project that funded my research.

Mark Gammon and Sam Whyman from ITI, for their guidance and support during my secondment at ITI.

Dr Bernard Murrain and the rest of the Aromath team from INRIA, for their contribution to my research during my secondment at INRIA.

My colleagues and friends Christos, Michalis and Mitchell for all the good coffee and chat in the office.

Last but not least, my family members Panagiotis, Christina and Efi, for their continuous love, help and support.

This work has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant *"ARCADES: Algebraic Representations in Computer-Aided Design for complEx Shapes"* (agreement No. 675789).

Special thanks to Professor Michael Scott from Brigham Young University, for granting me his permission to use some of his Figures in Chapter 4.

Chapter 1

Introduction

Accurate and efficient calculation of the pressure distribution of lifting surfaces like wings and propellers has been a popular topic of interest since the beginning of the 20th century in the fields of aerodynamics and hydrodynamics. Lifting forces acting on a wing, which are calculated through pressure integration on its surface, are essential in determining its aeroelastic and acoustic behaviour. Torque and thrust characteristics are calculated in a similar way for a propeller and are of equal importance to its smooth design and operation. At the same time, accurate evaluation of the minimum pressure value near a propeller tip allows correct identification of the region and extend of cavitation.

Early attempts on determining the pressure distribution of two and three dimensional airfoils focused on lifting surface theory and an overview of methods based on this approach can be found for example in [25] and [61]. Numerical implementation of these methods was fast and efficient but lead to reduced accuracy due to the fundamental simplifications that were made, with zero thickness of the lifting surface being the most critical one. In recent times, Computational Fluid Dynamics (CFD) methods and especially the ones solving full-blown Reynolds Averaged Equations (RANS) have become the new standard in many applications. Examples of such methods include [65] for wings and airfoils and [54] for ducted propellers. They offer high accuracy at the cost of increased computational resources. RANS methods are also sensitive to the mesh (or grid) that is used and thus, proper mesh generation is of great importance but it can also be a time-consuming process.

Boundary Element Methods (BEM) have existed since the 60s and strike a balance between Lifting Surface and CFD methods. They use a boundary representation of the lifting surface that can adequately model complex geometries (e.g. propeller blades or wings with geometric discontinuities) but at the same time follow a simplified potential flow approach that is significantly faster to implement numerically than RANS. Hess and Smith [29] in 1964 were the first ones that attempted to numerically solve the boundary value problem of a body of arbitrary geometry in potential flow by introducing the so called "panel" methods. Hess [38] in 1972 extended that method to include lifting forces on a lifting body and later Morino [50] in 1974 introduced a different continuous formulation based on the perturbation potential. In the years that followed Morino's approach became the norm. Consequently, it is evident that boundary element methods for lifting flows have been developed for decades and have reached a level of maturity. There is still interest though in improving their numerical robustness and accuracy, especially in the region of the trailing edge of the wing. As a result, research is still being carried out mainly in the topic of Kutta condition and wake alignment.

In this work, the main aim is to utilise the benefits of the concept of Isogeometric Analysis (IGA) [82] and develop an IGA-based Boundary Element Method for the problem of an external three-dimensional steady uniform flow around a wing that produces satisfactory results with few degrees of freedom. The method is implemented through an in-house solver developed in C++. IGA offers an alternative to traditional panel methods by removing the need for geometry approximation (meshing, panelling etc.) and allowing a smooth representation of the unknown solution field. A novel numerical implementation of Kutta condition is introduced that aims to offer superior accuracy when compared to existing methods. Unstructured, analysis-suitable T-splines are used for the representation of the boundary geometry and their local refineability is used in order to model complex parts of the involved geometries and enhance the performance of Kutta condition by refining the trailing edge curve basis. The solver is also tightly integrated with CAD and that sets the groundwork for future efficient wing shape optimisers.

This thesis is subdivided into 5 sections: Chapter 2 gives a brief review of existing works in the fields of BEM for lifting flows and Isogeometric Analysis with special emphasis on IGA-BEM and T-splines. Chapter 3 describes the formulation as a PDE problem accompanied by boundary conditions. It includes the description of the perturbation potential approach and the process of producing a weak form of Laplace's equation. A brief introduction to analysis suitable T-splines along with the concepts of Bézier extraction and generalised Greville's abscissae are given in Chapter 4. In Chapter 5, the discrete formulation of the problem using isogeometric analysis is presented. The novel numerical implementation of Kutta condition which leads to a non-linear system is also given; along with older versions of it based on panel methods. In Chapter 6, numerical results of the method are presented for three different cases:

1. An unswept wing with a NACA0012 airfoil profile
2. A swept wing with a 20^o sweep angle and a NACA0012 airfoil profile
3. A cambered unswept wing with a NACA4412 airfoil profile

These results are discussed and compared with experiments and a low-order panel method. The behaviour of the IGA-based Kutta condition on the trailing edge is also investigated, in conjunction with the existence of a tip singularity. The last two chapters include conclusions based on the results of this work and suggestions for future work with main considerations on the problem of propeller flow and wake alignment.

Lastly, three appendices are also included in the final pages of this thesis. The first appendix includes an introduction to vortex sheets and proof of some equations that are used in Chapter 3. The second appendix describes the approach for the treatment of nearly singular integrals that occur in the Boundary Integral Equation. Appendix 3 serves as a user manual for the in-house developed IGA-BEM solver.

Chapter 2

Literature Review

1 Boundary Element Methods for Lifting Flows

Formulations based on boundary integral equations (BIEs) for potential lifting flows can be traced back to the first half of the 20th century. According to [49], Prandtl [69] was probably the first who, in 1921, developed his lifting line theory for three dimensional, steady, incompressible flows around wings with high-aspect ratios. Munk [62] in 1924 used a similar approach to analyse the lift characteristics of two-dimensional airfoils in steady, incompressible flows. The series required for analytic solutions of those two methods were introduced by [24] and an inversion formula that associates the vorticity distribution with the flow velocity was given by [76] and [77].

It wasn't though until 1964 and the pioneering work of Hess and Smith [29] that set the groundwork for boundary element methods (BEM) for lifting flows by introducing the so called "panel" methods. They used a BIE produced from Green's theorem for an unknown potential distribution on the body surface. Getting the normal derivative of this BIE with regard to the perturbation potential led to a velocity formulation with a source and dipole distribution on the body surface. The dipole density was set equal to zero and only the source distribution remained to be found. These unknown sources satisfied Laplace's equation and were determined by solving the BIE after applying a Neumann type condition on the body surface. The BIE was discretised by dividing the body surface in quadrilateral panels and satisfying the BIE at the centroid of each panel and the unknown source coefficients were calculated by solving a linear system. Then, the potential of every point of the external flow domain (including the body surface) was calculated through its integral representation. This approach could work on bodies of complex geometries but lifting forces could not be predicted since they are attributed to viscous phenomena.

In order to counter this, Hess [38] improved his original method to calculate lift forces on a lifting body by adding a wake surface leaving the trailing edge of the body and a numerical implementation of the so called *Kutta* condition. The wake was assumed to be a surface of zero thickness that was shed from the trailing edge and extended to infinity. A continuous set of dipoles was distributed on the wake surface in an effort to simulate the trailing vorticity of the wing which is essential for the existence of lift. This approach was based on the principle that a surface distribution of dipoles is equivalent to a distribution of vortices on the boundary curves of that surface. Consequently, the dipole distribution was no longer set equal to zero and the extra unknown dipole coefficients could be determined by the extra number of equations that occurred due to the application of Kutta condition. According to this, the velocities of the upper and lower parts of the body surface should have a common limit on the trailing edge. In the discrete setting, this led to a quadratic system that was solved numerically by applying a Newton-Raphson iterative scheme. Numerical implementation of the method showed good agreement between numerical and experimental results of pressure coefficients with the exception of thin wings and especially in the neighbourhood of the trailing edge.

Later, Morino [50] followed a different approach by attempting to solve the BIE based on the perturbation potential formulation without differentiating it. This led to an unknown potential distribution on the body which could be determined by applying a Neumann boundary condition on the body and wake and a Kutta condition of equal pressure (same as Hess) on the trailing edge. Morino's Kutta condition assumed that the perturbation potential difference between the upper and lower sides of the wake (potential jump) is equal to the corresponding perturbation potential difference between the upper and lower parts of the wing on the trailing edge. In the discrete setting of this approach, the potential jump remained constant along " strips" of chord-wise panels and only the perturbation potential coefficients had to be found which led to simply solving a linear system that was produced after satisfying the BIE on the centroids of the panels. Numerical application of this method showed that it was superior to Hess' for thin wings and it seemed to produce reasonable results near the trailing edge too. Its main downside was that it could not satisfy the zero pressure jump condition on the trailing edge near the tips of the wing. This was due to the fact that crossflow effects happen in that region and Morino's Kutta condition could only work when the velocity had chord-wise components.

Lee [40] in 1987 compared five different boundary integral formulations for the lifting problem:

- Potential-Based Formulations
  - Perturbation Potential Approach
  - Total Potential Approach
- Velocity-Based Formulations
  - Dipole Distribution
  - Vortex Distribution
  - Source Distribution

with the Perturbation Potential being Morino's and the Source Distribution being Hess' approaches respectively. These formulations use different singularity distributions on the body surface but describe identical external flows and yield the same results in the continuous setting. Lee investigated the numerical behaviour of each of these formulations by solving them for a two-dimensional airfoil in uniform flow. He came to the conclusion that the perturbation potential approach by Morino was superior to the rest since it worked well with very thin airfoil sections and potential-based formulations produced one degree less singular kernels compared to the velocity-based ones. Potential formulations also came with some practical benefits like faster calculation of influence coefficients and reduced computer storage requirements since they are based on a scalar field unlike velocity-based formulations (that require thrice as much space). Lee was interested in steady propeller flows so he decided to choose the perturbation potential approach since the difficulty in defining the total velocity potential for rotational flows made the total potential formulation hard to apply on propeller flows.

As mentioned earlier, the weakness of Morino's method was its Kutta condition that didn't satisfy the zero pressure jump on the trailing edge of the wing near the tips. In order to counter this, Lee applied an iterative Kutta condition, the so-called " Iterative Pressure Kutta " (IPK) condition. Details of the formulation of IPK can be found in [35]. As a first iteration step, Morino's problem was solved and the unknown perturbation potential coefficients were acquired. These coefficients were used to calculate a new value for the potential jump values on the trailing edge based on a Newton-Raphson forward scheme. Then the linear system of the discrete form of Morino's BIE was solved again for the new potential jump values. This process was repeated until convergence. The Jacobian of the Newton-Raphson scheme was calculated numerically by applying a finite difference scheme. Application of this method showed good agreement between numerical and experimental results for three-dimensional wings, body and fuselage configurations and propellers in [40]. A further description of this method with an emphasis on the effects of a duct for propeller flows was presented in [35]. A similar iterative Kutta condition was also used by [27] based on an approach introduced by [93]. There has also been a more recent attempt to improve IPK by [79] in 2017. They introduced a method that allows the analytic calculation of the Jacobian components of the Newton-Raphson scheme, in an effort to improve the robustness of the method.

An alternative approach that was based on a distribution of surface vortices was introduced by Belibasakis and Politis in 1995 [42]. This method used a velocity-based formulation and led to a more singular kernel when compared to potential methods. It showed encouraging results and offered some significant advantages when it came to applying a pressure type Kutta condition on the trailing edge. It was later implemented on steady and unsteady propeller problems in [43] and demonstrated good accuracy and efficiency even for complex body shapes.

All works mentioned so far assumed the existence of a known wake surface. For uniform flows of rectangular wings the wake surface had the shape of a simple plane surface parallel to the uniform flow velocity or at an angle included between the dihedral line of the upper and lower parts of the trailing edge. For propeller flows the wake surface had a helicoidal shape consisting of two parts according to [33]:

- A transition region where the contraction and slip stream phenomena happen.
- An ultimate region with helical tip and hub vortices.

The geometry of these regions was fully determined by a set of parameters:

- The ultimate radius of the contracted slipstream.
- the radius of the hub vortex at the end of the transition wake.
- The length of the transition wake region.
- The contraction angle of the tip vortex as it leaves the tip.

The values of these parameters could only be evaluated through experimental observations though, which made this whole approach unreliable for propeller types that had no experimental data. At the same time the effect of tip vortices for both uniform and rotational flows could not be taken into account. This effect would mean the appearance of a "roll-up" of the wake surface in the BEM setting. While the exact pressure distribution on the wake is of no importance, the tip vortex behaviour has an impact on the calculated pressure distribution of lifting body which can be significant, especially in rotational flows.

A major challenge with predicting the geometry of the trailing wake was the fact that the wake, being a material surface [64], moves with the velocity of the fluid at any given point. However this velocity is not a priori known and requires solution of the boundary value problem before it can be evaluated. In an attempt to treat this issue, many researchers in the past developed iterative schemes that would help determine the exact geometry of the wake. The first attempt was by Suciu and Morino [14] in 1976 who determined the wake geometry by enforcing the condition that the induced velocity is tangent to the wake surface and the potential jump remains constant streamwise. Similar attempts were also made by [18] and [28]. Greeley [9] in 1982 developed an iterative process for determining the trailing vortex lines of the wake for a lifting surface method. This approach avoided the direct computation of the induced velocities on the wake in order to avoid excessive computational cost. Instead, he prescribed a radius function for each vortex line, calculated the velocities accurately only on a small set of points and then used an interpolation scheme to compute the velocities at every point on the wake.

Pyo [74] in 1995 developed a panel method with a new wake alignment scheme based on a higher order panel representation of the wake surface. First, the BVP was solved by using a known wake, based on Lee's previous work. Then, Pyo assumed that the velocity of a point on the trailing wake is induced by the dipole distribution of the wake only (and thus, ignored the effect of the singularity distribution of the wing). The induced velocity at each wake point was evaluated by differentiating the wake integral of the BIE and the calculation was simplified by assuming constant dipole strength over each wake panel. Then the new locations of the wake panels were determined by forcing them to be tangent to the induced velocity vector. The BVP was solved again with the new wake geometry and the process was repeated until convergence. The benefits of this method lied in the fact that higher order wake elements could approximate the wake roll-up region better than the low order ones. The latter showed unstable behaviour due to generated gaps in the geometry of the wake and caused the iterative procedure to diverge in some cases. This method was also coupled with the Flow Adapted Grid (FLAG) approach based on [75]. A FLAG is aligned with the wake geometry around the trailing edge and can be used to detect the point on the from where the tip vortices are shed. This approach provided the existing panel methods with some interesting benefits like improved convergence rates for the Iterative Pressure Kutta condition and improved accuracy for pressure coefficients near the tips.

Pyo's wake alignment method was capable of simulating the roll-up effect of the wake geometry but its dependency on higher order wake panels made it require a significantly higher computational cost compared to previous low order panel methods. The development of an efficient alignment scheme has since then, been an active topic of interest for the scientific community and notable works include Lee and Kinnas 2004 [26], Politis 2004 [21] and 2016 [23], Gaggero and Brizzolara 2007 [73], Tian and Kinnas 2012 [92], Kim et al. 2015 [20] and Abdel-Maksoud and Song 2016 [78].

2 Isogeometric Analysis and T-splines

In the meantime, the introduction of *Isogeometric Analysis* (*IGA*) [82] by T.J. Hughes in 2005 offered an alternative approach to traditional problems in numerical methods like Finite and Boundary Element Methods (FEM and BEM). The IGA concept is based on using the same basis for both the geometric model and the unknown solution field. Moreover, the domain discretisation is produced from the underlying geometric representation. This has the benefit of eliminating the geometrically inaccurate, time-consuming and error-prone process of meshing the geometrical model. There have also been observed significant benefits of IGA related to the smoothness of the basis functions in FEM like increased accuracy for reduced degrees of freedom and faster convergence. These were demonstrated in [32] and [31].

IGA applications, due to the nature of the method, require a geometric modelling tool that is flexible enough for both modelling and analysis. Non-Uniform Rational B-Splines (NURBS) constitute the de facto standard for representing curves and surfaces in CAD packages and have already been used in many IGA applications. Examples include [88] and [89] in fluid dynamics, [90] and [81] in fluid-structure interaction (FSI), [3] and [52] in vibrations and [30] and [67] in shells and plates. NURBS however, are not ideal for IGA due to their global tensor product structure and come with a set of disadvantages as stated in [57]:

- NURBS modelling requires multi-patch representations in order to model complex geometries that come with the following issues:
    - Gaps and overlaps may appear at the interface between two patches.
    - Continuity can only be C^0 on the interface between two patches. This a problem especially in IGA, where the smoothness of the representation also affects the smoothness of the solution.
- A large number of control points may be superfluous in the sense that they do not offer significant geometric information. This means increased computational cost in IGA applications.
- Refinement techniques create full rows or columns of control points that may not be needed and, as a result, increase computational cost.

Many alternatives have been developed in the last decade that aim to remedy the deficiencies of NURBS. Examples of these include hierarchical splines [8], THB [5], LR-Splines [80] and T-splines [83], [91].

In this paper unstructured, analysis-suitable T-splines are used. T-splines form a generalisation of NURBS and were introduced in 2003 by T.W Sederberg [83]. The linear independence requirements of their basis functions were demonstrated in 2012 by Li et al. [86]. A T-spline surface that satisfies these requirements is called *analysis suitable* and may be used for IGA. T-splines allow the existence of extraordinary vertices and T-junctions which offer some interesting advantages in both geometric modelling and analysis:

- Complex geometries may be modelled by a single T-spline patch
- Local refinement is possible
- Fewer superfluous points are required compared to NURBS
- Multiple NURBS may be merged into a single T-spline patch

A local refinement algorithm for analysis suitable T-splines was developed in 2012 by Scott et al. [59]. The issue of extraordinary vertices in IGA with T-splines is presented in [57]. In that work, the process of defining a T-spline basis on an unstructured T-mesh is described along with the definition of *Generalised Greville points* which are necessary for collocation methods. Treatment of the reduced smoothness on the neighbourhood of extraordinary vertices is also discussed by applying a fairing process on the Bézier patches that surround the extraordinary vertex in question and are retrieved through a Bézier decomposition. The Bézier decomposition of a spline surface is called *Bézier extraction* and was first introduced in 2010 by Borden et al. [60] for NURBS and then in 2011 by Scott et al. [56] for T-splines.

Early attempts in coupling BEM with CAD may be found in [11] and [12] where shape optimisation is performed with NURBS representing the geometry and polynomials representing the solution field. IGA-enhanced BEM was introduced in [32] and early applications are included in [6] and [70]. It has become more popular in recent years with works in different fields like acoustics [94], elastostatics [72], cracks [4], heat transfer [63], electromagnetics [71], heat conduction [45], [47] etc.

In the field of hydrodynamics, Belibassakis et al. in 2013 [44] developed an IGA-BEM solver for the ship wave resistance problem. They solved the Neumman-Kelvin problem based on a potential flow formulation by using a NURBS representation for the submerged body geometries. Numerical results showed good agreement with experiments and faster convergence for fully submerged bodies when compared with traditional low order panel methods. Later in 2014, Ginnis et al. [1] solved the same problem by using a single patch T-spline representation for both the geometry and the unknown solution field. The T-spline method produced superior convergence to the reference solution compared to the NURBS one due to the local refineability properties of T-splines. At one of the examples, the same error was achieved for 62.5 % fewer degrees of freedom for the T-spline model. The more efficient T-spline solver was later coupled with a T-spline parametric modeller by [48] in order to perform ship-hull shape optimisation. In that work, bicubic T-splines with extraordinary points were used for the representation of the ship-hull models. The representation had C^2 continuity nearly everywhere with the exception of the neighbourhood of extraordinary vertices where the continuity is G^1. Two different examples were tested; one for the bulbous area of the ship and one for the full hull. Results showed that the method was robust enough and could be used in real life applications in ship building.

IGA-BEM for two dimensional lifting flows was introduced by Politis et al. [7] in 2014. In that work, the exterior Neumann problem was solved for a steady lifting flow around an airfoil. A perturbation potential formulation was followed and a Boundary Integral Equation (BIE) was produced based on Morino's [50] approach. The circulation on the trailing wake was assumed to be equal to the potential jump on the trailing edge of the airfoil. A NURBS representation was used for the geometry of the airfoil and the unknown pertubation potential field. The wake was assumed to be a straight line lying on the bisector of the dihedral line of the trailing edge and extending to infinity. A collocation scheme was applied and the location of the collocation points was determined by the Greville abscissae of the underlying geometric representation. Numerical results for the velocity potential and pressure coefficient distributions of an airfoil showed good accuracy for fewer degrees of freedom compared to traditional low-order panel methods.

The previous NURBS-based IGA-BEM method for the airfoil flow was used later in 2016 by Kostas et al. [41] in order to perform shape optimisation on an airfoil geometry. The existing solver was coupled with a NURBS parametric modeller that generated airfoil geometries based on eight parameters. The optimiser employed a controlled elitist genetic algorithm. Two criteria were used as objective functions:

- Maximum Lift Coefficient
- Minimum Deviation of the Hydrofoil Area from a Reference Area

The optimisation procedure first produced an optimised airfoil based on allowing small parameter deviations from a NACA 4412 profile. The resulting profile showed a 3% improvement on the lift coefficient compared to NACA 4412. A pareto front of all airfoil shapes without parameter restrictions was also produced, based on an average lift coefficient calculated for three different angles of attack.

The next reasonable step would be to develop a similar IGA-BEM method for three-dimensional flows which is the aim of this thesis.

Chapter 3

Formulation of the Problem

1 Perturbation Potential

Let $\mathbf{O}xyz$ be a left-handed coordinate system, with its origin at the centre of a full span wing and the $y-$axis directed upwards; see, Figure 3.1. Now consider a flow with linear velocity $\mathbf{V}_l$ and angular velocity $\mathbf{\Omega}$ of an ideal fluid incident on the wing. The total undisturbed velocity at a point with a position vector $\mathbf{r}$ is $\mathbf{V}_\infty = \mathbf{V}_l + \mathbf{\Omega} \times \mathbf{r}$. For uniform flows (such as the ones investigated in this work) $\mathbf{V}_\infty = \mathbf{V}_l$. The flow, except from a free vortex sheet emanating from the trailing edge of the wing, is considered irrotational. Under these assumptions, the flow can be described by the total potential $\Phi = \phi_\infty + \phi$ comprising the undisturbed flow potential ϕ_∞ and the perturbation potential ϕ, which represents the disturbance of the undisturbed uniform flow due to the presence of the wing. The perturbation potential satisfies Laplace's equation.

$$\nabla^2 \phi(\mathbf{x}) = 0, \quad \mathbf{x} \in \Omega. \tag{3.1}$$

Here Ω denotes the open fluid domain bounded by its boundary $\partial\Omega := S = S_B \cup S_W$, where S_B denotes the boundary surface of the wing and S_W denotes the trailing vortex

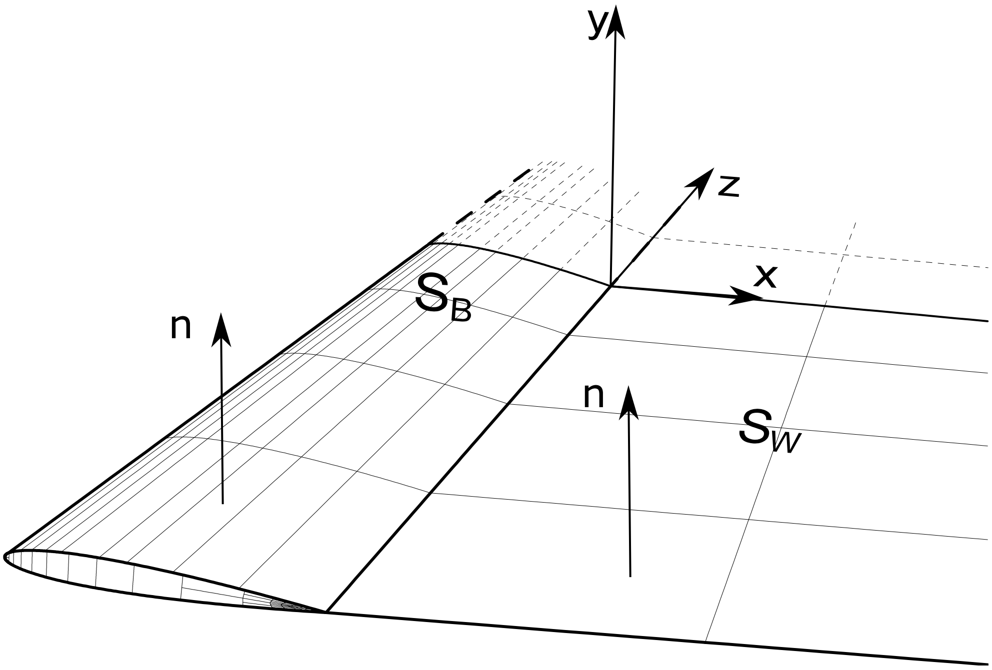


Figure 3.1: Boundary surfaces and coordinate system

wake; see Figure 3.1. The existence of S_W is essential for it enables the potential approach to take into account the circulation that is necessary for the existence of lifting forces on the wing [64]. In general S_W is not known a priori and its final form is part of the solution of the problem. The next step is to provide the boundary conditions that hold true on $\partial\Omega$.

On the wing surface S_B the following non-penetrating boundary condition is applied

$$\frac{\partial \phi}{\partial n}(\mathbf{x}) = -\mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{x}), \quad \mathbf{x} \in S_B, \tag{3.2}$$

where $\mathbf{V}_\infty = \nabla\phi_\infty$ is the undisturbed inflow velocity and $\mathbf{n}(\mathbf{x})$ is the outward-pointing unit normal vector of S_B.

The wake sheet is regarded as a regular surface of *discontinuity* in Ω, where tangential velocity components exhibit discontinuity, although the same is not true for the normal velocity components. According to [51], the boundary conditions for the wake surface can be obtained by applying the mass and momentum conservation equations. Mass conservation yields:

$$\Delta \frac{\partial \phi}{\partial n} \equiv \frac{\partial \phi_u}{\partial n} - \frac{\partial \phi_l}{\partial n} = 0 \quad on \quad S_W \tag{3.3}$$

where $\bullet_{l,u}$ denote the lower and upper side of the wake, respectively, and $\mathbf{n}(\mathbf{x})$ is the unit normal of S_W.

Momentum conservation gives:

$$p_u - p_l = 0 \quad on \quad S_W, \tag{3.4}$$

which can be further manipulated, using Bernoulli's equation:

$$\mid \mathbf{V_u} \mid^2 = \mid \mathbf{V_l} \mid^2 \overset{\times \frac{1}{2}}{\Longrightarrow} \frac{1}{2}(\mathbf{V_u} - \mathbf{V_l})\frac{1}{2}(\mathbf{V_u} + \mathbf{V_l}) = 0 \Rightarrow \mathbf{V_d} \cdot \mathbf{V_m} = 0. \tag{3.5}$$

This last result means that $\mathbf{V_d}$ and $\mathbf{V_m}$ should be orthogonal. Condition (3.5) is satisfied via Kutta condition and an appropriate wake surface. In its general form, Kutta condition requires the velocity to remain bounded, $|\nabla \phi| < \infty$, at the *trailing edge* (TE).

Finally, the perturbation velocity must vanish at an infinite distance from the wing,

$$\nabla \phi \rightarrow 0, \quad \text{when} \quad \sqrt{x^2 + y^2 + z^2} \rightarrow \infty. \tag{3.6}$$

2 Weak Form of Laplace's Equation

In this section, a weak form of Laplace's equation is produced. The first step is to apply the method of weighted residuals (as described for example in [2]). If ϕ is an approximate solution of (3.1) then the residual error is:

$$R = \nabla^2 \phi \tag{3.7}$$

where R=0 when ϕ is an exact solution and the differentiation variables of the Laplacian operator ∇^2 correspond to Cartesian coordinates x, y, z. The aim is to find a proper weight function ω that will evenly distribute R over a domain of integration. The choice of this weight function determines whether a Finite Element or Boundary Element method is implemented. Integrating R over the fluid domain Ω yields:

$$\int_\Omega R\omega d\Omega = \int_\Omega \nabla^2 \phi \omega d\Omega = 0 \tag{3.8}$$

In order to solve this integral equation Green's first identity will be applied first, which, in its general form, is:

$$\int_\Omega (f\nabla \cdot \nabla g + \nabla f \cdot \nabla g) d\Omega = \int_S f \frac{\partial g}{\partial n} dS \tag{3.9}$$

Applying this for $f = \omega$ and $g = \phi$ and since $\int_\Omega \omega \nabla^2 \phi d\Omega = 0$ yields:

$$\int_\Omega \nabla\omega \cdot \nabla\phi d\Omega = \int_S \frac{\partial \phi}{\partial n} \omega dS \tag{3.10}$$

Before proceeding with the solution of the integral equation (3.10) one should first find the fundamental solution of the 3D Laplace equation. It is widely known (see for example [22]) that this corresponds to a source located at a point **P** with (ξ, η, ζ) coordinates and is given by:

$$G(\mathbf{P}, \mathbf{Q}) = \frac{1}{4\pi r(\mathbf{P}, \mathbf{Q})} \tag{3.11}$$

where $r(\mathbf{P}, \mathbf{Q})$ is the distance between points **P** and $\mathbf{Q}(x, y, z) \in \Re^3$.

This fundamental solution will be used as the weighting function of the integral equation (3.10) or $\omega = G(\mathbf{P}, \mathbf{Q})$. Applying Green's first identity again on the left hand side of (3.10) for $f = \phi$ and $g = \omega$ gives:

$$-\int_{\Omega} \phi \nabla^2 \omega d\Omega + \int_S \phi \frac{\partial \omega}{\partial n} dS = \int_S \frac{\partial \phi}{\partial n} \omega dS \tag{3.12}$$

At this point, the properties of the fundamental solution ω dictate the behaviour of the domain integral of (3.12) based on the relative position of **P** with respect to the domain Ω (for proof see [66]).

- When **P** lies inside the domain Ω: $\int_{\Omega} \phi \nabla^2 \omega d\Omega = -\phi(\xi, \eta, \zeta)$. In this way a point value replaces the domain integral.
- When **P** lies outside the domain then the integral is zero everywhere.
- When **P** lies on the boundary surface S the domain Ω can be enlarged by a small sphere of radius ϵ. The new boundary is $S' = S_{-\epsilon} \cup S_{\epsilon}$ where $S_{-\epsilon}$ and $S_{-\epsilon}$ are shown in Figure 3.2. **P** is still inside the enlarged domain so (3.12) becomes:

$$\phi(P) + \int_{S'} \phi \frac{\partial \omega}{\partial n} dS = \int_{S'} \frac{\partial \phi}{\partial n} \omega dS \tag{3.13}$$

In order to proceed, four new integrals must be evaluated for very small values of ϵ. The limiting values of these integrals as $\epsilon \to 0$ are:

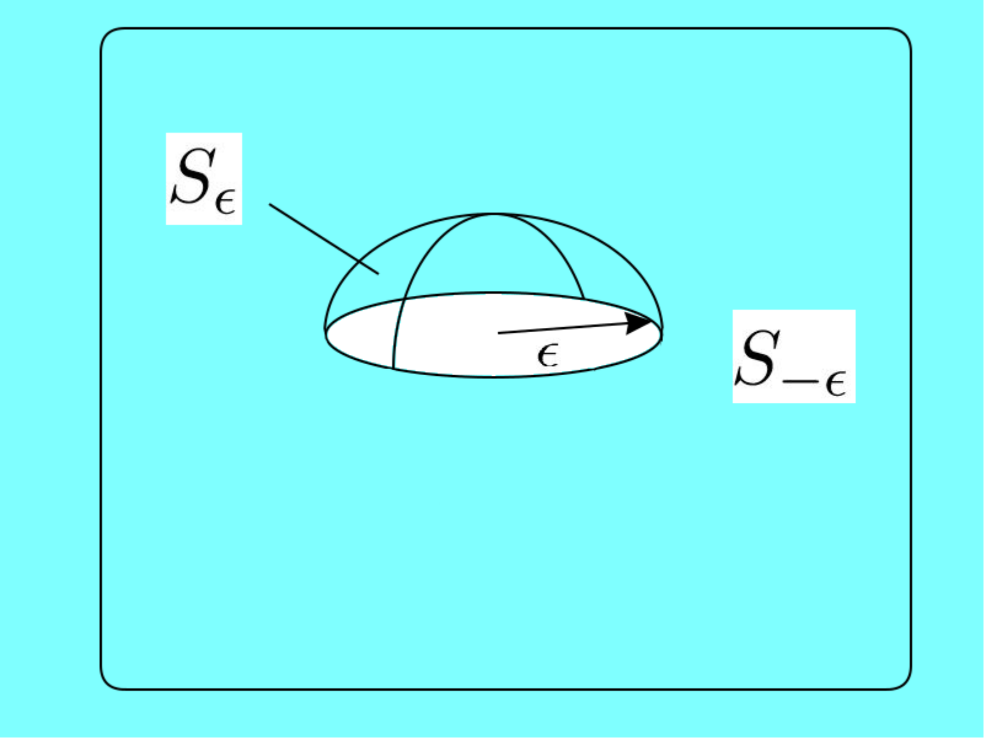


Figure 3.2: Enlarged domain

$$lim_{\epsilon\to 0}\int_{S_\epsilon}\phi\frac{\partial\omega}{\partial n}dS = -\frac{-\phi(\mathbf{P})}{2}$$

$$lim_{\epsilon\to 0}\int_{S_\epsilon}\omega\frac{\partial\phi}{\partial n}dS = 0$$

$$lim_{\epsilon\to 0}\int_{S_{-\epsilon}}\omega\frac{\partial\phi}{\partial n} = \int_S\frac{\partial\phi}{\partial n}$$

$$lim_{\epsilon\to 0}\int_{S_{-\epsilon}}\phi\frac{\partial\omega}{\partial n}\phi dS = \int_S\frac{\partial\omega}{\partial n}\phi dS$$

For further information and proofs on the procedures required to produce and evaluate these integrals one could have a look at [66] or [19].

Combining the above results leads to the final form of the *Boundary Integral Equation* (BIE):

$$c(\mathbf{P})\phi(\mathbf{P}) + \int_S \phi(\mathbf{Q}) \frac{\partial}{\partial n} \left(-\frac{1}{4\pi r(\mathbf{P},\mathbf{Q})} \right) dS = \int_S \frac{\partial \phi(\mathbf{Q})}{\partial n} \left(-\frac{1}{4\pi r(\mathbf{P},\mathbf{Q})} \right) dS \quad (3.14)$$

where

$$c(\mathbf{P}) = \begin{cases} 1 & \mathbf{P} \in \Omega \\ \frac{1}{2} & \mathbf{P} \in S \quad \text{and S smooth at } \mathbf{P} \\ \frac{\alpha}{2\pi} & \mathbf{P} \in S \quad \text{and S not smooth at } \mathbf{P} \\ 0 & \mathbf{P} \quad \textit{outside} \quad \Omega \end{cases} \quad (3.15)$$

and α is the inner solid angle that originates at $\mathbf{P}$ when S is not smooth at $\mathbf{P}$. Furthermore r is the euclidean distance between point $\mathbf{P}$ and boundary point $\mathbf{Q}$ and ϕ corresponds to the potential distribution on the boundary surface.

3 Boundary Integral Equation and Boundary Conditions

The boundary integral equation (3.14) for a point P on the boundary surface S is:

$$2\pi\phi(\mathbf{P}) + \int_S \phi(\mathbf{Q}) \frac{\partial}{\partial n(\mathbf{Q})} \left(-\frac{1}{r(\mathbf{P},\mathbf{Q})} \right) dS(\mathbf{Q}) = \int_S \frac{\partial \phi(\mathbf{Q})}{\partial n(\mathbf{Q})} \left(-\frac{1}{r(\mathbf{P},\mathbf{Q})} \right) dS(\mathbf{Q}) \quad (3.16)$$

It should be noted that in this work $\mathbf{P}$ never lies on edges of the geometry and, thus, $c(\mathbf{P}) = \frac{1}{2}$. Recalling the boundary conditions (3.2) and (3.3) described in Section 1 of this chapter and substituting them in (3.16) yields:

$$2\pi\phi(\mathbf{P}) - \int_{S_B} \phi(\mathbf{Q}) \frac{\partial}{\partial n(\mathbf{Q})} \frac{1}{r(\mathbf{P},\mathbf{Q})} dS(\mathbf{Q}) - \int_{S_W} \Delta\phi(\mathbf{Q}) \frac{\partial}{\partial n(\mathbf{Q})} \frac{1}{r(\mathbf{P},\mathbf{Q})} dS(\mathbf{Q}) \\ = \int_{S_B} V_\infty \cdot n(\mathbf{Q}) \left(\frac{1}{r(\mathbf{P},\mathbf{Q})} \right) dS(\mathbf{Q}) \quad (3.17)$$

where $\Delta\phi(\mathbf{Q}) = \phi_u(\mathbf{Q}) - \phi_l(\mathbf{Q})$ is the so-called potential jump with $\phi_u(\mathbf{Q})$, $\phi_l(\mathbf{Q})$ denoting the values of ϕ on the point $\mathbf{Q}$ for the upper and lower parts of the wake, respectively. It has been proven (see [34] and Appendix A) that the potential jump on the wake varies only along its span-wise direction and its value is determined by the value on the trailing edge of the wing. As a result, equation (3.17) may be written:

$$2\pi\phi(\mathbf{P}) - \int_{S_B} \phi(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) - \int_{S_W} \Delta\phi|_{S_{W,TE}} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\ = \int_{S_B} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q}) \quad (3.18)$$

where $G(\mathbf{P},\mathbf{Q}) = \frac{1}{r(\mathbf{P},\mathbf{Q})}$ and $\Delta\phi|_{S_{W,TE}}$ denotes the restriction of the potential jump on the wake's edge that coincides with the Trailing Edge (TE). The far-field condition of (3.6) is satisfied automatically. The zero-pressure jump condition of (3.4) is satisfied through a zero-pressure jump on the trailing edge (Kutta condition) and an appropriate wake geometry. This condition will be further discussed in Chapters 5 and 6.

Chapter 4

Analysis Suitable T-Splines

1 T-Mesh and Basis Functions

The geometric models of the wings are created using T-splines [1]. This chapter describes the necessary process for the creation of an analysis-suitable bicubic T-spline surface based on [55].

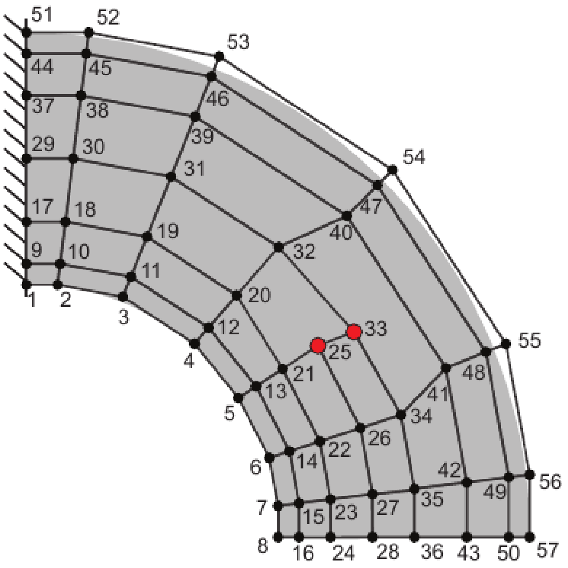
A T-spline surface can be created from a T-mesh. A T-mesh is a quadrilateral mesh which allows the existence of T-junctions and extraordinary vertices. An example of a T-mesh with T-junctions is shown in Figure 4.1. Each number in Figure 4.1 corresponds to a T-mesh vertex to which, a basis function is anchored [2] [91]. The use of anchors helps create an one-to-one correspondence between the basis functions and parametric space entities. Each vertex is assigned to a control point $\mathbf{d}_i \in \Re^{d_s}$ and a weight $w_i \in \Re$ where $1 \leq i \leq A$, A is the total number of global control points and d_s is the dimension of the spline space.

Each quadrilateral element on a T-mesh is called a T-mesh element (shaded in Figure

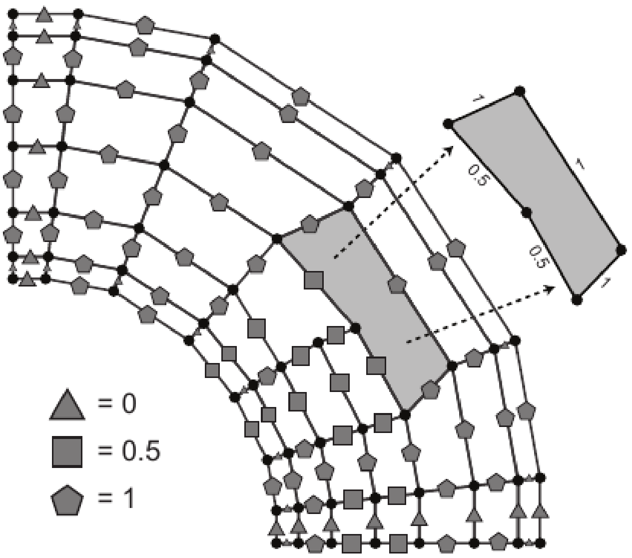
---

[1] One could use for example, Autodesk's Fusion 360 `https://www.autodesk.com/products/fusion-360/`.

[2] This applies only when both degrees are odd. When they are both even basis functions are anchored to faces and when one of them is odd and the other one is even, they are anchored to edges.



(a) T-mesh control points (figure courtesy of Michael Scott [55])



(b) Knot intervals on a T-mesh (figure courtesy of Michael Scott [55])

Figure 4.1: Example of a T-mesh

4.1). In order to properly define a T-spline basis, a proper knot interval configuration must be applied first. Such a configuration can be found in Figure 4.1 where the sum of all knot intervals on one side must be equal to the ones on the opposite side. Now a knot interval vector can be assigned to each vertex $i \in A$ such that $\boldsymbol{\Delta\Xi}_i = \{\Delta\Xi_i^j\}_{j=1}^{d_s}$. Then a local knot vector can be assigned to each control point: $\boldsymbol{\Xi}_i = \{\Xi_i^j\}_{j=1}^{d_s}$. These knot vectors can be formed by assigning an origin value for each parametric direction. In T-spline formulations, the knot interval approach is mostly preferred. Most existing B-spline algorithms are based on a knot vector structure but they can be easily modified to use knot interval vectors too.

Each local knot vector $\boldsymbol{\Delta\Xi}_i$ defines a function domain $\hat{\Omega}_i \subset \Re^{d_s}$:

$$\hat{\Omega}_i = \otimes_{j=1}^{d_s} \hat{\Omega}_i^j \tag{4.1}$$

where $\hat{\Omega}_i^j = [0, \Delta\Xi_{i,1}^j + \Delta\Xi_{i,2}^j + \Delta\Xi_{i,3}^j + \Delta\Xi_{i,4}^j] \subset \Re$ for the bicubic case and each domain has its each own coordinate system $\xi_{\mathbf{i}} = (\xi_i^1, \xi_i^2)$. The procedure for finding $\Delta\Xi_{i,1}^j, \Delta\Xi_{i,2}^j, \Delta\Xi_{i,3}^j, \Delta\Xi_{i,4}^j$ is shown in Figure 4.2.

A T-spline basis function can now be defined through the tensor product of the univariate basis functions $\{N_i^j(\xi_i^j|\Xi_i^j)\}_{j=1}^{d_s}$ where:

$$N_i(\xi_{\mathbf{i}}|\boldsymbol{\Xi}_{\mathbf{i}}) \equiv \prod_{j=1}^{d_s} N_i^j(\xi_i^j|\Xi_i^j) \tag{4.2}$$

and the univariate basis function $N_i^j(\xi_i^j|\Xi_i^j)$ can be evaluated using the Cox-de Boor recursion formula. The resulting basis functions are C^∞ between knots and have reduced smoothness (C^{p-1}, where p is the degree of the basis function) when evaluated exactly on knot values.

At this stage it should be noted that when no T-junctions exist in the T-mesh the resulting T-spline basis corresponds to a B-spline one. A smooth B-spline surface can then be created by assigning a control point $\mathbf{d_i}$ to each basis function:

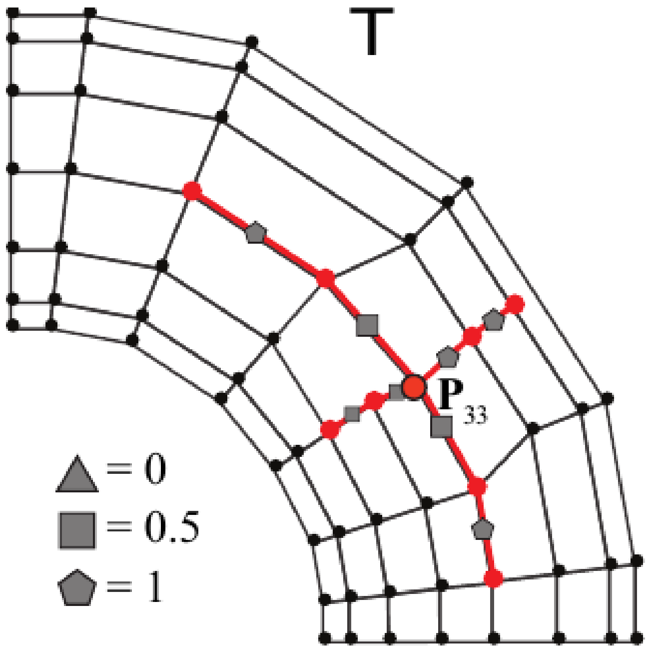


Figure 4.2: Creation of knot interval vector for N_{33}: Starting by d_{33} march to the left until p-1 vertices or perpendicular edges are encountered and add the knot interval values traversed to the knot interval vector. Then march to the right and follow the same steps. This creates the knot interval vector for the first parametric direction (figure courtesy of Michael Scott [55]).

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{i=1}^{n_A} \mathbf{d_i} R_i(\boldsymbol{\xi}) \tag{4.3}$$

$$R_i(\boldsymbol{\xi}) = \frac{w_i N_i(\boldsymbol{\xi})}{\sum_{i=1}^{n_A} w_i N_i(\boldsymbol{\xi})} \tag{4.4}$$

where R_i are the rational B-spline basis functions and $\boldsymbol{\xi} = (\xi^1, \xi^2)$ are parametric values that correspond to a global knot vector created by combining all local knot vectors.

The procedure for making an analysis suitable T-spline surface is presented in the next section.

2 T-Splines for Analysis

Before a T-spline surface can be defined, the concept of T-Spline elements which are necessary for the application of finite/boundary element methods is given.

A T-Spline element Ω^e is a region in the physical space which is bounded by lines of reduced continuity (knot lines). T-spline basis functions have reduced continuity on these lines and C^∞ between them.

For each vertex $\mathbf{d_i}$ the local basis function mesh T_i is given by:

$$T_i = \otimes_{j=1}^{d_s} \Xi_i^j \tag{4.5}$$

An example of a local basis function mesh can be seen in Figure 4.7. The T-mesh can be enhanced by augmenting all the local basis function meshes, as shown in Figure 4.3. Then the **elemental T-mesh** is created. The elemental T-mesh may include lines of reduced continuity which are not included in the T-mesh, as for example shown in Figure 4.3. An example of an elemental T-mesh can be found in Figure 4.4.

When the lines of reduced continuity that originate from T-junctions do not intersect

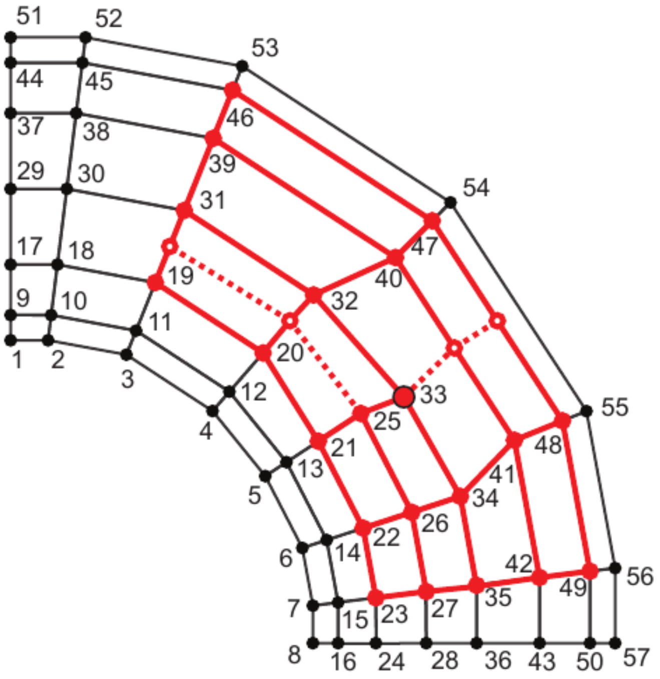


Figure 4.3: T-mesh T (black) enhanced with T_{33} (red) (figure courtesy of Michael Scott [55]).

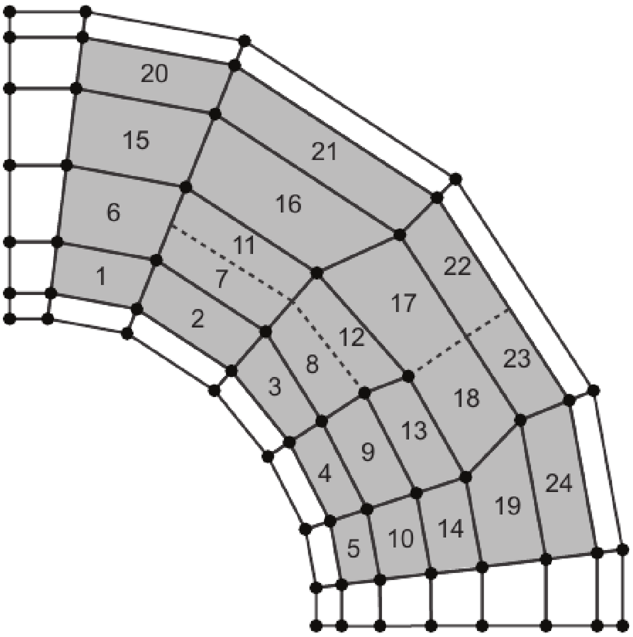


Figure 4.4: Elemental T-mesh with numbering of elements. The elements on the boundary are not numbered because their area is zero (figure courtesy of Michael Scott [55]).

with each other the T-spline surface is called **analysis suitable**. An example for the comparison between analysis suitable and unsuitable element meshes can be seen in Figure 4.5. [87] and [13] proved the existence of many interesting properties of analysis suitable T-splines, such as:

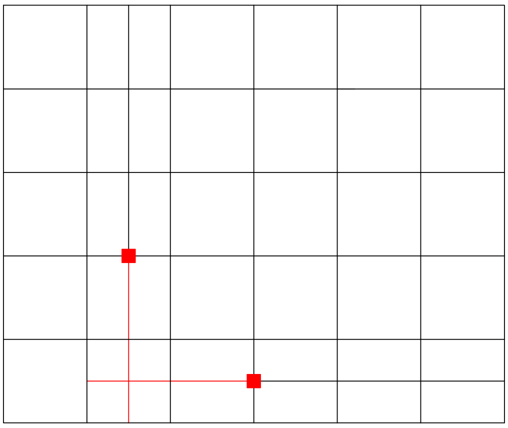
- The basis functions of analysis suitable T-splines are locally linearly independent
- The basis functions of analysis suitable T-splines form a complete basis for the space of polynomials of degree p.

Each T-spline element supports a number of basis functions which are non-zero for this element's domain. In order to determine which of these functions are non-zero the IEN array (Table 4.1) can be used which maps the local basis function α for element e to the global control point index $i \in A$ and, as a result, $\alpha = 1, \dots, n_e \leq n_A$. Each element may support a different number of basis functions while in NURBS each element supports $(p+1)^{d_s}$ functions.

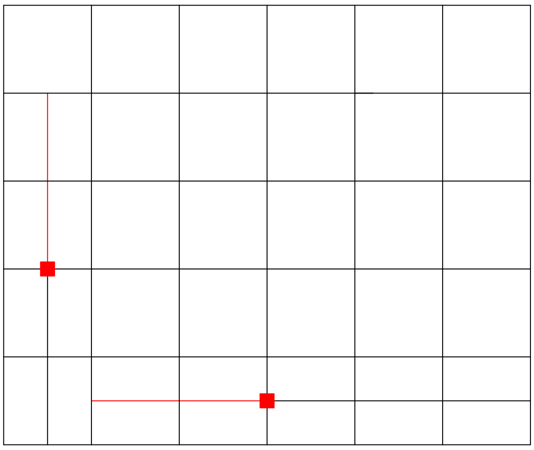
Traditionally, shape functions in FEM/BEM are defined within a parent element domain. Then let $S^e = \{\tilde{\Phi}^e, \{\hat{\Phi}^{n_a}_{\alpha=1}\}$ be a set of affine mappings for element e where

- $\tilde{\Phi}^e : \tilde{\Omega} \to \hat{\Omega}^e$ is an one-to-one and onto affine map from the parent element domain to the element domain:
$$\xi^e = \tilde{\Phi}^e(\tilde{\boldsymbol{\xi}}) \tag{4.6}$$
- $\hat{\Phi}^e_\alpha : \tilde{\Omega}^e \to \hat{\Omega}_i$ for $\alpha = 1, ..., n_a$ is a one-to-one affine mapping from the element domain to the local basis function domain for basis function $i = IEN(\alpha, e)$:
$$\xi_i = \hat{\Phi}^e_\alpha(\boldsymbol{\xi^e}) \tag{4.7}$$

These aforementioned mappings map from the parent element domain to the local basis function domain where the local basis functions are defined (Figure 4.6). Now element-based basis functions can be created which correspond to shape function in FEM/BEM



(a) Analysis unsuitable



(b) Analysis suitable

Figure 4.5: Analysis suitability condition - T junction extensions (red) must not intersect

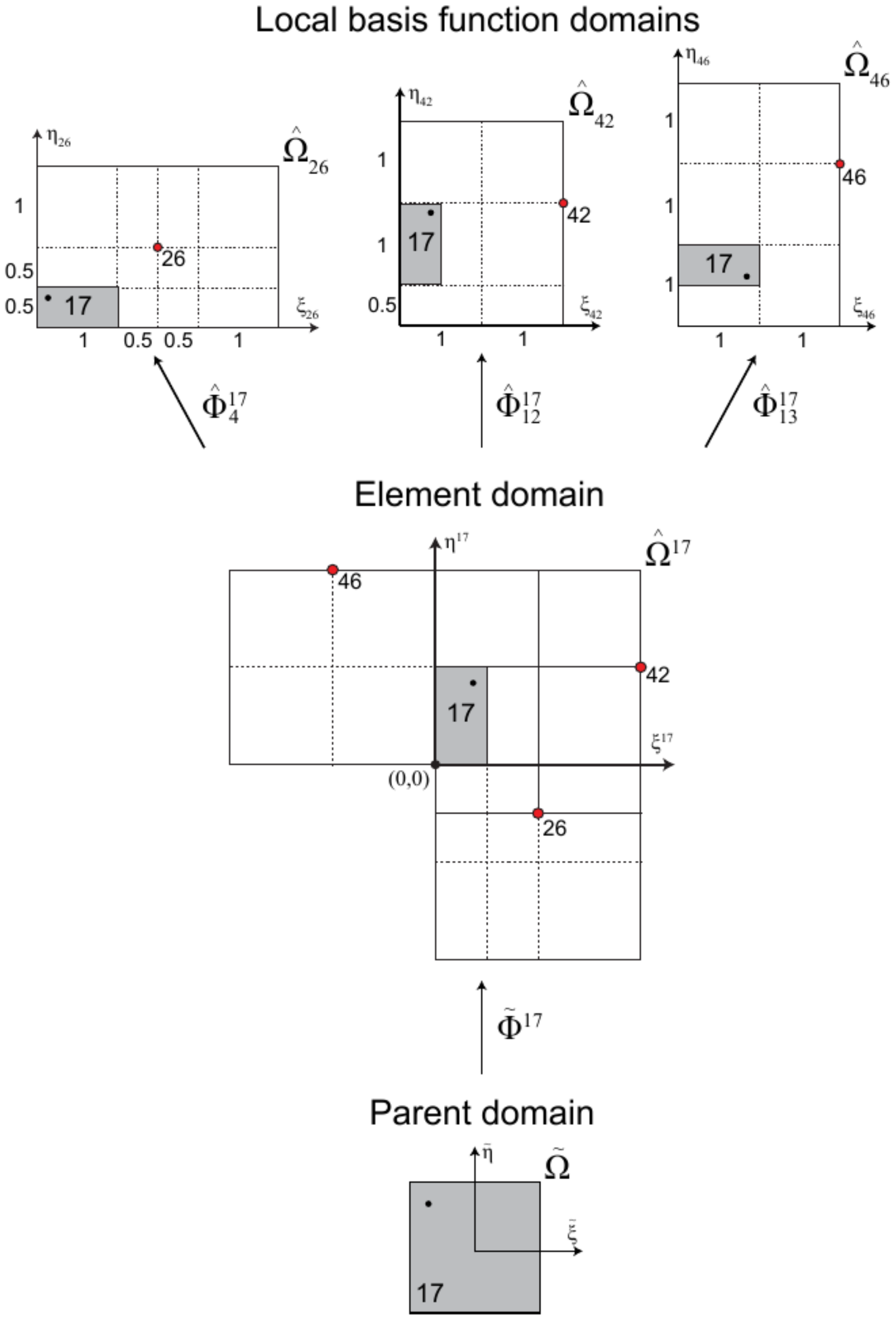


Figure 4.6: Mapping from parent element domain to local basis domain for element 17 (figure courtesy of Michael Scott [55]).

analysis.

$$N_i(\boldsymbol{\xi_i})|_e = N_i(\hat{\Phi}^e_\alpha(\boldsymbol{\xi^e}))|_e = N_i(\hat{\Phi}^e_\alpha(\tilde{\Phi}^e(\tilde{\boldsymbol{\xi}})))|_e = N^e_\alpha(\tilde{\boldsymbol{\xi}}) \tag{4.8}$$

where $|_e$ refers to element e. Now the T-spline geometric mapping from the parent element domain to the physical domain can be produced.

$$\tilde{\mathbf{x}}^{\mathbf{e}}(\tilde{\xi}) = \frac{\sum_{\alpha=1}^{n_\alpha} \mathbf{d}^e_\alpha w^e_\alpha N^e_\alpha(\tilde{\boldsymbol{\xi}})}{\sum_{\alpha=1}^{n_\alpha} w^e_\alpha N^e_\alpha(\tilde{\boldsymbol{\xi}})} = \sum_{\alpha=1}^{n_\alpha} \mathbf{d}^e_\alpha R^e_\alpha(\tilde{\boldsymbol{\xi}}) \tag{4.9}$$

where $\mathbf{d}^e_\alpha = \mathbf{d}_{IEN(\alpha,e)}$ and $w^e_\alpha = w_{IEN(\alpha,e)}$ are the control points and weights respectively that correspond to the $\alpha - th$ basis function of the $e - th$ element and are given by the IEN array and $R^e_\alpha(\tilde{\boldsymbol{\xi}})$ are the rational T-spline basis functions. Equation (4.9) can also be written in matrix form.

$$\tilde{\mathbf{x}}^e(\tilde{\boldsymbol{\xi}}) = \frac{(\mathbf{d}^e)^T \mathbf{W}^e \mathbf{N}^e(\tilde{\boldsymbol{\xi}})}{(\mathbf{w}^e)^T \mathbf{N}^e(\tilde{\boldsymbol{\xi}})} = (\mathbf{d}^e)^T \mathbf{R}^e(\tilde{\boldsymbol{\xi}}) \tag{4.10}$$

where $\mathbf{d}^e$ is a $n_\alpha \times 3$ array which includes the control points for element e, $\mathbf{w}^e$ is weight vector for element e, $\mathbf{W}^e = diag(\mathbf{w}^e)$ and $\mathbf{N}^e = \{N^e_\alpha\}_{\alpha=1}^{n_\alpha}$, $\mathbf{R}^e = \{R^e_\alpha\}_{\alpha=1}^{n_\alpha}$ are vectors of the polynomial and rational T-spline basis functions.

3 Bézier Extraction

The T-splines elements created in the previous section must be further simplified before they can be used in traditional FEM/BEM algorithms. For this purpose a Bézier extraction operator will be applied which maps the T-spline basis to the Bernstein polynomial space. The main idea behind this is to apply knot insertion until the Bézier decomposition of the the T-spline surface is acquired. Then, each T-spline basis function can be written as:

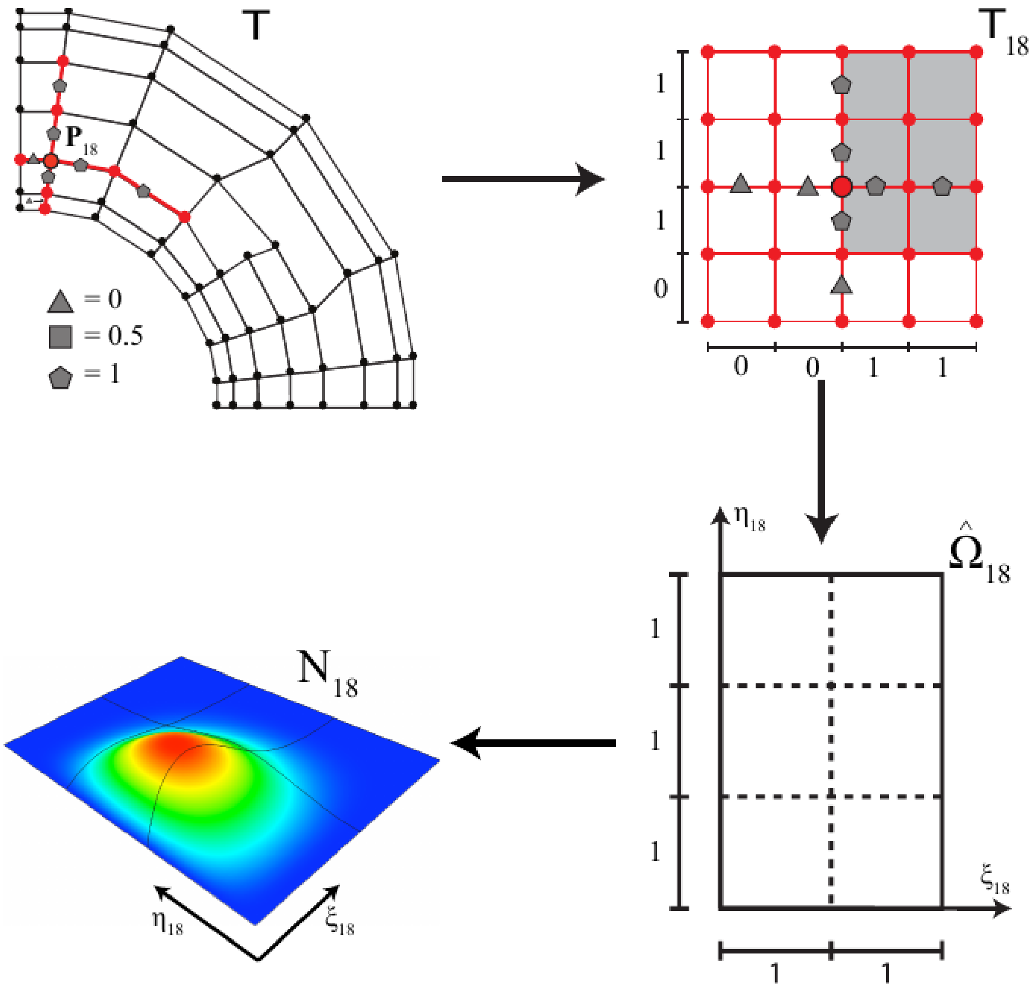


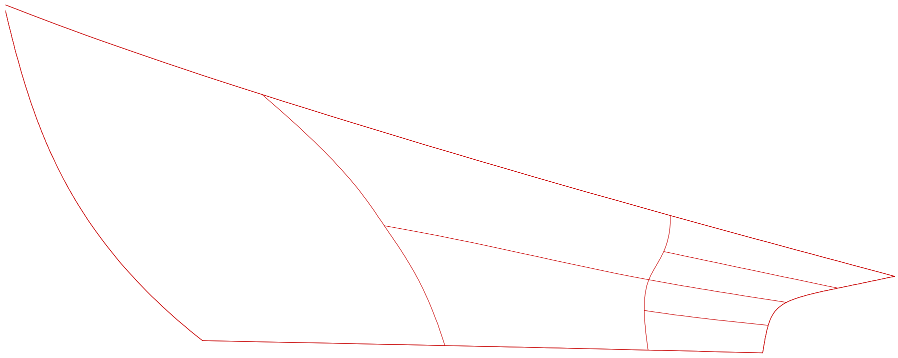
Figure 4.7: From upper left and clockwise: Creation of local knot vector, local basis mesh, local basis domain and basis function in the physical domain (figure courtesy of Michael Scott [55]).

	Element function number (a)																
e	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	9	10	11	12	17	18	19	20	29	30	31	32	
2	2	3	4	5	10	11	12	13	18	19	20	21	25	30	31	32	
3	3	4	5	6	11	12	13	14	19	20	21	22	25	26	31	32	
4	4	5	6	7	12	13	14	15	20	21	22	23	25	26	27	32	
5	5	6	7	8	13	14	15	16	21	22	23	24	25	26	27	28	
6	9	10	11	12	17	18	19	20	29	30	31	32	37	38	39	40	
7	10	11	12	13	18	19	20	21	25	30	31	32	33	38	39	40	
8	11	12	13	14	19	20	21	22	25	26	31	32	33	34	39	40	
9	12	13	14	15	20	21	22	23	25	26	27	32	33	34	35	39	40
10	13	14	15	16	21	22	23	24	25	26	27	28	33	34	35	36	40
11	10	11	12	18	19	20	21	25	30	31	32	33	38	39	40	41	
12	11	12	19	20	21	22	25	26	31	32	33	34	39	40	41	42	
13	12	20	21	22	23	25	26	27	32	33	34	35	39	40	41	42	
14	21	22	23	24	25	26	27	28	33	34	35	36	40	41	42	43	
15	17	18	19	20	29	30	31	32	37	38	39	40	44	45	46	47	
16	18	19	20	25	30	31	32	33	38	39	40	41	45	46	47	48	
17	19	20	25	26	31	32	33	34	39	40	41	42	46	47	48	49	
18	20	25	26	27	32	33	34	35	39	40	41	42	46	47	48	49	
19	25	26	27	28	33	34	35	36	40	41	42	43	47	48	49	50	
20	29	30	31	32	37	38	39	40	44	45	46	47	51	52	53	54	
21	30	31	32	33	38	39	40	41	45	46	47	48	52	53	54	55	
22	31	32	33	34	39	40	41	42	46	47	48	49	53	54	55	56	
23	32	33	34	35	39	40	41	42	46	47	48	49	53	54	55	56	
24	33	34	35	36	40	41	42	43	47	48	49	50	54	55	56	57	

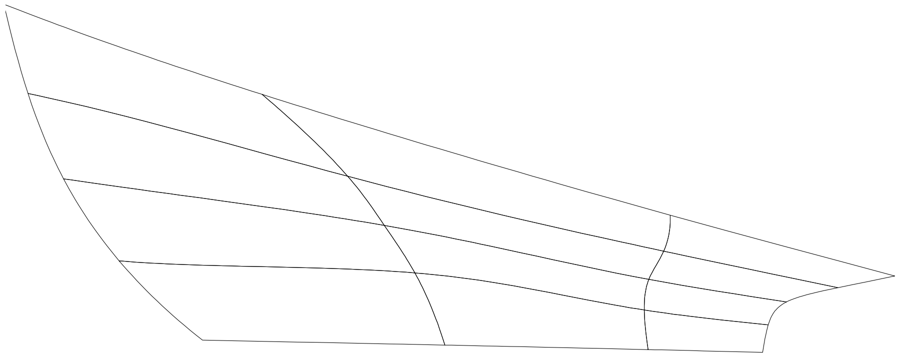
Table 4.1: IEN array for the given T-mesh (table courtesy of Michael Scott [55]).

$$N_\alpha^e(\tilde{\boldsymbol{\xi}}) = \sum_{b=1}^{(p+1)^{d_s}} c_{\alpha,b}^e B_b(\tilde{\boldsymbol{\xi}}) \tag{4.11}$$

where $B_b(\tilde{\boldsymbol{\xi}})$ are Bernstein polynomial functions and $c_{\alpha,b}^e$ are the Bézier extraction coefficients. The method for calculating these coefficients can be found for example in [55]. Equation (4.11) can also be written in matrix form:



(a) T-spline surface



(b) Bézier decomposition

Figure 4.8: Example of Bézier extraction

$$\mathbf{N_e}(\tilde{\boldsymbol{\xi}}) = \mathbf{C_e}\mathbf{B}(\tilde{\boldsymbol{\xi}}) \tag{4.12}$$

where $\mathbf{C_e}$ is the so-called *Bézier extraction operator*. Then the rational T-spline basis functions become:

$$R^e(\tilde{\boldsymbol{\xi}}) = \sum_{\alpha=1}^{n_\alpha} \frac{w_\alpha^e N_\alpha^e(\tilde{\boldsymbol{\xi}})}{\sum_{\alpha_1=1}^{n_\alpha} w_\alpha^e N_\alpha^e(\tilde{\boldsymbol{\xi}})} = \sum_{\alpha=1}^{n_\alpha} \frac{w_\alpha^e \sum_{b=1}^{(p+1)^{d_s}} c_{\alpha,b}^e B_b(\tilde{\boldsymbol{\xi}})}{\sum_{\alpha_1=1}^{n_\alpha} w_\alpha^e \sum_{b=1}^{(p+1)^{d_s}} c_{\alpha,b}^e B_b(\tilde{\boldsymbol{\xi}})} \tag{4.13}$$

and in matrix form

$$\mathbf{R}^e(\tilde{\boldsymbol{\xi}}) = \frac{\mathbf{W}^e \mathbf{C_e} \mathbf{B}(\tilde{\boldsymbol{\xi}})}{(\mathbf{w}^e)^T \mathbf{C_e} \mathbf{B}(\tilde{\boldsymbol{\xi}})} \tag{4.14}$$

where $\mathbf{B}(\tilde{\xi})$ is a set of Bernstein polynomials. Bézier extraction lets all Bézier elements have a constant number (16 in the case of bicubics) of the same basis functions which is compatible with the use of shape functions in FEM/BEM.

4 Unstructured T-mesh

The approach that has been described so far in the previous sections mentioned structured T-meshes only. Unstructured T-meshes are also viable and offer more flexibility in geometric modelling by supporting both T-junctions and extraordinary vertices, i.e., vertices with a valence of 3, 5 and above; see for example the two vertices depicted with red squares in Figure 4.9 and Figure 4.10. All elements touching an extraordinary vertex constitute its one-ring neighbourhood and similarly, the elements neighbouring this one-ring area form the two-ring neighbourhood of the same vertex, and so on.

A valid knot interval configuration must be applied on the unstructured T-mesh before the T-spline bases and surface can be defined. A similar approach is followed as for the structured ones; a non-negative knot interval value is assigned on every edge of the T-mesh while making sure that knot intervals on opposite edges of the same element

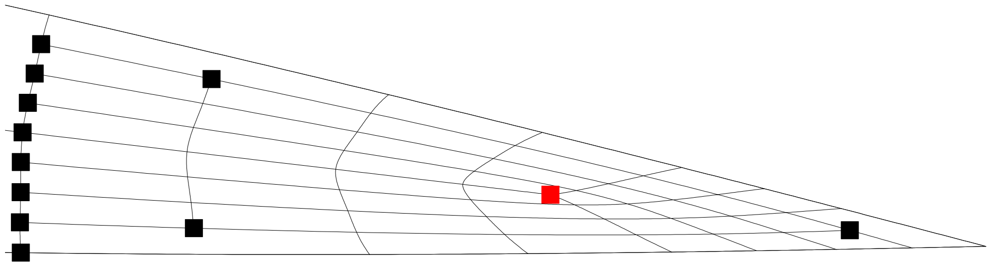


Figure 4.9: Example of a T-mesh part demonstrating T-junctions (black squares) and an extraordinary vertex (red square) of valence 3

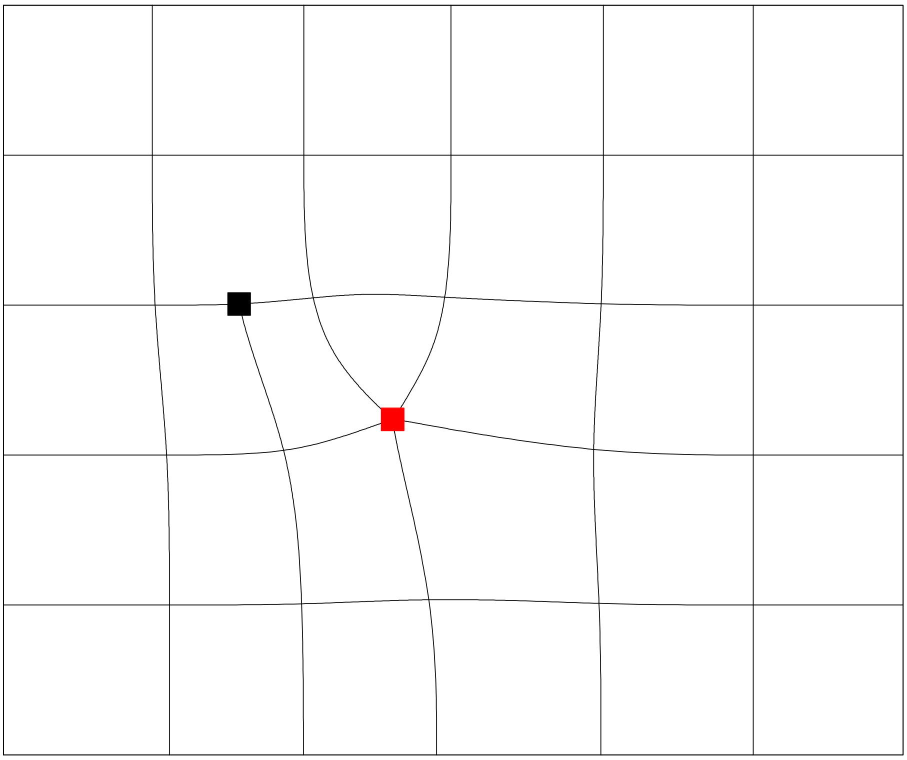


Figure 4.10: Example of a T-mesh part demonstrating a single T-junction (black square) and an extraordinary vertex (red square) of valence 5

sum to the same value. Additionally, it is assumed that the assigned values for the knot intervals along the spoke edges of each extraordinary vertex are all, either zero or non-zero.

T-spline basis functions are linearly independent when the unstructured T-mesh satisfies the following topological requirements, as stated in [57]:

1. T-junction extensions should not intersect with each other.
2. no one-bay[3] face extension spans an element in the three-ring neighbourhood of an extraordinary point.
3. no extraordinary point should lie within the three-ring neighbourhood of another extraordinary point.

This ensures that the resulting T-spline surface is analysis suitable. Then, the unstructured T-spline basis can be defined in the same way as for structured T-meshes.

5 Generalised Greville Points

Greville's abscissae have been a concept of major interest in the field of B-splines and CAD in general. In their original form they refer to knot averages which lie close to the maximum of each B-spline basis function, as described for example in [17]. Although they have traditionally been associated mostly with spline approximations, they have also become popular in IGA-based collocation methods. In these methods, a Boundary Integral Equation is discretely satisfied on a chosen set of points; the so-called collocation points. Greville's abscissae have proven to be a robust choice as collocation points and come with many advantages which may be found for example in [15] and [16].

---

[3] The portion of a face extension lying in the face immediately adjacent to the T-junction is called a one-bay face extension.

In this section the concept of Greville's abscissae is generalised for T-meshes and extraordinary points, as described in [57]. For each vertex, an n-ring collocation point depends on its n-ring knot interval vectors. An n-ring knot interval vector for vertex A, $\mathbf{K}_{A,n} = \{K^i_{A,n}\}^N_{i=1}$ where N is the valence of vertex A and $K^i_{A,n} = \{k^i_{A,1}, k^i_{A,2}, ..., k^i_{A,n}\}$, can be found my marching through each direction until n edges or vertices have been encountered. Then, the collocation point coordinates on the local basis domain are given by:

$$\boldsymbol{\alpha}_{A,n} = \left\{ \begin{matrix} \frac{\sum_{j=1}^{N} -(n-j+1)k^3_{A,j} + \sum_{j=1}^{n}(n-j+1)k^1_{A,j}}{2n+1} \\ \frac{\sum_{j=1}^{N} -(n-j+1)k^4_{A,j} + \sum_{j=1}^{n}(n-j+1)k^2_{A,j}}{2n+1} \end{matrix} \right\} \tag{4.15}$$

For smooth surfaces e.g. ship hulls [1], 1-ring collocation points are adequate. It should be noted that if the T-mesh has no T-junctions or extraordinary points then 1-ring collocation points are equal to the corresponding Greville points. For more complex surfaces e.g. with edges like the wingtips, 2-ring collocation points are used. Local basis coordinates in the local basis domain for a 2-ring collocation point are given by:

$$\boldsymbol{\alpha}_{A,2} = \left\{ \begin{matrix} \frac{2k^1_{A,1} + k^1_{A,2} - 2k^3_{A,1} - k^3_{A,2}}{5} \\ \frac{2k^2_{A,1} + k^2_{A,2} - 2k^4_{A,1} - k^4_{A,2}}{5} \end{matrix} \right\} \tag{4.16}$$

Then the physical coordinates of the collocation point can be found by using the T-spline geometric map from the local basis domain to the physical domain given by (4.10).

Chapter 5

Isogeometric Boundary Element Method

This chapter presents the implementation of the isogeometric approach on the Boundary Integral Equation (3.18). The first section presents the developed IGA-BEM for a single T-spline patch representation of the associated wing-body boundary S_B. It also includes a brief description of older versions of Kutta condition based on low order panel methods and the introduction of a novel IGA-based implementation of it. The second section extends the method for a multi-patch representation of S_B that consists (in the general case) of N_P patches. This is useful for rectangular wings and specifically for $N_P = 2$ and when the wing consists of two surfaces (main wing and cap surfaces - see Chapter 6) that are not smoothly connected.

1 Single-Patch Representation

As a first approach, each of the wing-body and wake boundary surfaces S_B and S_W are represented as single patch bicubic T-spline surfaces which consist of n_{eB} and n_{eW}

T-spline elements respectively:

$$S_B = \cup_{e=1}^{n_{eB}} S_B^e, \quad S_W = \cup_{e=1}^{n_{eW}} S_W^e \tag{5.1}$$

where each of the T-spline elements $S_{[\cdot]}^e$, $[\cdot] \in \{B, W\}$ are given by the general expression which was described in the previous chapter:

$$S_{[\cdot]}^e = \tilde{\mathbf{x}}^e(\tilde{\boldsymbol{\xi}}) = \frac{(\mathbf{d}^e)^T \mathbf{W}^e \mathbf{C_e} \mathbf{B}(\tilde{\boldsymbol{\xi}})}{(\mathbf{w}^e)^T \mathbf{C_e} \mathbf{B}(\tilde{\boldsymbol{\xi}})} = (\mathbf{d}^e)^T \mathbf{R}^e(\tilde{\boldsymbol{\xi}}) \tag{5.2}$$

By application of the Isogeometric Analysis approach, the unknown potential distribution on the wing body may be expressed using the exact same basis functions employed for the geometric representation of its boundary surface:

$$\phi(\mathbf{P}) = \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}), \quad \mathbf{P} \in S_B \tag{5.3}$$

where n_{WB} is the number of control points $\mathbf{d_i}$ of the wing-body surface, ϕ_i is the i-th control coefficient and $\tilde{R}_i(\mathbf{P}) = R^e(\tilde{\boldsymbol{\xi}}(\mathbf{P}))$, $\mathbf{P} \in S_B$.

Now let the spanwise and chordwise coordinates of (3.18) be identical with the parametric directions η, ξ respectively. By inserting (5.3) in (3.18) and assuming that the wing surface consists of n_B Bézier elements:

$$\begin{aligned} 2\pi \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{(B),e}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\ - \int_{S_W} \Delta\phi(\eta) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) = \sum_{e=1}^{n_B} \int_{S_{(B),e}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q}) \end{aligned} \tag{5.4}$$

The value of $\Delta\phi(\eta)$ that satisfies (3.4) on the trailing edge can be found by applying the so-called Kutta condition. Various implementations of Kutta condition are explored in the next subsections.

1.1 Low Order Morino Kutta condition

One of the earliest implementations of Kutta condition can be attributed to Morino. [50]. Following his approach, the potential jump on the trailing edge can be regarded as the difference between the potential values on the upper and lower parts of the wing on the trailing edge:

$$\Delta\phi(\eta) = \phi(\xi_{u,TE}, \eta) - \phi(\xi_{l,TE}, \eta) \tag{5.5}$$

where $\xi_{u,TE}, \xi_{l,TE}$ are the parametric values that correspond to the trailing edge on the upper and lower parts of the wing respectively. In the spirit of low order panel methods, Morino also assumed that the potential jump remains constant along each pair of upper and lower elements of the trailing edge and, thus, it also remains constant along the corresponding strips starting from the TE and extending all along the wake surface. Applying this simplification removes the need for integrating the wing basis functions on the wake. Then, Morino Kutta condition in the IGA setting becomes:

$$\Delta\phi(\eta) = \Delta\phi_{iw} = \sum_{i=1}^{n_{WB}} \phi_i \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \tag{5.6}$$

where iw corresponds to the iw-th strip on the wake starting from the TE and mid is a reference to the parametric value that corresponds to a point which lies on the TE and in the middle of the iw-th element on the TE. Then, (5.4) becomes:

$$
\begin{aligned}
&2\pi \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
&-\sum_{i=1}^{n_{WB}} \phi_i \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) = \\
&\qquad \sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q})
\end{aligned} \tag{5.7}
$$

Equation (5.7) is collocated on n_{WB} points which are chosen to be the Generalised Greville points described in the previous chapter. Then, a linear system is formed:

$$
\begin{aligned}
&2\pi \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}_j) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS \\
&-\sum_{i=1}^{n_{WB}} \phi_i \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS \\
&\qquad = \sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P}_j,\mathbf{Q}) dS(\mathbf{Q})
\end{aligned} \tag{5.8}
$$

which can be also written as:

$$
\sum_{i=1}^{n_{WB}} \phi_i A_{ij} = B_j, \quad j = 1, ..., n_{WB} \tag{5.9}
$$

where

$$
\begin{aligned}
A_{ij} &= 2\pi \tilde{R}_i(\mathbf{P}_j) - \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
&- \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})
\end{aligned} \tag{5.10}
$$

$$B_j = \sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P}_j, \mathbf{Q}) dS(\mathbf{Q}) \tag{5.11}$$

The linear system (5.9) is solved and the values of the unknown coefficients ϕ_i are acquired. The velocity potential can then be calculated on any point on the wing surface from (5.3).

1.2 Non-linear Low Order Kutta condition

It is well known that Morino's Kutta condition, due to its linear nature, neglects three dimensional effects. As a result, it is not capable of ensuring a zero-pressure jump on the trailing edge near the tip of the wing where cross flow phenomena occur. In order to counter this, [75] introduced an iterative Kutta condition that attempted to enforce a zero pressure jump $\Delta p = 0$ on the trailing edge. The challenge that arose in this setting came from the fact that the pressure is non-linearly related to the potential ϕ.

In order to avoid the implications that come with this, an iterative Newton-Raphson scheme was used that allowed the calculation of the total potential jump at the trailing edge, as shown by [74].

The Newton-Rahpson method for non-linear systems of equations $\mathbf{F}(\mathbf{x})$ and a vector of variables $\mathbf{x}$ gives:

$$\mathbf{F}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{F}(\mathbf{x}) + \mathbf{J} \cdot \delta\mathbf{x} \tag{5.12}$$

Where $\mathbf{J}$ is the Jacobian of the system matrix and $\delta\mathbf{x}$ a correction. By setting $\mathbf{F}(\mathbf{x} + \delta\mathbf{x}) = 0$, a set of linear equations is obtained for the correction that moves each function closer to zero:

$$-\mathbf{J}^{-1} \cdot \mathbf{F}(\mathbf{x}) = \delta\mathbf{x} \tag{5.13}$$

Then, the new solution vector becomes:

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \delta\mathbf{x} = \mathbf{x}_{old} - \mathbf{J}^{-1} \cdot \mathbf{F}(\mathbf{x}) \tag{5.14}$$

by applying this on n_{iw} points on the trailing edge a system of n_{imax} non-linear equations (with regard to ϕ) is formed:

$$\Delta p_i = 0, \quad i = 1, ..., n_{imax} \tag{5.15}$$

By applying the Newton-Raphson scheme, a discrete form of Kutta condition was created:

$$\boldsymbol{\Delta\phi}^{(k)} = \boldsymbol{\Delta\phi}^{(k-1)} - \mathbf{J}^{-1}(\boldsymbol{\Delta}\mathbf{p})^{(k-1)} \tag{5.16}$$

where $\boldsymbol{\Delta\phi} = [\Delta\phi_1, ..., \Delta\phi_{n_{imax}}]$, $\boldsymbol{\Delta}\mathbf{p} = [\Delta p_1, ..., \Delta p_{n_{imax}}]$ and k corresponds to k-th iteration. The pressure difference can be calculated by using Bernoulli's law between the upper and lower points on the trailing edge. The Jacobian matrix was defined as $\mathbf{J}_{ij} = \frac{\partial(\Delta p_i)}{\partial(\Delta\phi_j)}$ and the values of the partial derivatives were approximated numerically by employing a finite difference scheme:

$$\frac{\partial(\Delta p_i)}{\partial(\Delta\phi_j)} \approx \frac{\Delta p_i^{(\beta)} - \Delta p_i^{(0)}}{\Delta\phi_j^{(\beta)} - \Delta\phi_j^{(0)}} \tag{5.17}$$

where $\Delta p_i^{(\beta)}$ was calculated by solving the problem again for a slightly perturbed $\Delta\phi_i^{(\beta)} = (1-\beta)\Delta\phi_i^{(0)}$ where $\beta = 0.01$. It's worth noting that the Jacobian was assumed to remain constant at each iteration. For the first iteration a Morino 2D Kutta approximation was used:

$$\boldsymbol{\Delta\phi}^{(0)} = (\boldsymbol{\phi}_u - \boldsymbol{\phi}_l)^{(0)} \tag{5.18}$$

Then for the k-th iteration:

$$\boldsymbol{\Delta\phi}^{(k)} = \boldsymbol{\Delta\phi}^{(0)} - \sum_{m=0}^{k} \mathbf{J}^{-1}(\boldsymbol{\Delta}\mathbf{p})^{(m)} \tag{5.19}$$

Applying this approach in the IGA setting and in BIE (5.4) yields:

$$
\begin{aligned}
&2\pi \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
&-\sum_{i=1}^{n_{WB}} \phi_i \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) = \\
&\sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q}) + \sum_{m=0}^{k} \sum_{iw=1}^{imax} \sum_{jw=1}^{jmax} J_{iw,jw}^{-1} \Delta p_{iw}^{(m)} \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})
\end{aligned}
\tag{5.20}
$$

By assigning n_{WB} collocation points again in BIE (5.7) the linear system is formed:

$$
\begin{aligned}
&2\pi \sum_{i=1}^{n_{WB}} \phi_i \tilde{R}_i(\mathbf{P}_j) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) - \\
&\sum_{i=1}^{n_{WB}} \phi_i \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid}, \eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid}, \eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) = \\
&\sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P}_j,\mathbf{Q}) dS(\mathbf{Q}) + \sum_{m=0}^{k} \sum_{iw=1}^{imax} \sum_{jw=1}^{jmax} J_{iw,jw}^{-1} \Delta p_{iw}^{(m)} \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})
\end{aligned}
\tag{5.21}
$$

which can be also written as:

$$
\sum_{i=1}^{n_{WB}} \phi_i A_{ij} = B_j, \quad j = 1, ..., n_{WB} \tag{5.22}
$$

where

$$A_{ij} = 2\pi \tilde{R}_i(\mathbf{P}_j) - \sum_{e=1}^{n_B} \int_{S_{e,B}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})$$
$$- \sum_{jw=1}^{jmax} \sum_{iw=1}^{imax} \big(\tilde{R}_i(\xi_{u,iw,mid},\eta_{u,iw,mid}) - \tilde{R}_i(\xi_{l,iw,mid},\eta_{l,iw,mid})\big) \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \tag{5.23}$$

$$B_j = \sum_{e=1}^{n_B} \int_{S_{e,B}} V_\infty \cdot n(\mathbf{Q}) G(\mathbf{P_j},\mathbf{Q}) dS(\mathbf{Q})$$
$$+ \sum_{m=0}^{k} \sum_{iw=1}^{imax} \sum_{jw=1}^{jmax} J^{-1}_{iw,jw} \Delta p^{(m)}_{iw} \int_{S_{e,iw,jw}} \frac{\partial G(\mathbf{P_j},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \tag{5.24}$$

The linear system of (5.21) is solved k times and the values of the unknown coefficients ϕ_i are acquired and updated at each step.

1.3 IGA-based Kutta Condition

The implementations of Kutta condition mentioned so far have been adopted from traditional low order panel methods. At this stage a novel IGA-based Kutta condition will be introduced that utilises the benefits of IGA by removing the need for " strip"-wise application and numerical finite difference approximations used in the iterative method.

As mentioned earlier, Kutta condition states that the pressure jump on the trailing edge must be equal to zero. The total velocity on body can be analysed into a tangential component $\mathbf{V}_t$ and a normal component $\mathbf{V}_n$. However, due to the Neumann boundary condition on the body: $\mathbf{V}_n = \mathbf{V}_\infty \cdot \mathbf{n} + \frac{\partial \phi}{\partial \mathbf{n}} = 0$. Then the application of Kutta condition enforces the zero-pressure jump on the trailing edge:

$$\Delta p = p_u - p_l = \frac{1}{2}\rho \mathbf{V}_{t,u}^2 - \frac{1}{2}\rho \mathbf{V}_{t,l}^2 = 0 \tag{5.25}$$

where $\mathbf{V}_{t,u}, \mathbf{V}_{t,l}$ are the upper and lower tangential velocities on the trailing edge respectively. In order to proceed, the tangential pressure $\mathbf{V}_t$ on a point $\mathbf{P}$ lying on an Bézier element $\mathbf{x}_e(\xi, \eta)$ on the wing must be evaluated. In general, the ξ and η parametric directions will not be orthogonal. In order to get the vector of the total tangent velocity, an orthogonal coordinate system with unit vectors $\mathbf{e_1}, \mathbf{e_3}$ on a tangent plane $T(P)$ with a point P on the surface being the origin of this system is needed, as shown in Figure 5.1.

Figure 5.2 demonstrates how this orthogonal system is created and how the total velocity is calculated on $T(P)$. Let $\mathbf{m}_1 = \frac{\partial \mathbf{x_e}(\xi,\eta)}{\partial \xi}$ and $\mathbf{m}_2 = \frac{\partial \mathbf{x_e}(\xi,\eta)}{\partial \eta}$ be tangent vectors parallel to the η and ξ isoparametric curves respectively. Let also $\mathbf{n} = \mathbf{m}_1 \times \mathbf{m}_2$ be a normal vector on $\mathbf{P}$ and $\mathbf{m}_3 = \mathbf{n} \times \mathbf{m}_1$ a vector orthogonal to $\mathbf{m}_1$. The unit vectors $\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3}$ parallel to those tangent vectors are given by:

$$\mathbf{e_1} = \frac{\mathbf{m}_1}{\mid \mathbf{m}_1 \mid}, \quad \mathbf{e_2} = \frac{\mathbf{m}_2}{\mid \mathbf{m}_2 \mid}, \quad \mathbf{e_3} = \frac{\mathbf{m}_3}{\mid \mathbf{m}_3 \mid} \tag{5.26}$$

The projections of the tangential perturbation velocity $\mathbf{v}_t$ on the η and ξ parametric directions are given by:

$$v_{t,\bar{\xi}} = \frac{\partial \phi(\xi,\eta)}{\partial \xi} \cdot \frac{1}{\mid \mathbf{m}_1 \mid}, \quad v_{t,\bar{\eta}} = \frac{\partial \phi(\xi,\eta)}{\partial \eta} \cdot \frac{1}{\mid \mathbf{m}_2 \mid} \tag{5.27}$$

Then the projection $v_{t,\bar{\zeta}}$ on the $\bar{\zeta}$ direction, as shown in Figure 5.2 is given by:

$$v_{t,\bar{\zeta}} = \frac{v_{t,\bar{\eta}} - v_{t,\bar{\xi}} \cdot cos\theta}{sin\theta} \tag{5.28}$$

in a similar way as presented in [57], where $\theta \neq 0$ is the angle between $\mathbf{m}_1$ and $\mathbf{m}_2$ and $cos\theta = \frac{\mathbf{m}_1 \cdot \mathbf{m}_2}{|\mathbf{m}_1| \cdot |\mathbf{m}_2|}$. Now the tangential perturbation velocity vector is given by:

$$\mathbf{v}_t = v_{t,\bar{\xi}} \mathbf{e_1} + v_{t,\bar{\zeta}} \mathbf{e_3} \tag{5.29}$$

Substituting (5.27) and (5.28) in (5.29) gives:

$$\mathbf{v}_t = \frac{\mathbf{e_1}}{\mid \mathbf{m}_1 \mid}\frac{\partial\phi(\xi,\eta)}{\partial\xi} + \frac{\mathbf{e_3}}{\mid \mathbf{m}_2 \mid sin\theta}\frac{\partial\phi(\xi,\eta)}{\partial\eta} - \frac{cos\theta \quad \mathbf{e_3}}{\mid \mathbf{m}_1 \mid sin\theta}\frac{\partial\phi(\xi,\eta)}{\partial\xi} \tag{5.30}$$

Similarly the tangential undisturbed velocity vector is given by:

$$\mathbf{V}_{\infty,t} = (V_\infty \cdot \mathbf{e_1})\mathbf{e_1} + (V_\infty \cdot \mathbf{e_3})\mathbf{e_3} \tag{5.31}$$

Now, recalling (5.3) and (5.25), the pressure difference on the trailing edge can be calculated as follows:

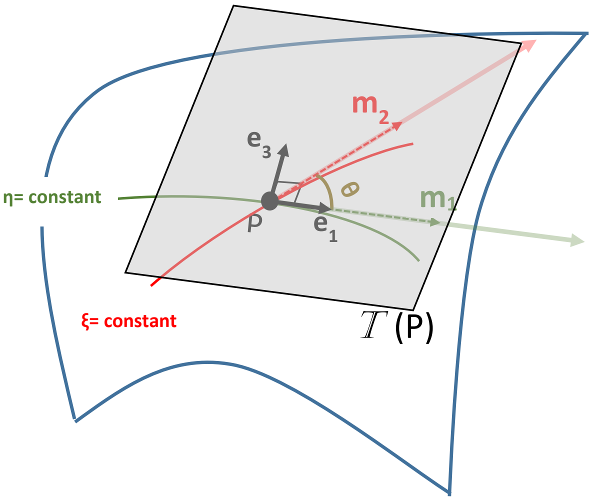


Figure 5.1: Tangent plane T of point P and initial velocity decomposition on non-orthogonal coordinates

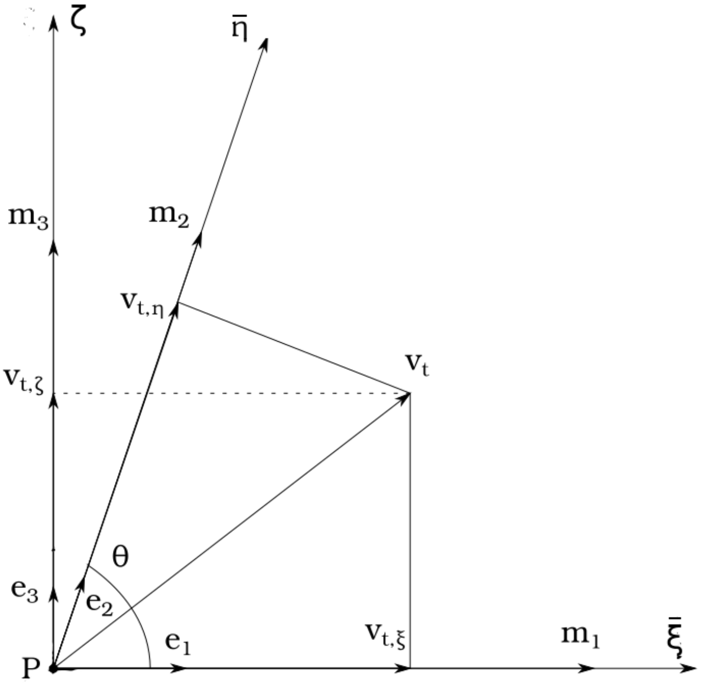


Figure 5.2: Calculation of $\mathbf{V}_t$ on tangent plane $T(P)$

$$\begin{aligned}
&\Delta p = \Delta p(\xi_{TE,u}, \xi_{TE,l}, \eta) = \\
&\frac{1}{2}\rho\Big(\mathbf{V}_{\infty,u,t} + \frac{\mathbf{e}_{1,u}}{\mid \mathbf{m}_{1,u} \mid}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,u},\eta)}{\partial \xi} + \frac{\mathbf{e}_{3,u}}{\mid \mathbf{m}_{2,u} \mid \cdot sin\theta_u}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,u},\eta)}{\partial \eta} \\
&\qquad - \frac{cos\theta_u \mathbf{e}_{3,u}}{\mid \mathbf{m}_{1,u} \mid sin\theta_u}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,u},\eta)}{\partial \xi}\Big)^2 \\
&-\frac{1}{2}\rho\Big(\mathbf{V}_{\infty,l,t} + \frac{\mathbf{e}_{1,l}}{\mid \mathbf{m}_{1,l} \mid}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,l},\eta)}{\partial \xi} + \frac{\mathbf{e}_{3,l}}{\mid \mathbf{m}_{2,l} \mid \cdot sin\theta_l}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,l},\eta)}{\partial \eta} \\
&\qquad - \frac{cos\theta_l \mathbf{e}_{3,l}}{\mid \mathbf{m}_{1,l} \mid sin\theta_l}\sum_{i=1}^{n_{WB}} \phi_i \frac{\partial \tilde{R}_i(\xi_{TE,l},\eta)}{\partial \xi}\Big)^2 \qquad (5.32)
\end{aligned}$$

where, once again, ξ_{TE} corresponds to the parametric values on the trailing edge and u, l correspond to the upper and lower parts of the trailing edge respectively. If this is applied on n_{TE} points on the trailing edge a system of n_{TE} non-linear equations (with regard to ϕ_i) is formed:

$$\Delta p(\xi_{TE,u}, \xi_{TE,l}, \eta_j) = 0, \quad j = 1, ..., n_{TE} \qquad (5.33)$$

or

$$\begin{aligned}
&\mathbf{V}^2_{\infty,t,u,j} - \mathbf{V}^2_{\infty,t,l,j} + \\
&\sum_{i=1}^{n_{WB}} \phi_i \Big(R^j_{\xi,u,i} \Big(\frac{2\mathbf{V}_{\infty,t,u,j}\cdot \mathbf{e}_{1,u,j}}{\mid \mathbf{m}_{1,u,j}\mid} - \frac{2\mathbf{V}_{\infty,t,u,j}\cdot\mathbf{e}_{3,u,j} cos\theta_u}{\mid \mathbf{m}_{1,u,j}\mid sin\theta_u}\Big) + R^j_{\eta,u,i}\frac{2\mathbf{V}_{\infty,t,u,j}\cdot\mathbf{e}_{3,u,j}}{\mid\mathbf{m}_{2,u,j}\mid sin\theta_{u,j}} \\
&\quad + R^j_{\xi,l,i}\Big(\frac{2\mathbf{V}_{\infty,t,l,j}\cdot\mathbf{e}_{3,l,j}cos\theta_l}{\mid\mathbf{m}_{1,l,j}\mid sin\theta_{l,j}} - \frac{2\mathbf{V}_{\infty,t,l,j}\cdot\mathbf{e}_{1,l,j}}{\mid\mathbf{m}_{1,l,j}\mid}\Big) - R^j_{\eta,l,i}\frac{2\mathbf{V}_{\infty,t,l,j}\cdot\mathbf{e}_{3,l,j}}{\mid\mathbf{m}_{2,l,j}\mid sin\theta_{l,j}}\Big) \\
&\quad + \sum_{i=i}^{n_{WB}}\sum_{k=1}^{n_{WB}} \phi_i\phi_k \Big(R^j_{\xi,u,i}R^j_{\xi,u,k}\Big(\frac{1}{\mid\mathbf{m}_{1,u,j}\mid^2} + \frac{cos^2\theta_{u,j}}{\mid\mathbf{m}_{1,u,j}\mid^2 sin^2\theta_{u,j}}\Big) \\
&\quad + R^j_{\eta,u,i}R^j_{\eta,u,k}\frac{1}{\mid\mathbf{m}_{2,u,j}\mid^2 sin^2\theta_{u,j}} - R^j_{\eta,u,i}R^j_{\xi,u,k}\frac{2cos\theta_{u,j}}{\mid\mathbf{m}_{1,u,j}\mid\mid\mathbf{m}_{2,u,j}\mid sin^2\theta_{u,j}} \\
&\quad - R^j_{\xi,l,i}R^j_{\xi,l,k}\Big(\frac{1}{\mid\mathbf{m}_{1,l,j}\mid^2} - \frac{cos^2\theta_{l,j}}{\mid\mathbf{m}_{1,l,j}\mid^2 sin^2\theta_{l,j}}\Big) \\
&\quad - R^j_{\eta,l,i}R^j_{\eta,l,k}\frac{1}{\mid\mathbf{m}_{2,l,j}\mid^2 sin^2\theta_{l,j}} + R^j_{\eta,l,i}R^j_{\xi,l,k}\frac{2cos\theta_{l,j}}{\mid\mathbf{m}_{1,l,j}\mid\mid\mathbf{m}_{2,l,j}\mid sin^2\theta_{l,j}}\Big) = 0, \\
&\qquad\qquad j = 1, ..., n_{TE}. \qquad (5.34)
\end{aligned}$$

where

$$R^j_{\xi,i} = \frac{\partial \tilde{R}_i(\xi_{TE,u},\eta_j)}{\partial\xi}, \quad R^j_{\eta,i} = \frac{\partial \tilde{R}_i(\xi_{TE,u},\eta_j)}{\partial\eta}. \qquad (5.35)$$

Recalling BIE (5.4):

$$\begin{aligned}
&2\pi\sum_{i=1}^{n_{WB}}\phi_i\tilde{R}_i(\mathbf{P}) - \sum_{i=1}^{n_{WB}}\phi_i\sum_{e=1}^{n_B}\int_{S_{B,e}}\tilde{R}_i(\mathbf{Q})\frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})}dS(\mathbf{Q}) \\
&\quad - \int_{S_W}\Delta\phi(\xi_{TE},\eta)\frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})}dS(\mathbf{Q}) = \sum_{e=1}^{n_B}\int_{S_{B,e}}\mathbf{V}_\infty\cdot\mathbf{n}(\mathbf{Q})G(\mathbf{P},\mathbf{Q})dS(\mathbf{Q}) \qquad (5.36)
\end{aligned}$$

The unknown potential jump on the trailing edge can be projected on the spline space defined by the upper part of the trailing edge curve or:

$$\sum_{i=1}^{n_{TE}} \Delta\phi_i R_i(\xi_{TE}, \eta) \tag{5.37}$$

By substituting (5.37) in (5.36) and collocating it at n_{WB} Greville points, n_{WB} equations are formed:

$$\begin{aligned} 2\pi \sum_{i=1}^{n_{WB}} \phi_i R_i(\mathbf{P}_j) - \sum_{i=1}^{n_{WB}} \phi_i \sum_{e=1}^{n_B} \int_{S_{B,e}} R_i(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\ - \sum_{i=1}^{n_{TE}} \Delta\phi_i \int_{\eta_1}^{\eta_2} R_i(\xi_{TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}_j, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) = \\ \sum_{e=1}^{n_B} \int_{S_{B,e}} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_j, \mathbf{Q}) dS(\mathbf{Q}), \quad j = 1, ..., n_{WB}. \end{aligned} \tag{5.38}$$

where ξ_{TE}, ξ_∞ correspond to the parametric values of the trailing edge and a point far away from the wing up to which the wake surface extends. This point is associated with the length of the wake which, in this work, is assumed to be 14 times larger than the chord of the wing.

The discretised BIE and the zero pressure jump on the TE can be expressed as a single system of non-linear equations of the form:

$$\mathcal{P}_j(\Phi) = 0, j = 1, \cdots, n, \; n = n_{WB} + n_{TE}, \quad \Phi = (\phi_1, \cdots, \phi_{n_{WB}}, \Delta\phi_1, \cdots, \Delta\phi_{n_{TE}})^T, \tag{5.39}$$

where the linear part of the system is:

$$\mathcal{P}_j(\Phi) = \sum_{i=1}^{n_{WB}} A_{ij}\phi_i + \sum_{i=1}^{n_{TE}} B_{ij}\Delta\phi_i + c_j, \quad j = 1, \cdots, n_{WB}, \tag{5.40a}$$

$$A_{ij} = 2\pi R_i(\mathbf{P}_j) - \sum_{e=1}^{n_B} \int_{S_{B,e}} R_i(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}), \tag{5.40b}$$

$$B_{ij} = -\int_{\eta_1}^{\eta_2} R_i(\xi_{TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}_j, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}), \tag{5.40c}$$

$$c_j = \sum_{e=1}^{n_B} \int_{S_{B,e}} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_j, \mathbf{Q}) dS(\mathbf{Q}). \tag{5.40d}$$

and the non-linear part:

$$\mathcal{P}_{j+n_{WB}}(\Phi) = \sum_{i=1}^{n_{WB}} D_{ij}\phi_i + \sum_{i=1}^{n_{WB}} \sum_{k=1}^{n_{WB}} E_{ikj}\phi_i\phi_k + g_j, \quad j = 1, \cdots, n_{TE}, \tag{5.41a}$$

$$\begin{aligned} D_{ij} = {} & R^j_{\xi,u,i}\Big(\frac{2\mathbf{V}_{\infty,t,u,j} \cdot \mathbf{e}_{1,u,j}}{\mid \mathbf{m}_{1,u,j} \mid} - \frac{2\mathbf{V}_{\infty,t,u,j} \cdot \mathbf{e}_{3,u,j}}{\mid \mathbf{m}_{1,u,j} \mid sin\theta_{u,j}}\Big) + R^j_{\eta,u,i}\frac{2\mathbf{V}_{\infty,t,u,j} \cdot \mathbf{e}_{3,u,j} cos\theta_{u,j}}{\mid \mathbf{m}_{2,u,j} \mid sin\theta_{u,j}} \\ & + R^j_{\xi,l,i}\Big(\frac{2\mathbf{V}_{\infty,t,l,j} \cdot \mathbf{e}_{3,l,j} cos\theta_{l,j}}{\mid \mathbf{m}_{1,l,j} \mid sin\theta_{l,j}} - \frac{2\mathbf{V}_{\infty,t,l,j} \cdot \mathbf{e}_{1,l,j}}{\mid \mathbf{m}_{1,l,j} \mid}\Big) - R^j_{\eta,l,i}\frac{2\mathbf{V}_{\infty,t,l,j} \cdot \mathbf{e}_{3,l,j}}{\mid \mathbf{m}_{2,l,j} \mid sin\theta_{l,j}}, \end{aligned} \tag{5.41b}$$

$$\begin{aligned} E_{ikj} = {} & R^j_{\xi,u,i}R^j_{\xi,u,k}\Big(\frac{1}{\mid \mathbf{m}_{1,u,j} \mid^2} + \frac{cos^2\theta_{u,j}}{\mid \mathbf{m}_{1,u,j} \mid^2 sin^2\theta_{u,j}}\Big) + R^j_{\eta,u,i}R^j_{\eta,u,k}\frac{1}{\mid \mathbf{m}_{2,u,j} \mid^2 sin^2\theta_{u,j}} \\ & - R^j_{\eta,u,i}R^j_{\xi,u,k}\frac{2cos\theta_{u,j}}{\mid \mathbf{m}_{1,u,j} \mid\mid \mathbf{m}_{2,u,j} \mid sin^2\theta_{u,j}} - R^j_{\xi,l,i}R^j_{\xi,l,k}\Big(\frac{1}{\mid \mathbf{m}_{1,l,j} \mid^2} - \frac{cos^2\theta_{l,j}}{\mid \mathbf{m}_{1,l,j} \mid^2 sin^2\theta_{l,j}}\Big) \\ & - R^j_{\eta,l,i}R^j_{\eta,l,k}\frac{1}{\mid \mathbf{m}_{2,l,j} \mid^2 sin^2\theta_{l,j}} + R^j_{\eta,l,i}R^j_{\xi,l,k}\frac{2cos\theta_{l,j}}{\mid \mathbf{m}_{1,l,j} \mid\mid \mathbf{m}_{2,l,j} \mid sin^2\theta_{l,j}}, \end{aligned} \tag{5.41c}$$

$$g_j = \mathbf{V}^2_{\infty,t,u,j} - \mathbf{V}^2_{\infty,t,l,j}. \tag{5.41d}$$

In order to solve the non-linear system of (5.40a) and (5.41a) an Iterative Newton-Raphson scheme will be applied:

$$\Phi^{r+1} = \Phi^r - J_r^{-1}\mathcal{P}(\Phi^r) \tag{5.42}$$

where r corresponds to the r-th iteration and J_r^{-1} is the inverse of the Jacobian matrix of (5.40a) and (5.41a) of step r. The elements of the Jacobian matrix for each step are given by the following set of equations:

$$J_{j,l} = \frac{\partial \mathcal{P}_j(\Phi)}{\partial \phi_l} = A_{lj}, \quad j = 1, ..., n_{WB}, \quad l = 1, ..., n_{WB} \tag{5.43}$$

$$J_{j,n_{WB}+l} = \frac{\partial \mathcal{P}_j(\Phi)}{\partial \Delta\phi_l} = B_{lj}, \quad j = 1, ..., n_{WB}, \quad l = 1, ..., n_{TE} \tag{5.44}$$

$$J_{j,l} = \frac{\partial \mathcal{P}_j(\Phi)}{\partial \phi_l} = D_{lj} + \sum_{i=1}^{n_{WB}} (E_{lij} + E_{ilj})\phi_i, \quad j = n_{WB}+1, ..., n_{WB}+n_{TE}, \quad l = 1, ..., n_{WB} \tag{5.45}$$

$$J_{j,n_{WB}+l} = \frac{\partial \mathcal{P}_j(\Phi)}{\partial \Delta\phi_l} = 0, \quad j = n_{WB}+1, ..., n_{WB}+n_{TE}, \quad l = 1, ..., n_{TE} \tag{5.46}$$

It's worth mentioning here that the Jacobian matrix is updated at each iteration due to its dependency on ϕ_i.

The Newton-Raphson scheme of (5.42) requires a 0^{th} iteration Φ^0 vector. Following Morino's approach and assuming that the potential jump on the trailing edge of (5.4) is equal to the difference between the upper and the lower potentials gives:

$$\Delta\phi(\xi_{TE}, \eta) = \sum_{i=1}^{n_{WB}} \phi_i^0 \tilde{R}_i(\xi_{u,TE}, \eta) - \sum_{i=1}^{n_{WB}} \phi_i^0 \tilde{R}_i(\xi_{l,TE}, \eta) \tag{5.47}$$

where ϕ_i^0 are the unknown potential coefficients of the 0^{th} iteration. Then (5.4) becomes:

$$2\pi \sum_{i=1}^{n_{WB}} \phi_i^0 \tilde{R}_i(\mathbf{P}) - \sum_{i=1}^{n_{WB}} \phi_i^0 \sum_{e=1}^{n_B} \int_{S_{B,e}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})$$
$$- \sum_{i=1}^{n_{WB}} \phi_i^0 \int_{\eta_1}^{\eta_2} (\tilde{R}_i(\xi_{u,TE},\eta) - \tilde{R}_i(\xi_{l,TE},\eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) =$$
$$\sum_{e=1}^{n_B} \int_{S_{B,e}} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q}) \quad (5.48)$$

And by collocating (5.48) at n_{WB} Greville points the following linear system is formed:

$$2\pi \sum_{i=1}^{n_{WB}} \phi_i^0 \tilde{R}_i(\mathbf{P}_j) - \sum_{i=1}^{n_{WB}} \phi_i^0 \sum_{e=1}^{n_B} \int_{S_{B,e}} \tilde{R}_i(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})$$
$$- \sum_{i=1}^{n_{WB}} \phi_i^0 \int_{\eta_1}^{\eta_2} (\tilde{R}_i(\xi_{u,TE},\eta) - \tilde{R}_i(\xi_{l,TE},\eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}_j,\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q})$$
$$= \sum_{e=1}^{n_B} \int_{S_{B,e}} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_j,\mathbf{Q}) dS(\mathbf{Q}), \quad j = 1, ..., n_{WB}. \quad (5.49)$$

The unknown coefficients ϕ_i^0 are acquired by solving (5.49). The unknown potential jump coefficients of the 0^{th} iteration $\Delta\phi_i^0$ can be acquired by solving an interpolation problem on the trailing edge. Let there be $j = 1, ..., n_{TE}$ points on the trailing edge where the potential jump of the 0^{th} iteration is given by:

$$\Delta\phi(\xi_{TE},\eta_j) = \sum_{i=1}^{n_{WB}} \phi_i^0 (\tilde{R}_i(\xi_{u,TE},\eta_j) - \tilde{R}_i(\xi_{l,TE},\eta_j)) \quad (5.50)$$

Then the unknown $\Delta\phi_i^0$s can be found by solving the following linear system:

$$\sum_{i=1}^{n_{TE}} \Delta\phi_i^0 R_i(\xi_{TE},\eta_j) = \sum_{i=1}^{n_{WB}} \phi_i^0 (\tilde{R}_i(\xi_{u,TE},\eta_j) - \tilde{R}_i(\xi_{l,TE},\eta_j)) \quad j = 1, ..., n_{TE} \quad (5.51)$$

and the 0th iteration solution vector is given by:

$$\Phi^0 = (\phi_1^0, \cdots, \phi_{n_{WB}}^0, \Delta\phi_1^0, \cdots, \Delta\phi_{n_{TE}}^0)^T \tag{5.52}$$

2 Multi-patch Representation

The previous section was based on a single-patch representation of the wing-body boundary surface S_B. In an extension to this, now let the wing geometry consist of N_P T-spline patches. Hence,

$$S_B = \bigcup_{l=1}^{N_P} \bigcup_{e=1}^{ne_l} S_B^{le}, \quad S_W = \bigcup_{l=1}^{1} \bigcup_{e=1}^{qe_l} S_W^{le},$$

with each of the T-spline elements $S_{[\cdot]}^{le}$, $[\cdot] \in \{B, W\}$, given by the general expression:

$$S^{le} = \tilde{\mathbf{x}}_e^l(\tilde{\boldsymbol{\xi}}) = \frac{(\mathbf{d_e}^l)^T \mathbf{W_e}^l \mathbf{C_e}^l \mathbf{B}(\tilde{\boldsymbol{\xi}})}{(\mathbf{w_e}^l)^T \mathbf{C_e}^l \mathbf{B}(\tilde{\boldsymbol{\xi}})} = (\mathbf{d_e}^l)^T \mathbf{R_e}^l(\tilde{\boldsymbol{\xi}}). \tag{5.53}$$

It is also assumed that the trailing edge is included in one of the wing single T-spline patches only. Then the projection of ϕ in the multi-patch T-spline space is given by:

$$\phi(\mathbf{P}) = \sum_{i=1}^{n_{WB,l}} \phi_i^l \tilde{R}_i^l(\mathbf{P}), \quad \mathbf{P} \in S^l, \quad l = 1, ..., N_P \tag{5.54}$$

The projection of $\Delta\phi$ on the spline space of the trailing edge curve is given by:

$$\sum_{i=1}^{n_{TE}} \Delta\phi_i R_i^{l_{TE}}(\xi_{TE}, \eta) \tag{5.55}$$

where the superscript l_{TE} corresponds to the the T-spline patch that includes the trailing edge. By collocating (5.4) at $M = \sum_{l=1}^{N_P} n_{WB,l}$ generalised Greville points for

N_P wing T-spline patches, the same number of linear equations are formed:

$$\begin{aligned}
2\pi \sum_{i=1}^{n_{WB,l}} \phi_i^l R_i^l(\mathbf{P}_m) - \sum_{q=1}^{N_P} \sum_{i=1}^{n_{WB,q}} \phi_i^q \sum_{e=1}^{n_{e,q}} \int_{S_e} R_i^q(\mathbf{Q}) \frac{\partial G(\mathbf{P}_m, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
- \sum_{i=1}^{n_{TE}} \Delta\phi_i \int_{\eta_{l_{TE}}}^{\eta_2} R_i^{l_{TE}}(\xi_{TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}_m, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
= \sum_{q=1}^{N_P} \sum_{e=1}^{n_{e,q}} \int_{S_e} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_m, \mathbf{Q}) dS(\mathbf{Q}), \\
l \in \{1, \ldots, N_P\} : \mathbf{P}_m \in S_B^l, \quad m = 1, \ldots, M. \quad (5.56)
\end{aligned}$$

The application of the IGA-based Kutta condition will remain almost identical to the single patch approach since the trailing edge is part of a single T-spline patch. By assuming again that this patch corresponds to the index $l = l_{TE}$ the zero pressure jump requirement from (5.25) for n_{TE} collocation points on the trailing edge becomes:

$$\begin{aligned}
&\mathbf{V}^2_{\infty,t,up,j} - \mathbf{V}^2_{\infty,t,lo,j} + \\
&\sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE}} \Big(R^{j,l_{TE}}_{\xi,up,i} \Big(\frac{2\mathbf{V}_{\infty,t,up,j} \cdot \mathbf{e}_{1,up,j}}{\mid \mathbf{m}_{1,up,j} \mid} - \frac{2\mathbf{V}_{\infty,t,up,j} \cdot \mathbf{e}_{3,up,j} cos\theta_{up}}{\mid \mathbf{m}_{1,up,j} \mid sin\theta_{up}} \Big) \\
&+ R^{j,l_{TE}}_{\eta,up,i} \frac{2\mathbf{V}_{\infty,t,up,j} \cdot \mathbf{e}_{3,up,j}}{\mid \mathbf{m}_{2,up,j} \mid sin\theta_{up,j}} \\
&+ R^{j,l_{TE}}_{\xi,lo,i} \Big(\frac{2\mathbf{V}_{\infty,t,lo,j} \cdot \mathbf{e}_{3,lo,j} cos\theta_{lo}}{\mid \mathbf{m}_{1,lo,j} \mid sin\theta_{lo,j}} - \frac{2\mathbf{V}_{\infty,t,lo,j} \cdot \mathbf{e}_{1,lo,j}}{\mid \mathbf{m}_{1,lo,j} \mid} \Big) - R^{j,l_{TE}}_{\eta,lo,i} \frac{2\mathbf{V}_{\infty,t,lo,j} \cdot \mathbf{e}_{3,lo,j}}{\mid \mathbf{m}_{2,lo,j} \mid sin\theta_{lo,j}} \Big) \\
&+ \sum_{i=1}^{n_{WB,l_{TE}}} \sum_{k=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE}} \phi_k^{l_{TE}} \Big(R^{j,l_{TE}}_{\xi,up,i} R^{j,l_{TE}}_{\xi,up,k} \Big(\frac{1}{\mid \mathbf{m}_{1,up,j} \mid^2} + \frac{cos^2\theta_{up,j}}{\mid \mathbf{m}_{1,up,j} \mid^2 sin^2\theta_{up,j}} \Big) \\
&+ R^{j,l_{TE}}_{\eta,up,i} R^{j,l_{TE}}_{\eta,up,k} \frac{1}{\mid \mathbf{m}_{2,up,j} \mid^2 sin^2\theta_{up,j}} - R^{j,l_{TE}}_{\eta,up,i} R^{j,l_{TE}}_{\xi,up,k} \frac{2cos\theta_{up,j}}{\mid \mathbf{m}_{1,up,j} \mid\mid \mathbf{m}_{2,up,j} \mid sin^2\theta_{up,j}} \\
&- R^{j,l_{TE}}_{\xi,lo,i} R^{j,l_{TE}}_{\xi,lo,k} \Big(\frac{1}{\mid \mathbf{m}_{1,lo,j} \mid^2} - \frac{cos^2\theta_{lo,j}}{\mid \mathbf{m}_{1,lo,j} \mid^2 sin^2\theta_{lo,j}} \Big) \\
&- R^{j,l_{TE}}_{\eta,lo,i} R^{j,l_{TE}}_{\eta,lo,k} \frac{1}{\mid \mathbf{m}_{2,lo,j} \mid^2 sin^2\theta_{lo,j}} + R^{j,l_{TE}}_{\eta,lo,i} R^{j,l_{TE}}_{\xi,lo,k} \frac{2cos\theta_{lo,j}}{\mid \mathbf{m}_{1,lo,j} \mid\mid \mathbf{m}_{2,lo,j} \mid sin^2\theta_{lo,j}} \Big) = 0, \\
&j = 1, ..., n_{TE}. \quad (5.57)
\end{aligned}$$

where the subscripts up, lo correspond to the upper and lower parts of the trailing edge respectively. Then a new system of $M + n_{TE}$ non-linear equations can be formed:

$$\mathcal{P}_m(\Phi) = 0, \quad m = m(l, j) = \sum_{h=1}^{l-1} n_{WB,h} + j, \tag{5.58}$$

$$j = 1, \dots, n_{WB,l}, \quad l = 1, \dots, N_P + 1, \quad n_{WB,N_P+1} = n_{TE},$$

$$\Phi = (\phi_1^1, \dots, \phi_{n_{WB,1}}^1, \dots, \phi_1^{N_P}, \dots, \phi_{n_{WB,N_P}}^{N_P}, \Delta\phi_1, \dots, \Delta\phi_{n_{TE}})^T \tag{5.59}$$

where $m = M$ when $l = N_P$ and $j = n_{WB,N_P}$. The resulting system consists of a linear and non-linear part. The linear part of the system is:

$$\mathcal{P}_m(\Phi) = \sum_{q=1}^{N_P} \sum_{i=1}^{n_{WB,q}} A_{iqm}\phi_i^q + \sum_{i=1}^{n_{TE}} B_{im}\Delta\phi_i + c_m, \tag{5.60a}$$

$$m = \sum_{h=1}^{l-1} n_{WB,h} + j, \quad j = 1, \dots, n_{WB,l}, \quad l = 1, \dots, N_P,$$

$$A_{iqm} = 2\pi\delta_{lq}R_i^l(\mathbf{P}_m) - \sum_{e=1}^{n_{H,B}} \int_{S_{(H,B),e}} R_i^q(\mathbf{Q})\frac{\partial G(\mathbf{P}_m, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}), \tag{5.60b}$$

$$B_{im} = -\int_{\eta_1}^{\eta_2} R_i^{l_{TE}}(\xi_{TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}_m, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}), \tag{5.60c}$$

$$c_m = \sum_{e=1}^{n_{H,B}} \int_{S_{(H,B),e}} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_m, \mathbf{Q}) dS(\mathbf{Q}). \tag{5.60d}$$

The non-linear part of the system is:

$$\mathcal{P}_m(\Phi) = \sum_{i=1}^{n_{WB,l_{TE}}} D_{ij}\phi_i^{l_{TE}} + \sum_{i=1}^{n_{WB,l_{TE}}}\sum_{k=1}^{n_{WB,l_{TE}}} E_{ikj}\phi_i^{l_{TE}}\phi_k^{l_{TE}} + g_j, \tag{5.61a}$$

$$m = m(j) = \sum_{h=1}^{N_P} n_{WB,h} + j, \quad j = 1, \dots, n_{TE},$$

$$\begin{aligned} D_{ij} &= R^{j,l_{TE}}_{\xi,up,i}\Big(\frac{2\mathbf{V}_{\infty,t,up,j}\cdot\mathbf{e}_{1,up,j}}{\mid \mathbf{m}_{1,up,j}\mid} - \frac{2\mathbf{V}_{\infty,t,up,j}\cdot\mathbf{e}_{3,up,j}}{\mid \mathbf{m}_{1,up,j}\mid sin\theta_{up,j}}\Big) \\ &+R^{j,l_{TE}}_{\eta,up,i}\frac{2\mathbf{V}_{\infty,t,up,j}\cdot\mathbf{e}_{3,up,j}cos\theta_{up,j}}{\mid \mathbf{m}_{2,up,j}\mid sin\theta_{up,j}} \\ &+ R^{j,l_{TE}}_{\xi,lo,i}\Big(\frac{2\mathbf{V}_{\infty,t,lo,j}\cdot\mathbf{e}_{3,lo,j}cos\theta_{lo,j}}{\mid \mathbf{m}_{1,lo,j}\mid sin\theta_{lo,j}} - \frac{2\mathbf{V}_{\infty,t,lo,j}\cdot\mathbf{e}_{1,lo,j}}{\mid \mathbf{m}_{1,lo,j}\mid}\Big) - R^{j,l_{TE}}_{\eta,lo,i}\frac{2\mathbf{V}_{\infty,t,lo,j}\cdot\mathbf{e}_{3,lo,j}}{\mid \mathbf{m}_{2,lo,j}\mid sin\theta_{lo,j}}, \end{aligned} \tag{5.61b}$$

$$\begin{aligned} E_{ikj} &= R^{j,l_{TE}}_{\xi,up,i}R^{j,l_{TE}}_{\xi,up,k}\Big(\frac{1}{\mid \mathbf{m}_{1,up,j}\mid^2} + \frac{cos^2\theta_{up,j}}{\mid \mathbf{m}_{1,up,j}\mid^2 sin^2\theta_{up,j}}\Big)+ \\ &R^{j,l_{TE}}_{\eta,up,i}R^{j,l_{TE}}_{\eta,up,k}\frac{1}{\mid \mathbf{m}_{2,up,j}\mid^2 sin^2\theta_{up,j}} \\ &- R^{j,l_{TE}}_{\eta,up,i}R^{j,l_{TE}}_{\xi,up,k}\frac{2cos\theta_{up,j}}{\mid \mathbf{m}_{1,up,j}\mid\mid \mathbf{m}_{2,up,j}\mid sin^2\theta_{up,j}} - \\ &R^{j,l_{TE}}_{\xi,lo,i}R^{j,l_{TE}}_{\xi,lo,k}\Big(\frac{1}{\mid \mathbf{m}_{1,lo,j}\mid^2} - \frac{cos^2\theta_{lo,j}}{\mid \mathbf{m}_{1,lo,j}\mid^2 sin^2\theta_{lo,j}}\Big) \\ &- R^{j,l_{TE}}_{\eta,lo,i}R^{j,l_{TE}}_{\eta,lo,k}\frac{1}{\mid \mathbf{m}_{2,lo,j}\mid^2 sin^2\theta_{lo,j}} + R^{j,l_{TE}}_{\eta,lo,i}R^{j,l_{TE}}_{\xi,lo,k}\frac{2cos\theta_{lo,j}}{\mid \mathbf{m}_{1,lo,j}\mid\mid \mathbf{m}_{2,lo,j}\mid sin^2\theta_{lo,j}}, \end{aligned} \tag{5.61c}$$

$$g_j = \mathbf{V}^2_{\infty,t,up,j} - \mathbf{V}^2_{\infty,t,lo,j} \tag{5.61d}$$

and δ_{lq} is Kronecker's delta $\delta_{lq} = \begin{cases} 1, & \text{if } l = q, \\ 0, & \text{if } l \neq q. \end{cases}$.

In order to solve the non-linear system of (5.60a) and (5.61a) an iterative Newton-Raphson scheme is applied again:

$$\Phi^{r+1} = \Phi^r - J_r^{-1}\mathcal{P}(\Phi^r) \tag{5.62}$$

where r corresponds to the r-th iteration and J_r^{-1} is the inverse of the Jacobian matrix of (5.60a) and (5.61a) of step r. The elements of the Jacobian matrix $J_{m_1m_2} = \frac{\partial \mathcal{P}_{m_1}(\Phi)}{\partial \phi_{m2}}$ where ϕ_{m2} corresponds to the m_2^{th} component of vector Φ for each iterative step are given by the following set of equations:

$$\begin{aligned}
&J_{m_1m_2} = A_{j_1m_1j_2m_2},\\
&m_1 = m_1(l_1, j_1) = \sum_{h=1}^{l_1-1} n_{WB,h} + j_1, \quad m_2 = m_2(l_2, j_2) = \sum_{h=1}^{l_2-1} n_{WB,h} + j_2,\\
&j_1 = 1, \dots, n_{WB,l_1}, \quad j_2 = 1, \dots, n_{WB,l_2}, \quad l_1 = 1, \dots, N_P, \qquad l_2 = 1, \dots, N_P.
\end{aligned} \tag{5.63}$$

$$\begin{aligned}
&J_{m_1m_2} = B_{j_2m_1},\\
&m_1 = \sum_{h=1}^{l_1-1} n_{WB,h} + j_1, \quad m_2 = \sum_{h=1}^{N_P} n_{WB,h} + j_2,\\
&j_1 = 1, \dots, n_{WB,l_1}, \quad j_2 = 1, \dots, n_{TE}, \quad l_1 = 1, \dots, N_P.
\end{aligned} \tag{5.64}$$

$$\begin{aligned}
&J_{m_1m_2} = D_{j_2j_1} + \sum_{i=1}^{n_{WB,l_{TE}}} (E_{j_2ij_1} + E_{ij_2j_1})\phi_i,\\
&m_1 = \sum_{h=1}^{N_P} n_{WB,h} + j_1, \quad m_2 = \sum_{h=1}^{l_{TE}-1} n_{WB,h} + j_2,\\
&j_1 = 1, \dots, n_{TE}, \quad j_2 = 1, \dots, n_{WB,l_{TE}}.
\end{aligned} \tag{5.65}$$

$$\begin{aligned}
&J_{m_1m_2} = 0,\\
&m_1 = \sum_{h=1}^{N_P} n_{WB,h} + j_1, \quad m_2 = \sum_{h=1}^{l_2-1} n_{WB,h} + j_2,\\
&j_1 = 1, \dots, n_{TE}, \quad j_2 = 1, \dots, n_{WB,l_2} \quad with \quad l_2 \neq l_{TE}.
\end{aligned} \tag{5.66}$$

$$J_{m_1 m_2} = 0,$$
$$m_1 = \sum_{h=1}^{N_P} n_{WB,h} + j_1, \quad m_2 = \sum_{h=1}^{N_P} n_{WB,h} + j_2,$$
$$j = 1, \ldots, n_{TE}, \quad m = 1, \ldots, n_{TE}. \tag{5.67}$$

Morino's approach will be employed again as a 0^{th} iteration for the Newton-Raphson scheme of (5.62). Assuming that the potential jump on the trailing edge is equal to the difference between the perturbation potential values of the upper and lower parts of the trailing edge gives:

$$\Delta\phi(\xi_{TE}, \eta) = \sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} \tilde{R}_i^{l_{TE}}(\xi_{u,TE}, \eta) - \sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} \tilde{R}_i^{l_{TE}}(\xi_{l,TE}, \eta) \tag{5.68}$$

where $\phi_i^{l_{TE},0}$ are the unknown potential coefficients of the 0^{th} iteration and the superscript l_{TE} corresponds to T-spline patch of the wing that includes the trailing edge. Then substituting (5.68) in the multiple patch version of (5.4) gives:

$$\begin{aligned} 2\pi \sum_{i=1}^{n_{WB,l}} \phi_i^{l,0} R_i^l(\mathbf{P}) - \sum_{q=1}^{N_P} \sum_{i=1}^{n_{WB,q}} \phi_i^{q,0} \sum_{e=1}^{n_{e,q}} \int_{S_e} R_i^q(\mathbf{Q}) \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\ - \sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} \int_{\eta_1}^{\eta_2} (\tilde{R}_i^{l_{TE}}(\xi_{u,TE}, \eta) - \tilde{R}_i^{l_{TE}}(\xi_{l,TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P},\mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\ = \sum_{q=1}^{N_P} \sum_{e=1}^{n_{e,q}} \int_{S_e} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P},\mathbf{Q}) dS(\mathbf{Q}), \\ j = 1, ..., n_{WB,l}, \quad l = 1, ..., N_P. \end{aligned} \tag{5.69}$$

By collocating (5.69) at n_{WB} Greville points the following linear system is formed:

$$\begin{aligned}
2\pi \sum_{i=1}^{n_{WB,l}} \phi_i^{l,0} R_i^l(\mathbf{P}_j) - \sum_{q=1}^{N_P} \sum_{i=1}^{n_{WB,q}} \phi_i^{q,0} \sum_{e=1}^{n_{e,q}} \int_{S_e} R_i^q(\mathbf{Q}) \frac{\partial G(\mathbf{P}_j, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
- \sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} \int_{\eta_1}^{\eta_2} (\tilde{R}_i^{l_{TE}}(\xi_{u,TE}, \eta) - \tilde{R}_i^{l_{TE}}(\xi_{l,TE}, \eta) \int_{\xi_{TE}}^{\xi_\infty} \frac{\partial G(\mathbf{P}, \mathbf{Q})}{\partial n(\mathbf{Q})} dS(\mathbf{Q}) \\
= \sum_{q=1}^{N_P} \sum_{e=1}^{n_{e,q}} \int_{S_e} \mathbf{V}_\infty \cdot \mathbf{n}(\mathbf{Q}) G(\mathbf{P}_j, \mathbf{Q}) dS(\mathbf{Q}), \\
j = 1, ..., n_{WB,l}, \quad l = 1, ..., N_P.
\end{aligned} \tag{5.70}$$

The unknown perturbation potential coefficients are acquired by solving (5.70). The unknown potential jump coefficients of the 0th iteration $\Delta\phi_i^0$ can be acquired by solving the same interpolation problem on the trailing edge as described in the previous section. Let there be $j = 1, ..., n_{TE}$ points on the trailing edge where the potential jump of the 0th iteration is given by:

$$\sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} (\tilde{R}_i^{l_{TE}}(\xi_{u,TE}, \eta_j) - \tilde{R}_i^{l_{TE}}(\xi_{l,TE}, \eta_j)) \tag{5.71}$$

Then the unknown $\Delta\phi_i^0$s can be found by solving the following linear system:

$$\sum_{i=1}^{n_{TE}} \Delta\phi_i^0 R_i^{l_{TE}}(\xi_{TE}, \eta_j) = \sum_{i=1}^{n_{WB,l_{TE}}} \phi_i^{l_{TE},0} (\tilde{R}_i^{l_{TE}}(\xi_{u,TE}, \eta_j) - \tilde{R}_i^{l_{TE}}(\xi_{l,TE}, \eta_j)) \quad j = 1, ..., n_{TE} \tag{5.72}$$

Chapter 6

Numerical Results

In this chapter, numerical results of the newly developed IGA-BEM method described in the previous chapter are presented. The method employs a multi-patch representation of the underlying wing geometry and an application of the IGA-based Kutta condition. In the first two sections numerical results for three different test cases are presented and compared against existing experiments and numerical results from a low order panel method. The last section of this chapter focuses on investigating the behaviour of the novel IGA-based Kutta condition with special regard to the singular behaviour of the velocity near the tip.

1 Wings with constant NACA0012 airfoil section

In this section, the developed method is tested for two constant-section wing cases: a rectangular unswept wing and a swept one with a 20 degrees sweep angle. Both wings have a semi-span of length $s = 3m$, a chord of length $c = 1m$ and are generated by translating a NACA0012 profile, represented via the parametrisation described in [46]. A NACA0012 profile is shown in Figure 6.1 and its coordinates are given in table 6.1. The B-spline model of this symmetric airfoil is shown in Figure 6.2 along with its curvature graph and control point structure. The flow is uniform along the x-axis and the angle of attack is equal to 6.75^o. The geometric models of the wings and their respective wakes have been generated using the T-Spline plug-in for Rhinoceros 3D[1]. Views of these models are included in Figure 6.3. Each model consists of the main wing (238 control points), cap (75 control points) and wake (133 control points) surfaces. This leads to a total of 320 DoFs for each example. No T-junctions or extraordinary points exist in the main wing part, and hence, these parts assume a NURBS representation. However, as can be seen by the close-up of the cap surface in Figure 6.4b, T-junctions and extraordinary vertices exist there.

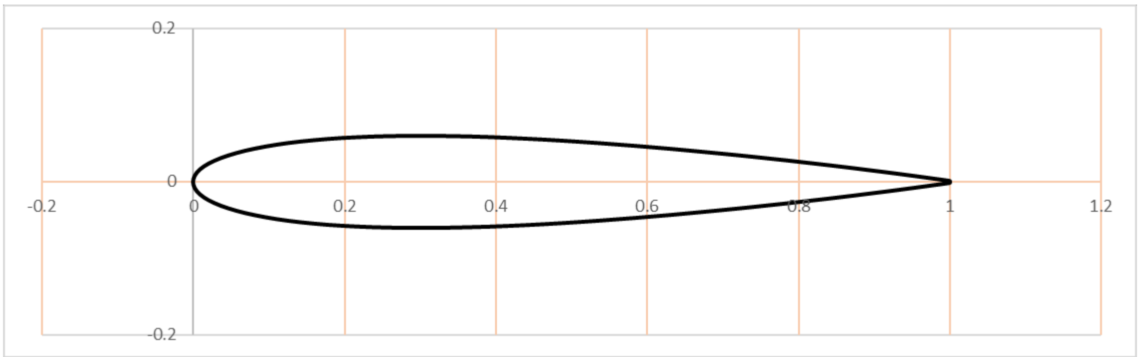


Figure 6.1: NACA0012 Profile

The pressure coefficient distributions on the pressure and suction sides of each wing are shown in Figures 6.5 and 6.6, respectively. It is worth noting that around the tips of both wings and the mid section of the unswept wing three-dimensional effects (or spanwise velocity components) occur. The velocity vectors for these regions are presented in Figure 6.7. Sectional chordwise pressure coefficients are calculated for each wing and compared with experimental results from [53], and simulation results from the open-source low-order panel method xflr5[2]. The corresponding xflr5 model has 3050 DoFs. Seven sections for each wing are selected and shown in Figures 6.8 to 6.11 for the unswept wing, and Figures 6.12 to 6.15 for the swept one. Good agreement is achieved between the IGA-BEM approach and experimental results for much fewer DoFs, when compared to the low-order simulation. At the same time, the IGA-enhanced Kutta

---

[1] http://www.rhino3d.com

[2] http://www.xflr5.tech/xflr5.htm

x	y
1	0
0.9994161	-0.0013419
0.9976658	-0.001587
0.9947532	-0.0019938
0.990685	-0.0025595
0.9854709	-0.0032804
0.9791229	-0.0041519
0.9716559	-0.0051685
0.9630873	-0.0063238
0.9534372	-0.0076108
0.942728	-0.0090217
0.9309849	-0.0105485
0.9182351	-0.0121823
0.9045085	-0.0139143
0.8898372	-0.0157351
0.8742554	-0.0176353
0.8577995	-0.0196051
0.8405079	-0.0216347
0.8224211	-0.0237142
0.8035813	-0.0258337
0.7840324	-0.0279828
0.7638202	-0.0301515
0.7429917	-0.0323294
0.7215958	-0.0345058
0.6996823	-0.03667
0.6773025	-0.0388109
0.6545085	-0.0409174
0.6313537	-0.0429778
0.6078921	-0.0449802
0.5841786	-0.0469124
0.5602683	-0.0487619
0.5362174	-0.0505161
0.5120819	-0.052162
0.4879181	-0.0536866
0.4637826	-0.0550769
0.4397317	-0.05632
0.4158215	-0.0574033
0.3921079	-0.0583145
0.3686463	-0.0590419
0.3454915	-0.0595747
0.3226976	-0.0599028
0.3003177	-0.0600172
0.2784042	-0.0599102
0.2570083	-0.0595755
0.2361799	-0.0590081
0.2159676	-0.0582048
0.1964187	-0.057164
0.1775789	-0.0558856
0.1594921	-0.0543715
0.1422005	-0.0526251
0.1257446	-0.0506513
0.1101628	-0.0484567
0.0954915	-0.0460489
0.0817649	-0.0434371
0.0690152	-0.040631
0.057272	-0.0376414
0.0465628	-0.0344792
0.0369127	-0.0311559
0.0283441	-0.0276827
0.0208771	-0.0240706
0.0145291	-0.02033
0.0093149	-0.0164706
0.0052468	-0.0125011
0.0023342	-0.0084289
0.0005839	-0.0042603
0	0

Table 6.1: Coordinates of a symmetric NACA0012 profile
(source: http://airfoiltools.com/airfoil/details?airfoil=n0012-il)

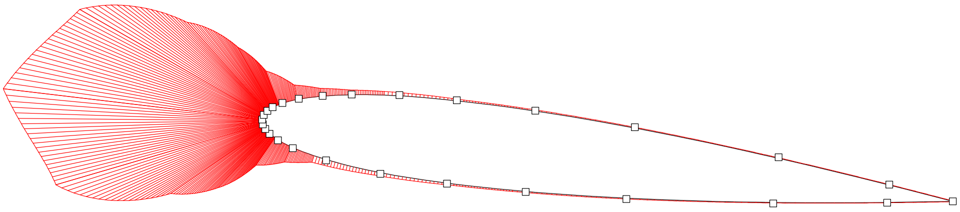
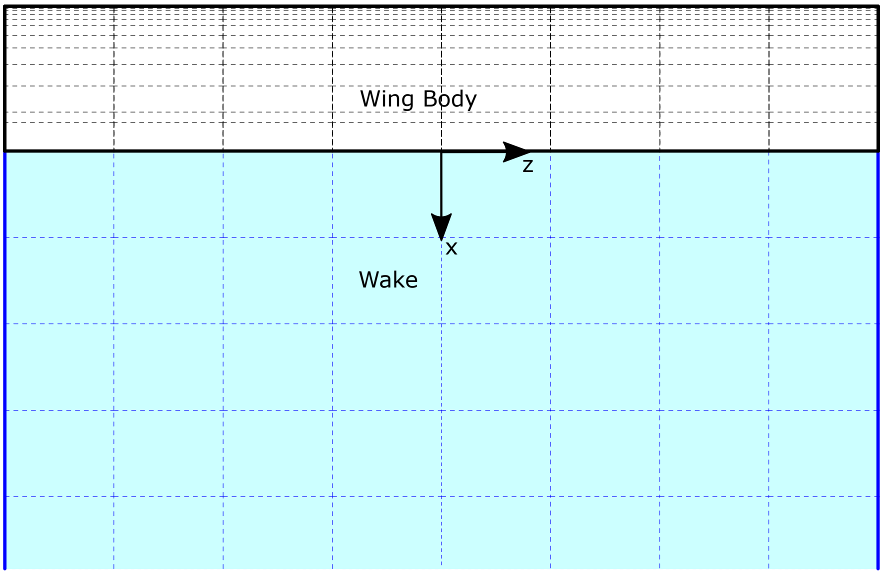


Figure 6.2: NACA0012 airfoil curvature and control point structure

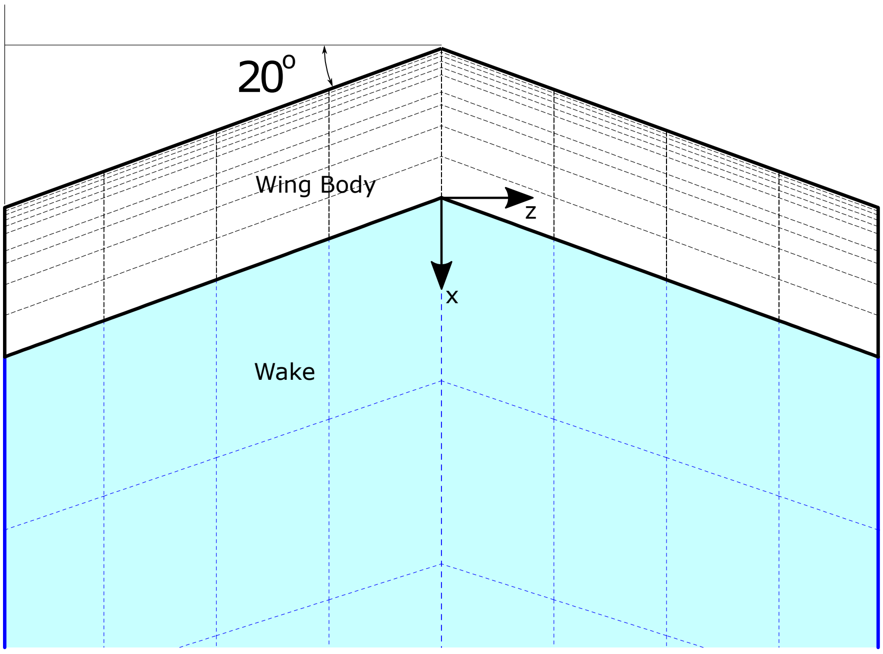
condition shows superior accuracy on the trailing edge in comparison with xflr5. This is illustrated in Figure 6.16 for the mid section of the unswept wing. It is clear that the IGA-BEM method satisfies the zero pressure jump on the trailing edge while xflr5 doesn't. At the same time, xflr5 shows an oscillatory behaviour that deteriorates for higher DoFs.

Deviations between IGA-BEM and experimental results are exhibited only for sections near the wing tip, as illustrated in Figure 6.11. This occurs due to the existence of a wing tip vortex near that region which cannot be modelled in the context of potential flow theory, as stated, for example, in [85]. A further investigation of the pressure coefficient behaviour at the tip is discussed in detail in the following section.

Finally, the effect of refinement on the sectional pressure distribution of the unswept wing is presented in Figure 6.17. It is evident that for the two investigated sections only minor deviations occur for different DoFs- especially the finer ones. An interesting behaviour is encountered when comparing the spanwise circulation, as it is presented in Figure 6.18. The two coarse representations seem to deviate significantly from the

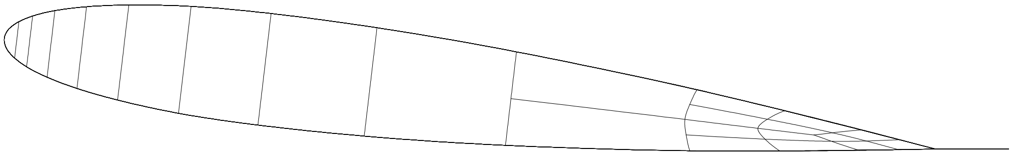


(a) Unswept

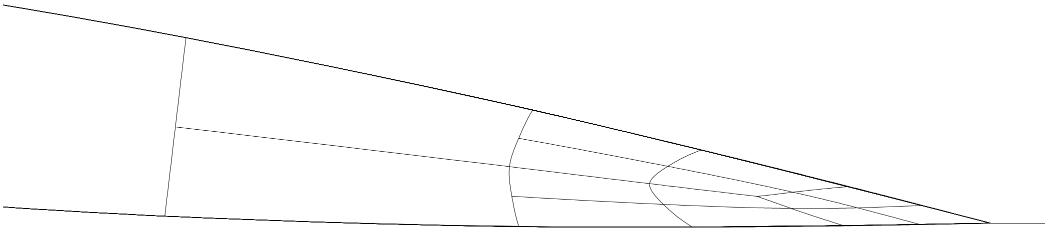


(b) Swept

Figure 6.3: Wing and wake surfaces of tested wings



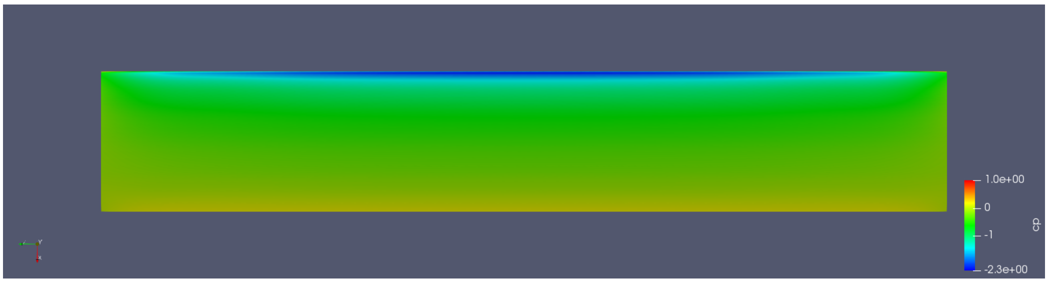
(a) Full Cap



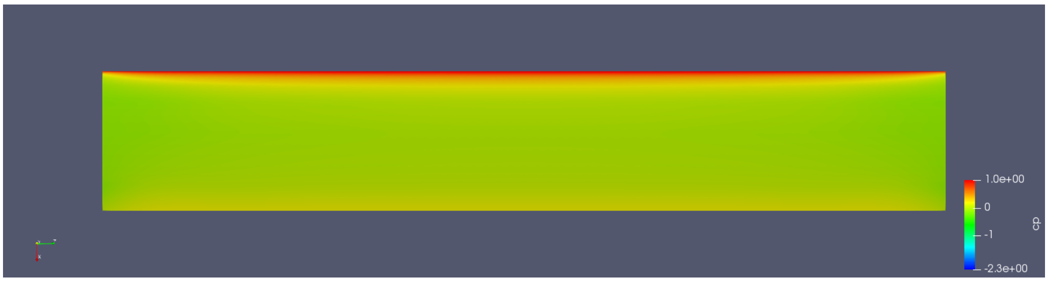
(b) T-junctions and Extraordinary Point of valence 3

Figure 6.4: Wing Cap Surface

two finer ones. At the same time, the two finer representations seem to agree mostly around the mid section area and have minor deviations when approaching the tips. An investigation and explanation of this behaviour is included in the next section.

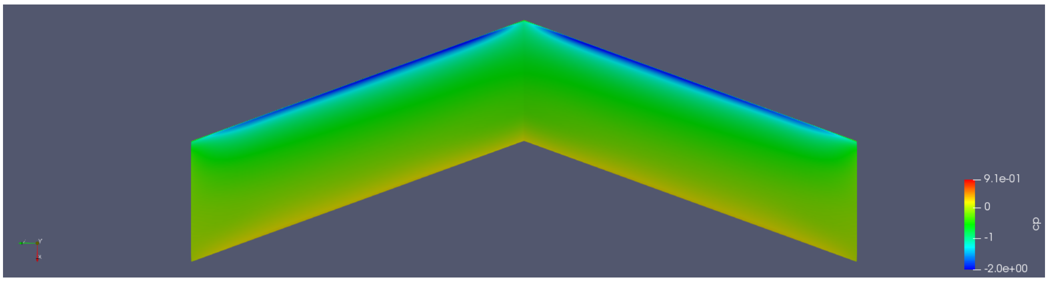


(a) Suction side

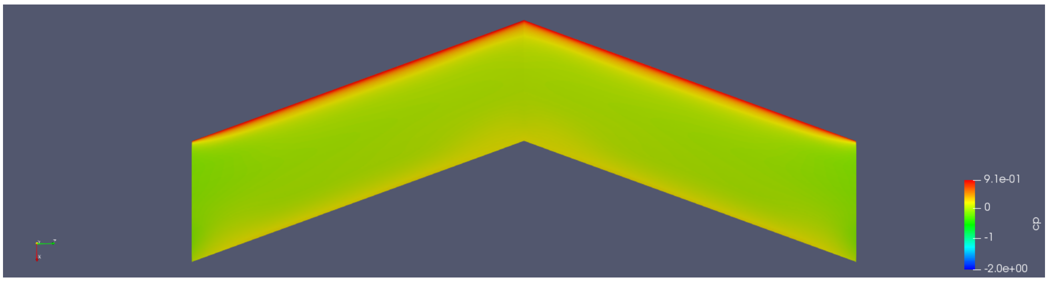


(b) Pressure side

Figure 6.5: Pressure coefficient distribution for an unswept wing

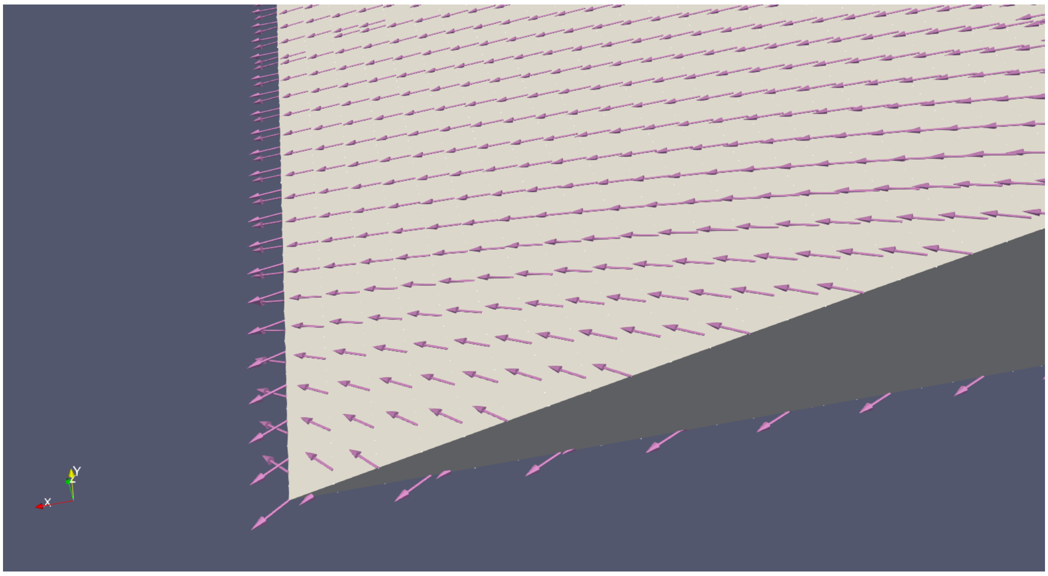


(a) Suction side

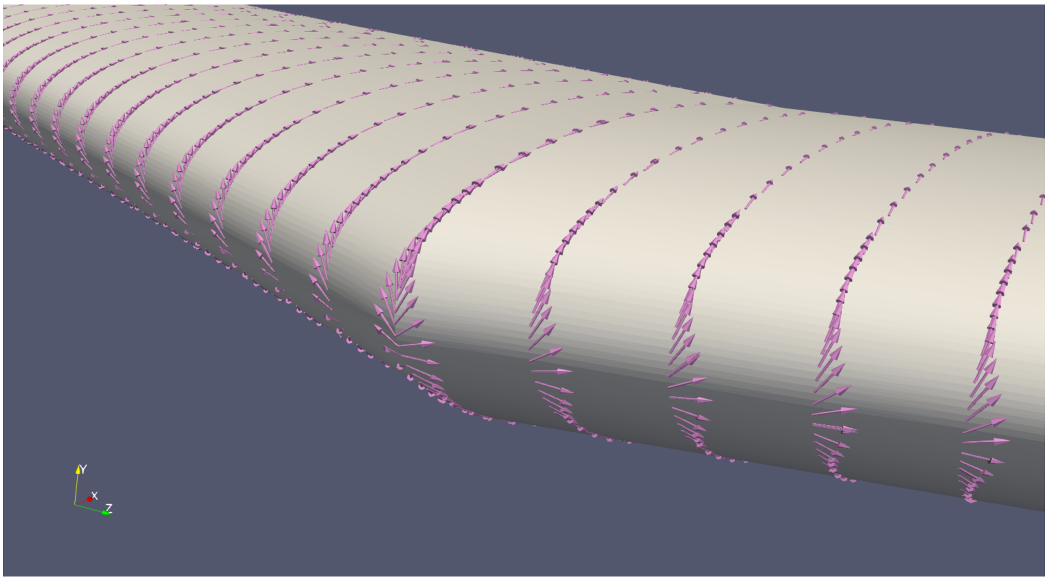


(b) Pressure side

Figure 6.6: Pressure coefficient distribution for a swept wing

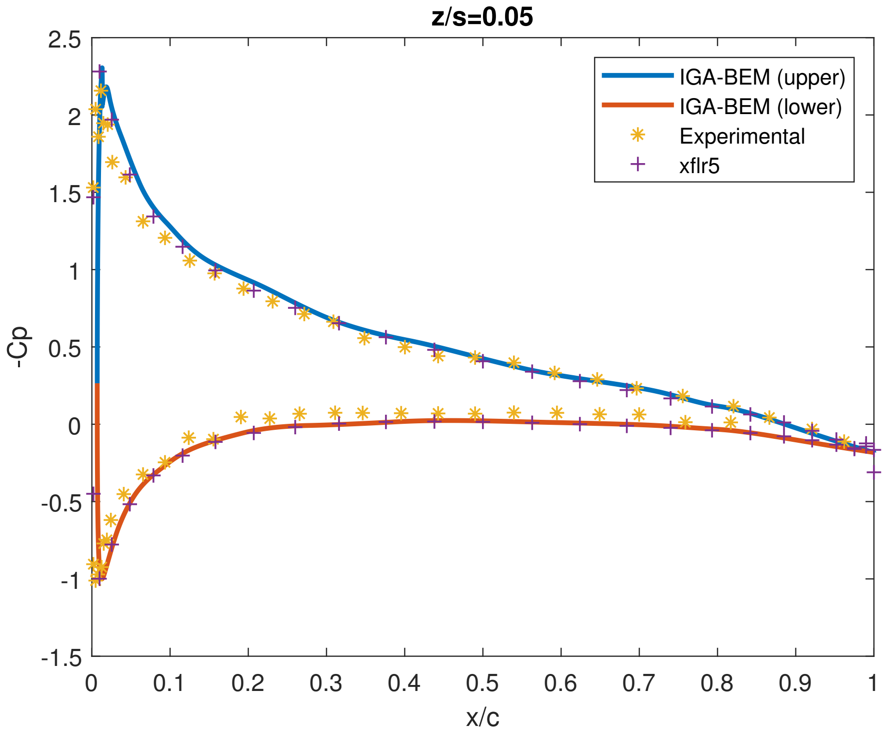


(a) Tip region of unswept wing

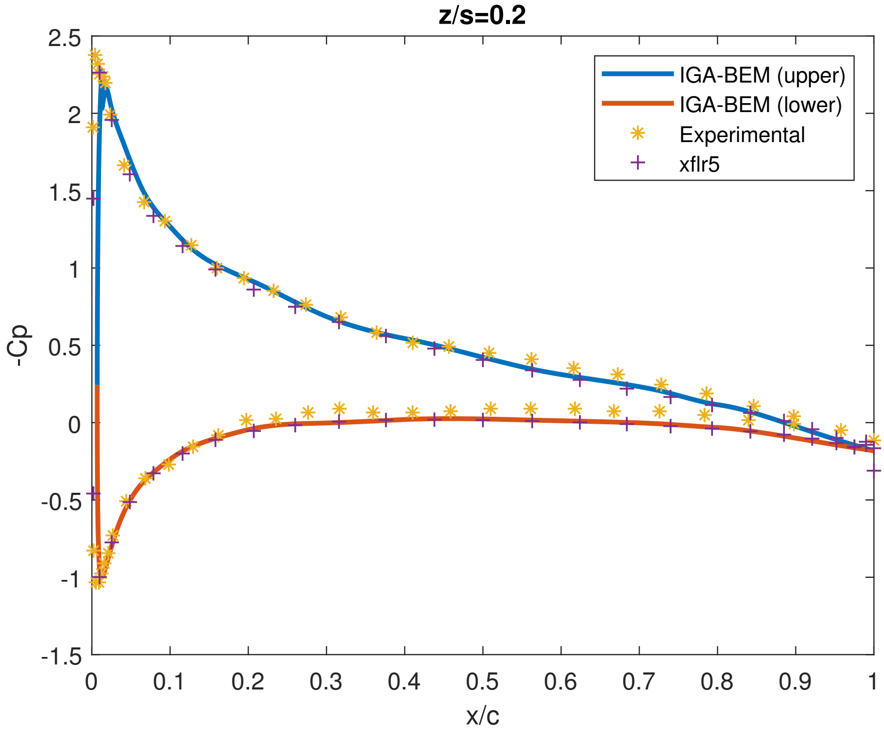


(b) Mid region of swept wing

Figure 6.7: Velocity vectors in regions where three-dimensional phenomena occur

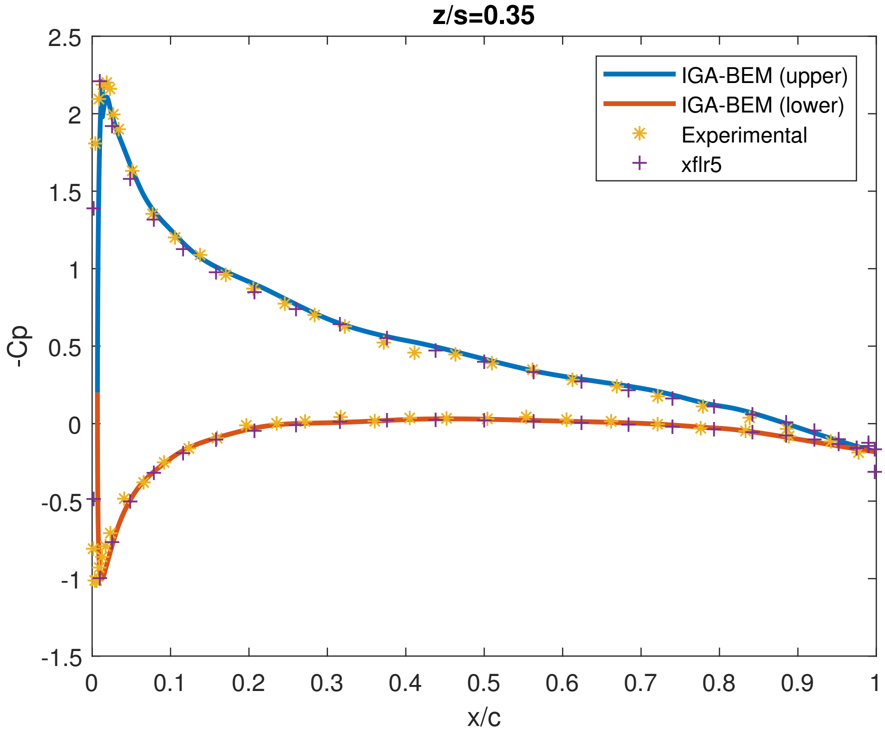


(a)

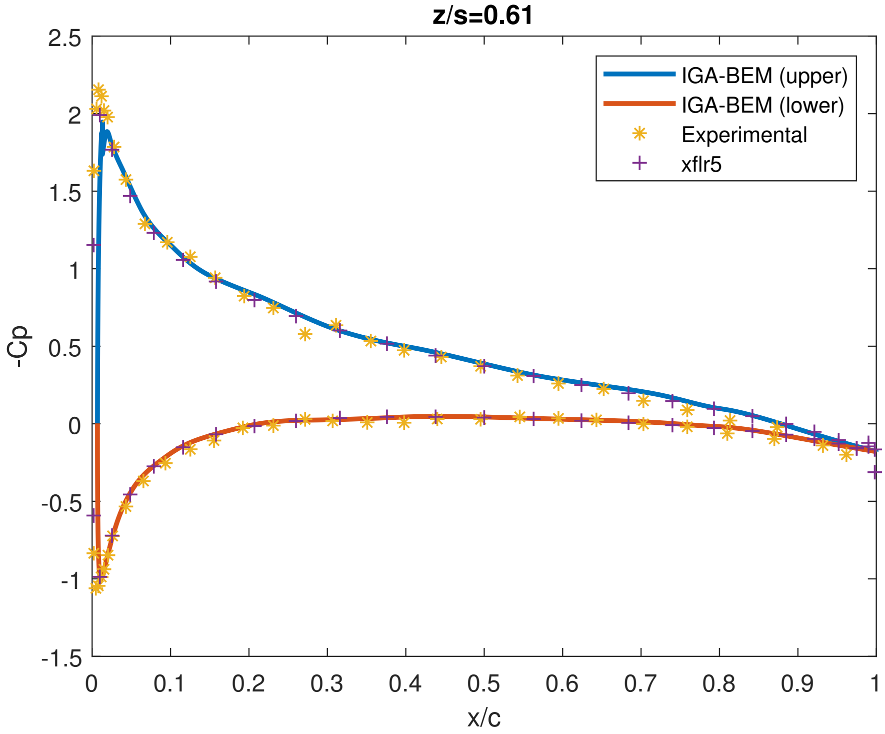


(b)

Figure 6.8: Pressure Coefficients for the Unswept Wing

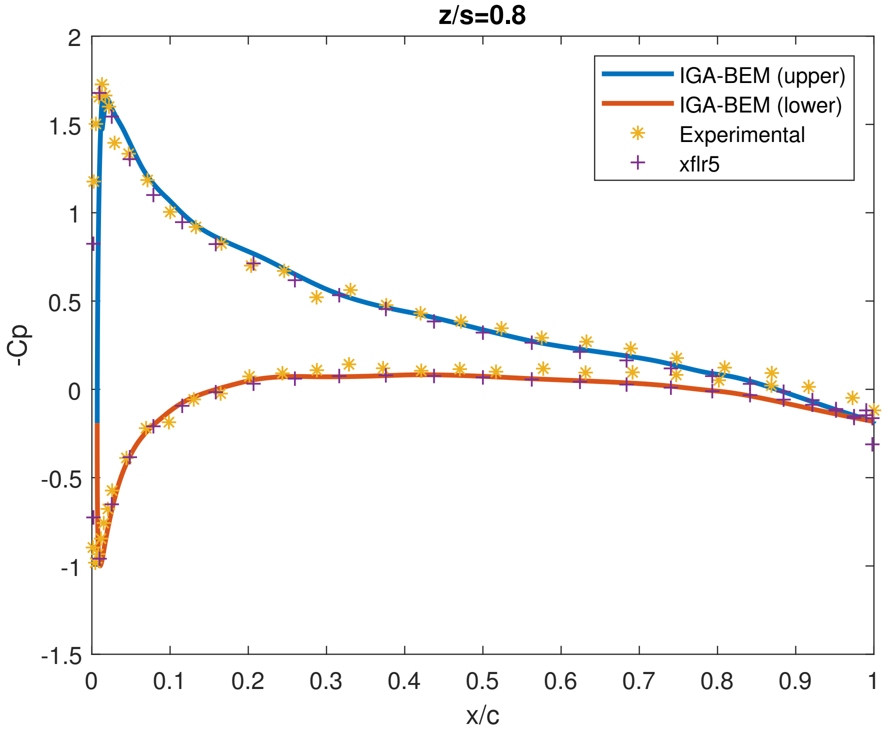


(a)

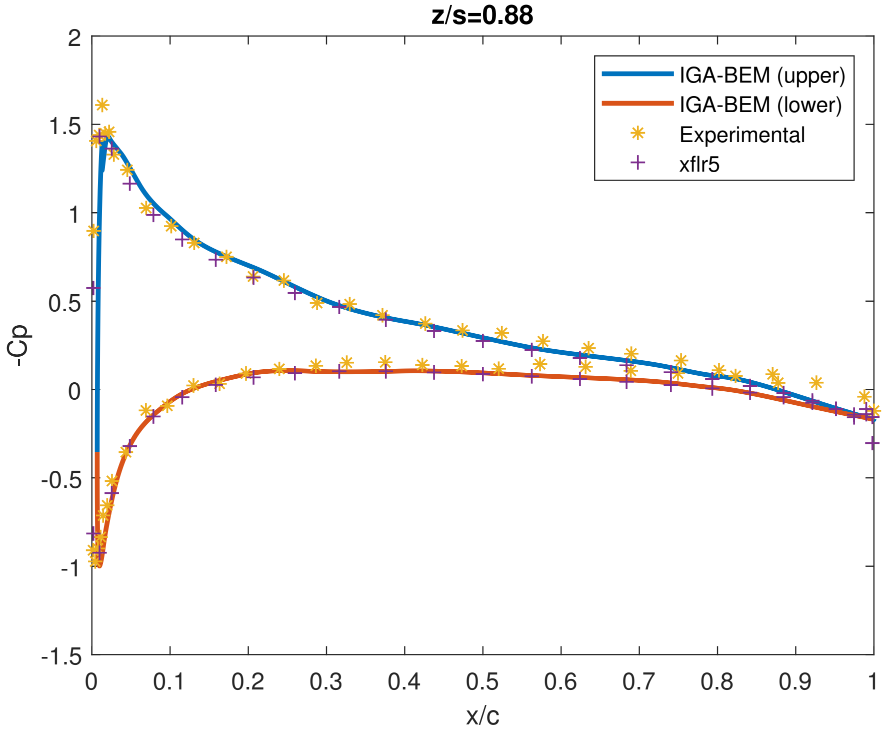


(b)

Figure 6.9: Pressure Coefficients for the Unswept Wing

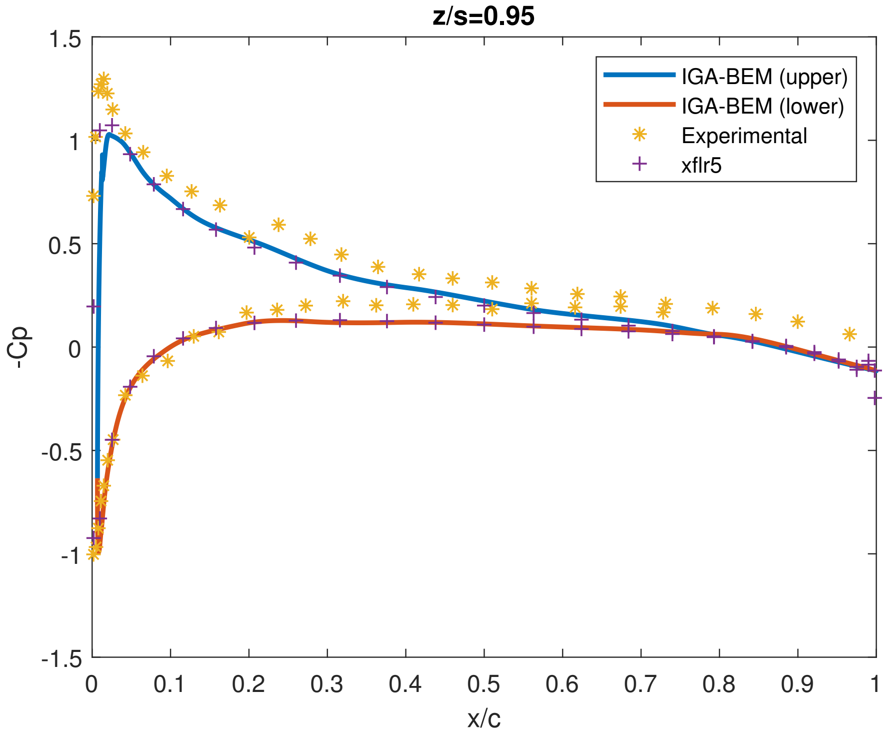


(a)



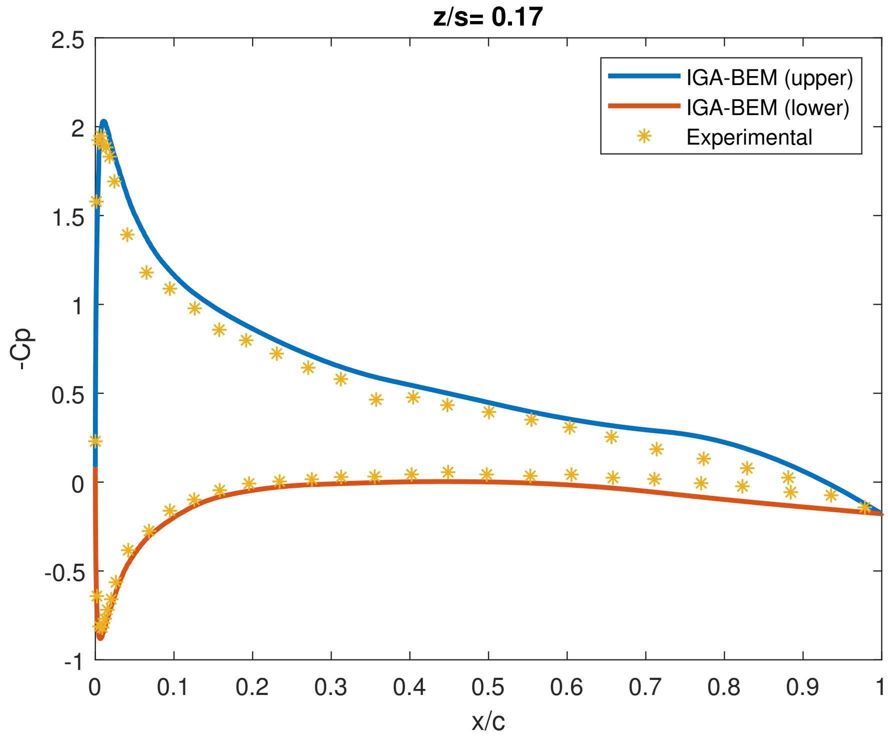
(b)

Figure 6.10: Pressure Coefficients for the Unswept Wing

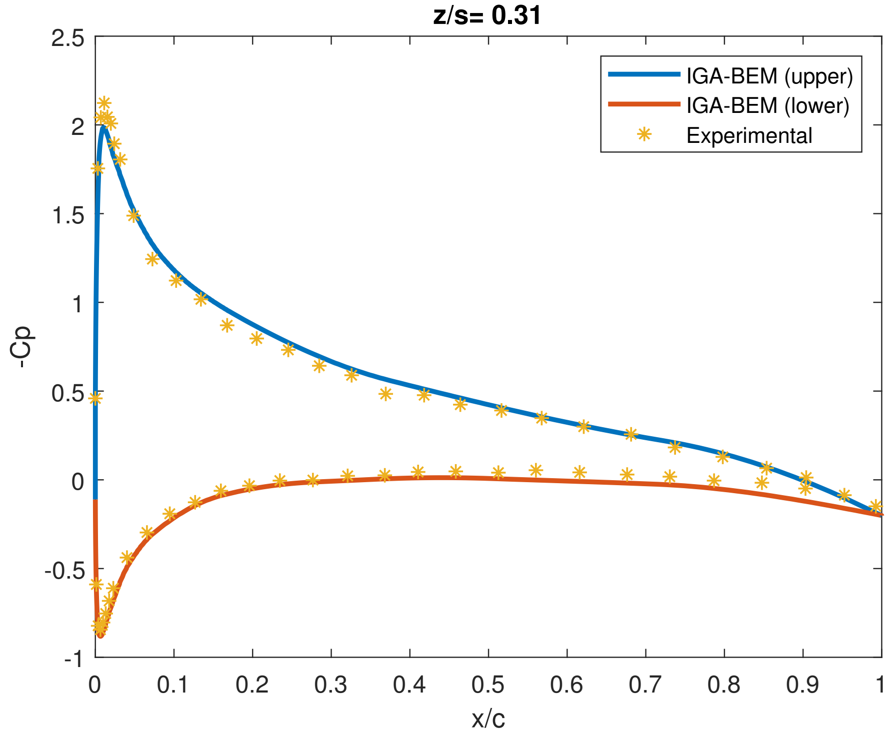


(a)

Figure 6.11: Pressure Coefficients for the Unswept Wing

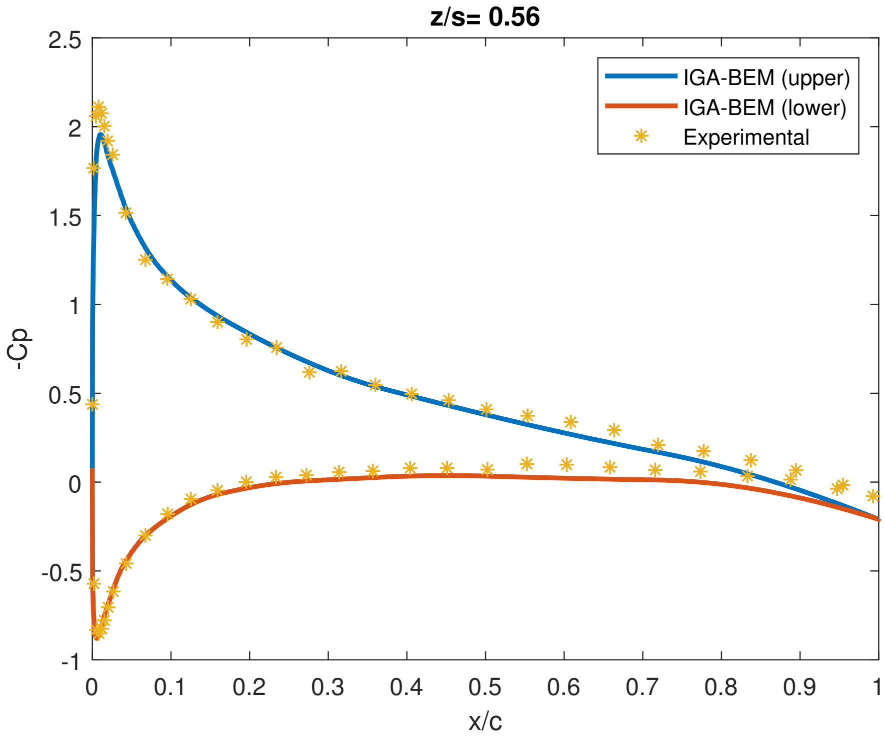


(a)

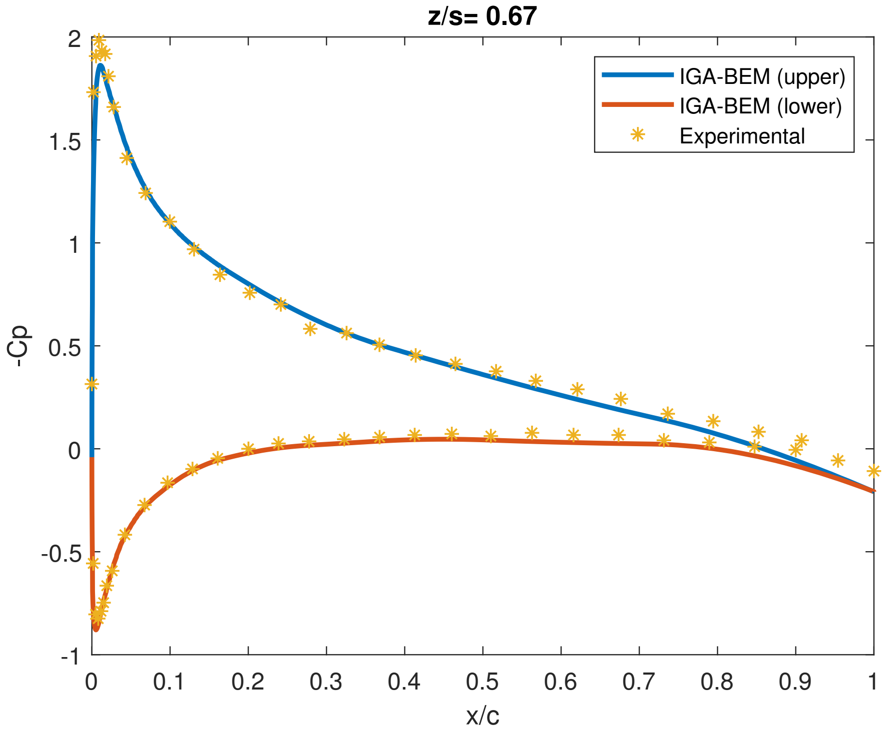


(b)

Figure 6.12: Pressure Coefficients for the Swept Wing

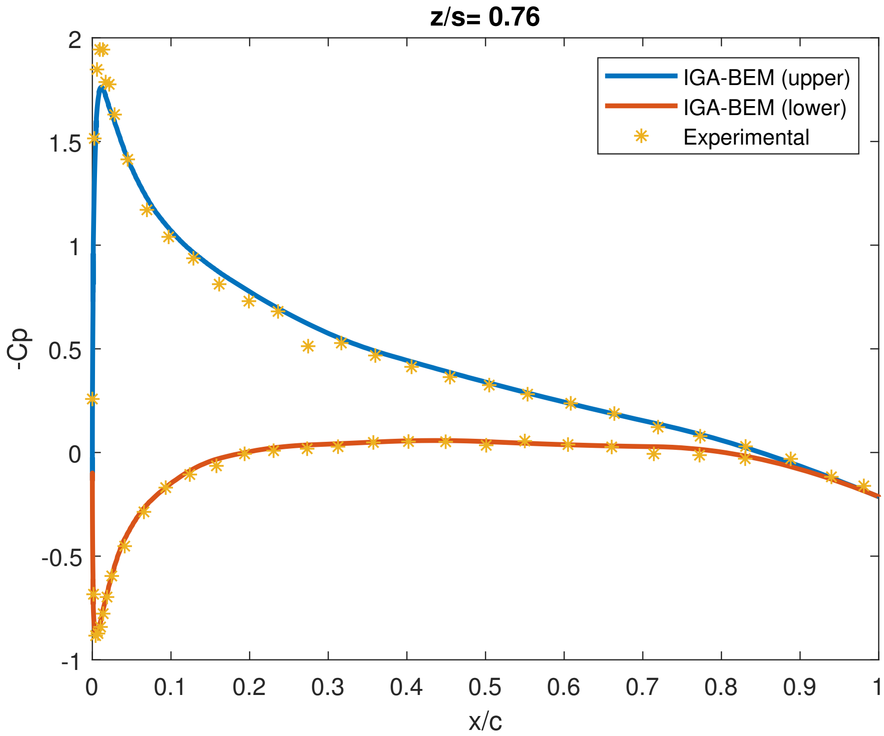


(a)

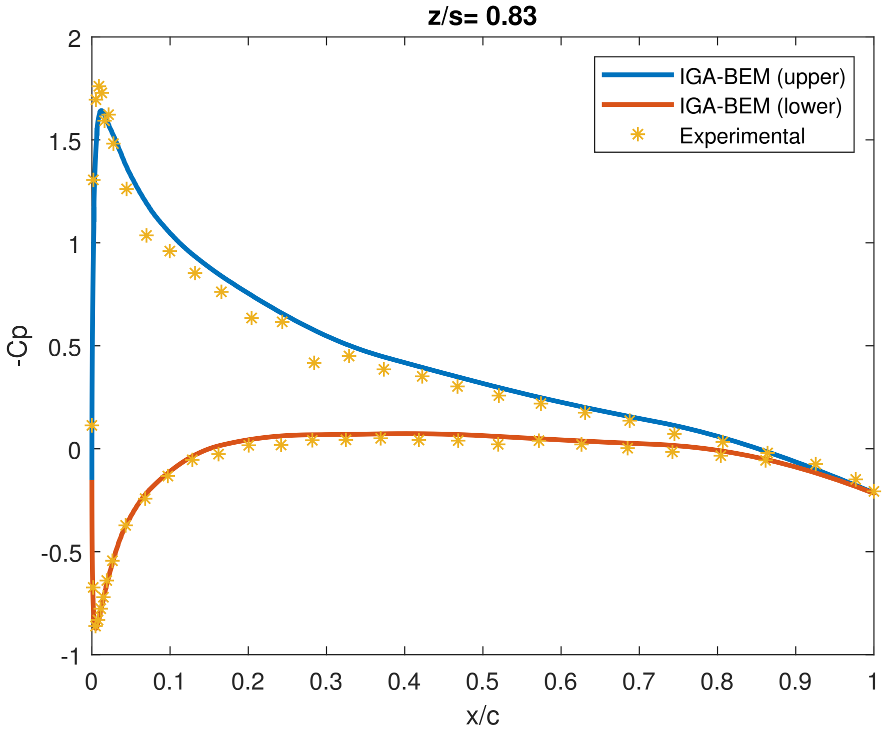


(b)

Figure 6.13: Pressure Coefficients for the Swept Wing

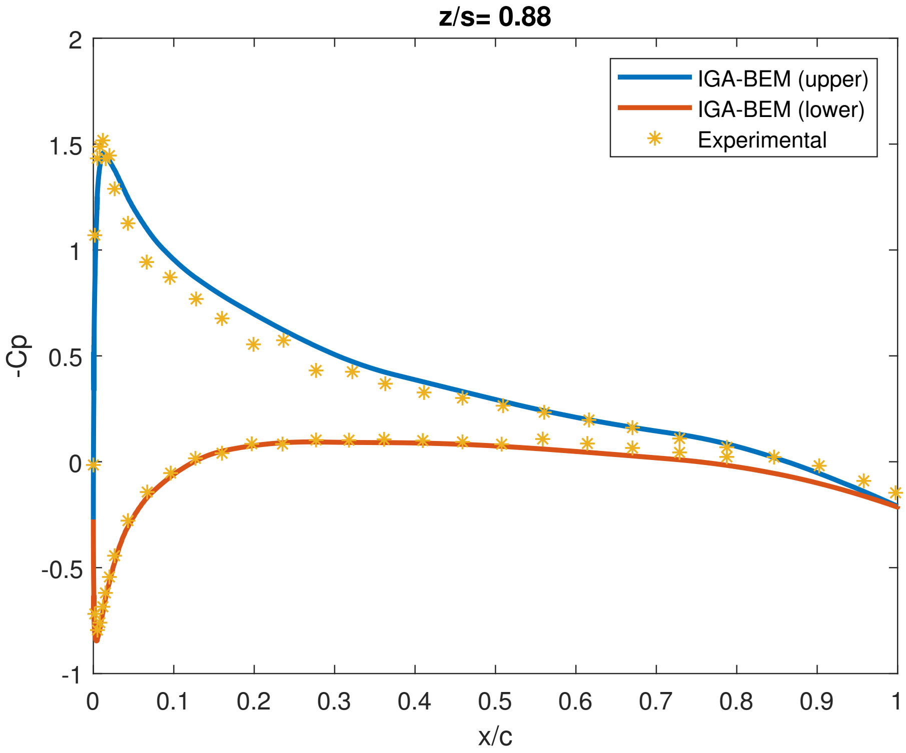


(a)



(b)

Figure 6.14: Pressure Coefficients for the Swept Wing



(a)

Figure 6.15: Pressure Coefficients for the Swept Wing

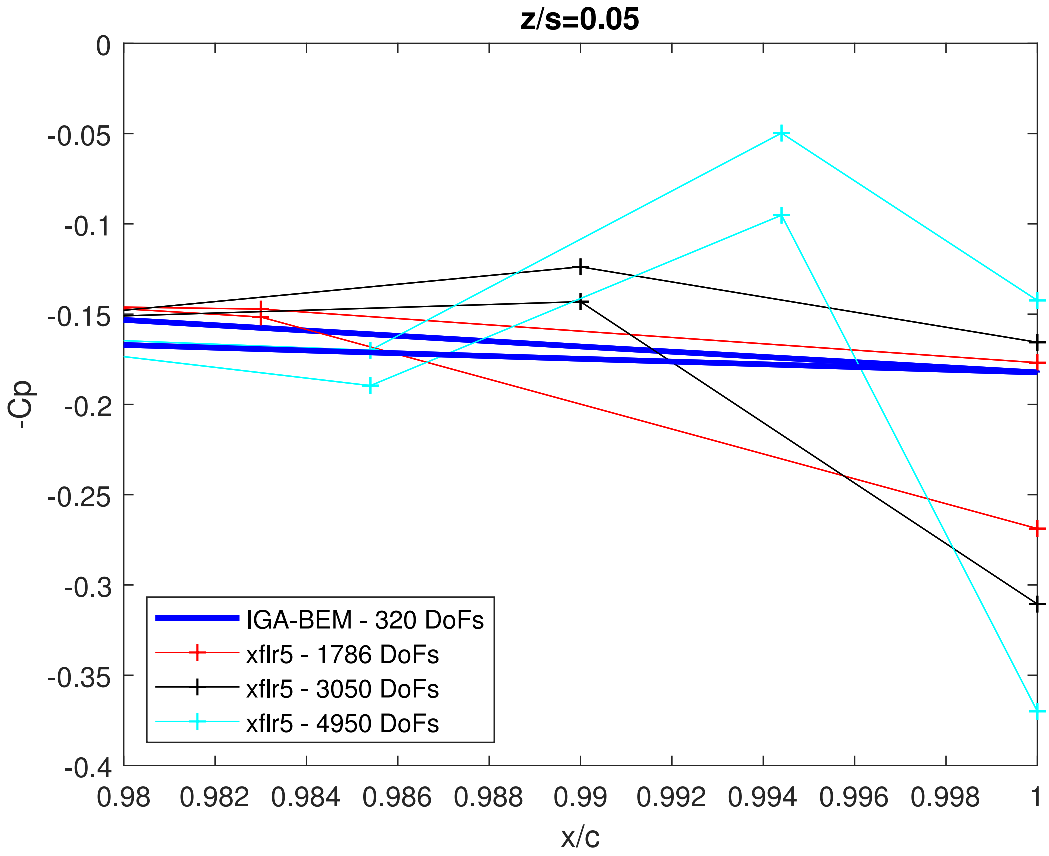
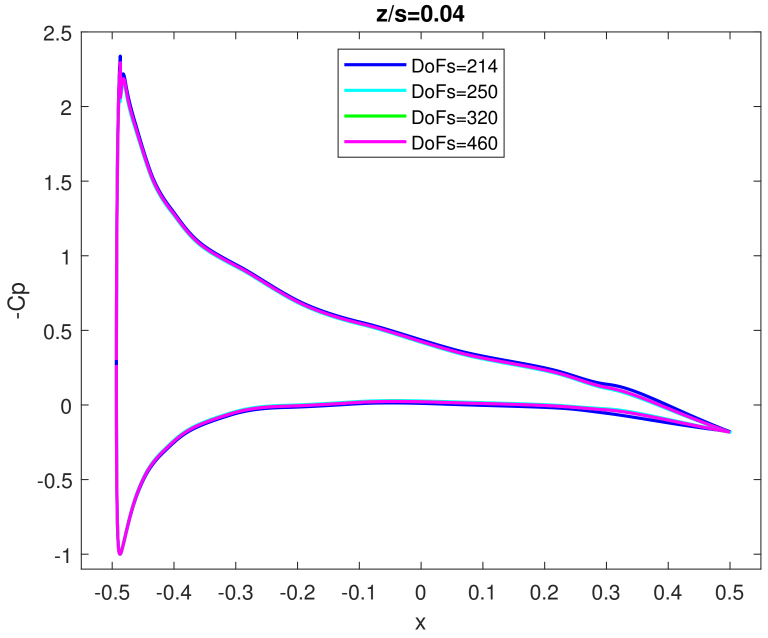
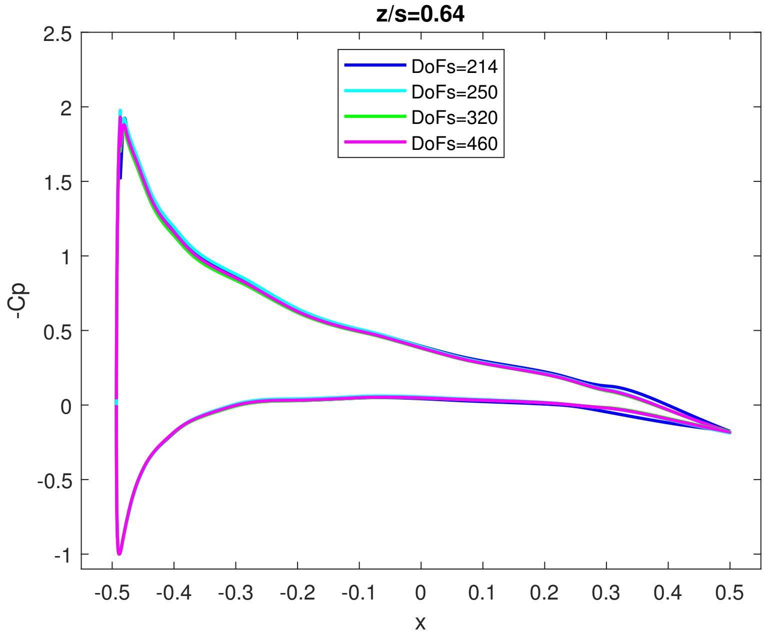


Figure 6.16: Kutta condition accuracy comparison near the mid section of the unswept wing



(a)



(b)

Figure 6.17: Effect of refinement on sectional pressure coefficient

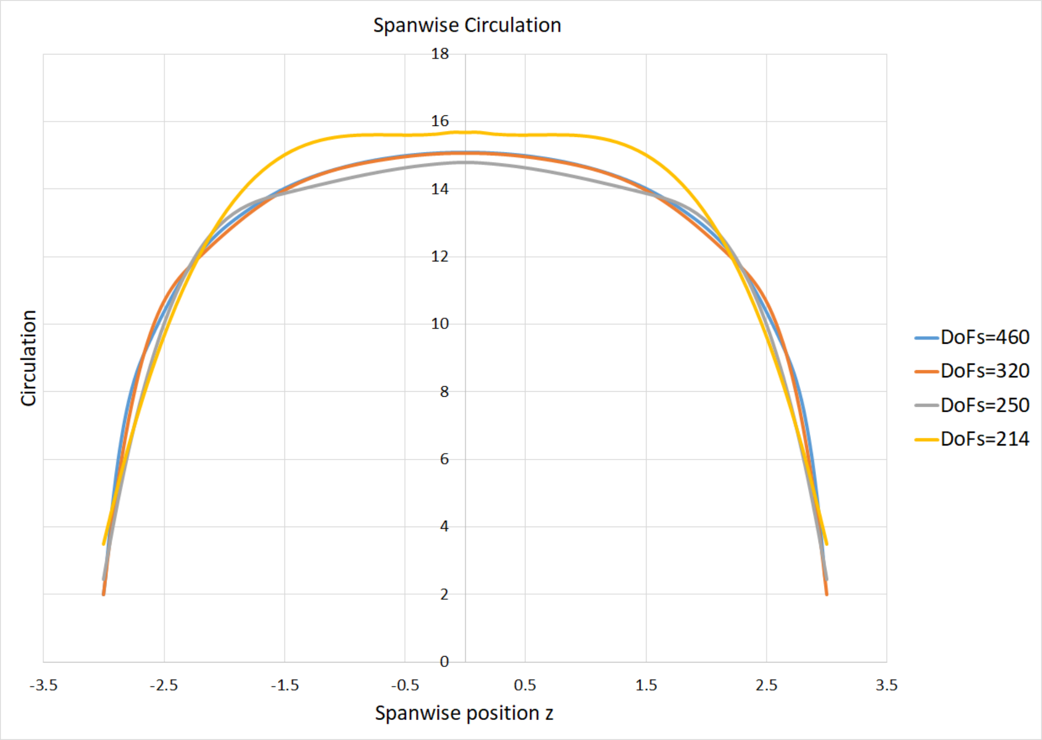


Figure 6.18: Effect of refinement on spanwise circulation

2 A Cambered Unswept Wing with a NACA4412 Airfoil Profile

In this section the method is applied on a rectangular unswept cambered wing with a NACA4412 airfoil profile, chord length $c = 1m$ and semi span $s = 3m$. Once again, the airfoil is generated by translating a NACA4412 profile, represented via the parametrisation described in [46]. A NACA4412 profile is shown in Figure 6.19 and its coordinates are given in able 6.2. The B-spline model of this cambered airfoil is shown in Figure 6.20 along with its curvature graph and control point structure. The flow is uniform along the x-axis with an angle of attack of 0 degrees. The model consists of the main wing (196 control points), cap (83 control points) and wake (133 control points) surfaces. This leads to a total of 286 DoFs. The model has a similar parametrisation to the one of the wings described in the previous section. The main wing part has a NURBS representation while T-junctions and extraordinary vertices exist only on the cap surface which is shown in Figure 6.22 with a close-up in Figure 6.22b.

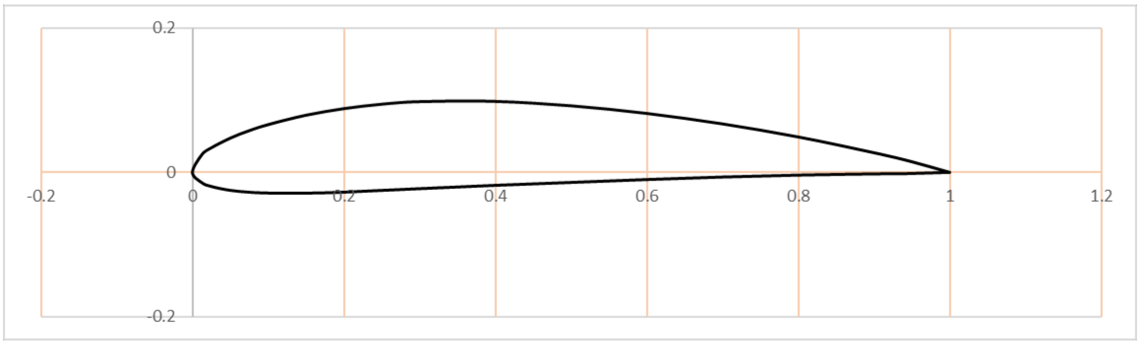


Figure 6.19: NACA4412 Profile

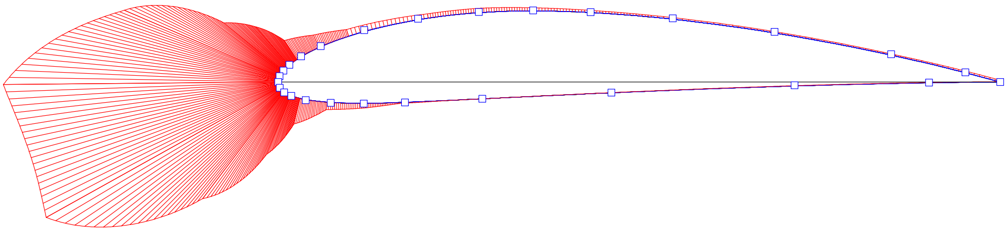


Figure 6.20: NACA4412 airfoil curvature and control point structure

1	0
0.95	0.0147
0.9	0.0271
0.8	0.0489
0.7	0.0669
0.6	0.0814
0.5	0.0919
0.4	0.098
0.3	0.0976
0.25	0.0941
0.2	0.088
0.15	0.0789
0.1	0.0659
0.075	0.0576
0.05	0.0473
0.025	0.0339
0.0125	0.0244
0	0

0	0
0.0125	-0.0143
0.025	-0.0195
0.05	-0.0249
0.075	-0.0274
0.1	-0.0286
0.15	-0.0288
0.2	-0.0274
0.25	-0.025
0.3	-0.0226
0.4	-0.018
0.5	-0.014
0.6	-0.01
0.7	-0.0065
0.8	-0.0039
0.9	-0.0022
0.95	-0.0016
1	0

Table 6.2: Coordinates of NACA4412 profile for upper (left) and lower (right) sides (source: http://airfoiltools.com/airfoil/details?airfoil=naca4412-il)

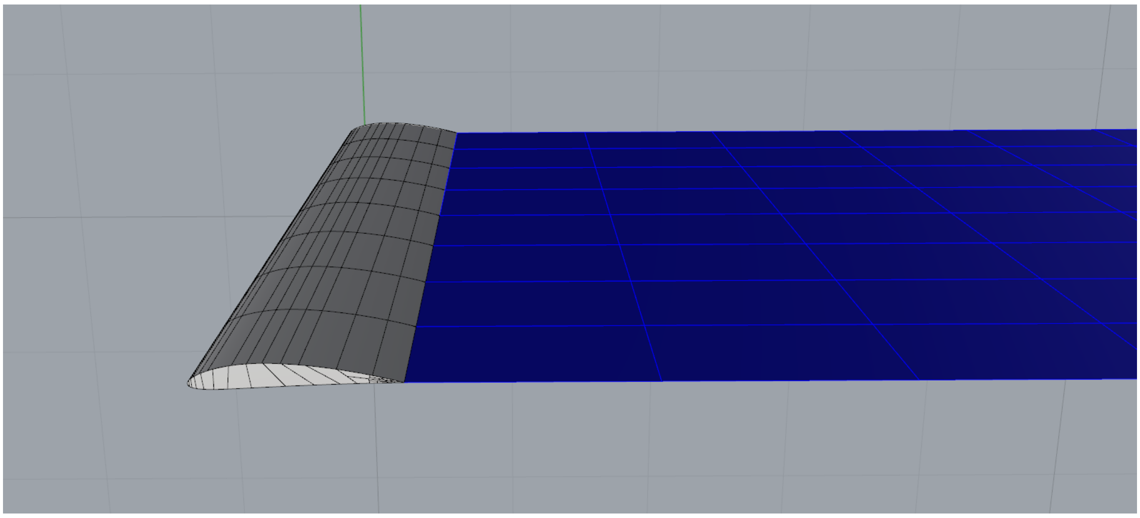
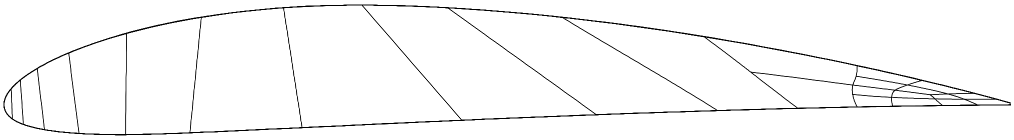
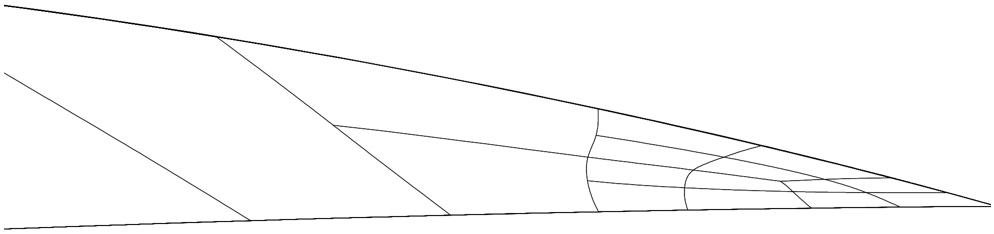


Figure 6.21: NACA4412 wing and wake models



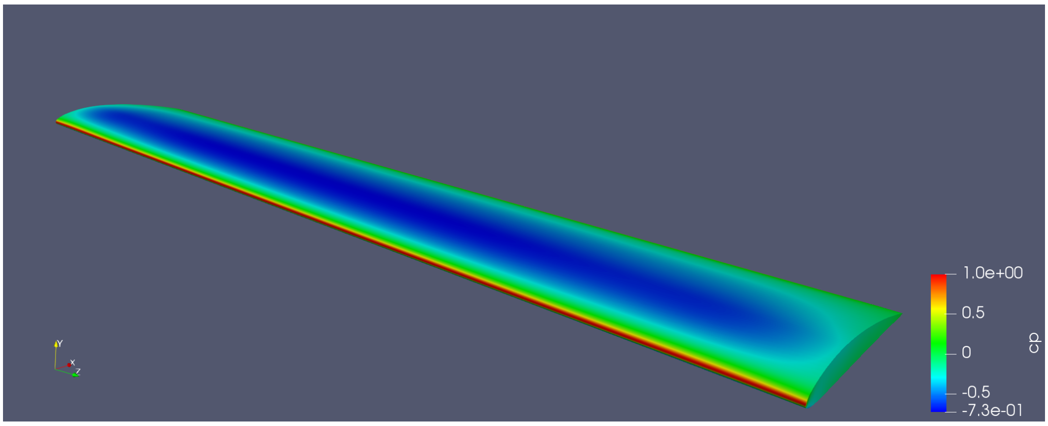
(a) Full Cap



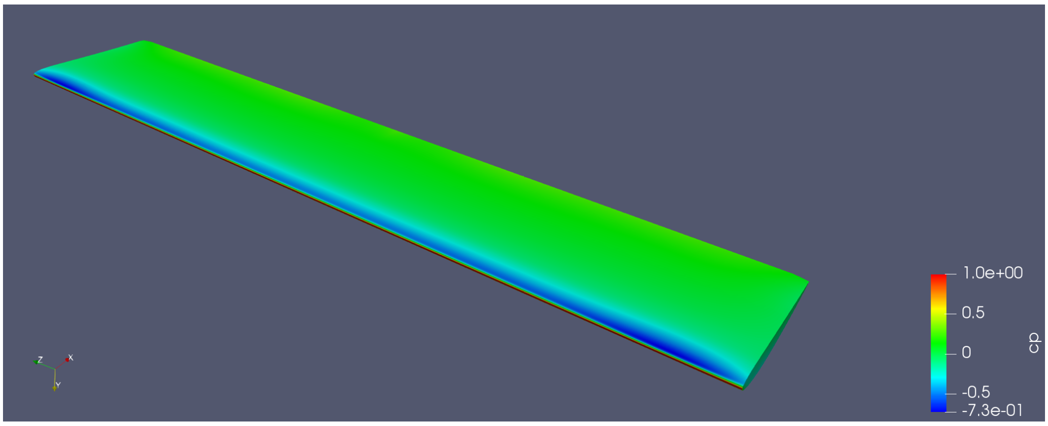
(b) T-junctions and Extraordinary Point of valence 3

Figure 6.22: NACA4412 Wing Cap Surface

The pressure coefficient distribution on the suction side of the wing is demonstrated in 6.23. Numerical results for pressure coefficients are compared with the ones from xflr5 for a low order model of 3050 degrees of freedom and are shown in Figures 6.24 to 6.28. Once again the two methods seem to be in agreement for the biggest part of the chord with the IGA-BEM method providing equivalent results for much fewer DoFs. At the same time, the low order panel method demonstrates a similar oscillatory behaviour near the trailing edge as to the one mentioned in the previous section.

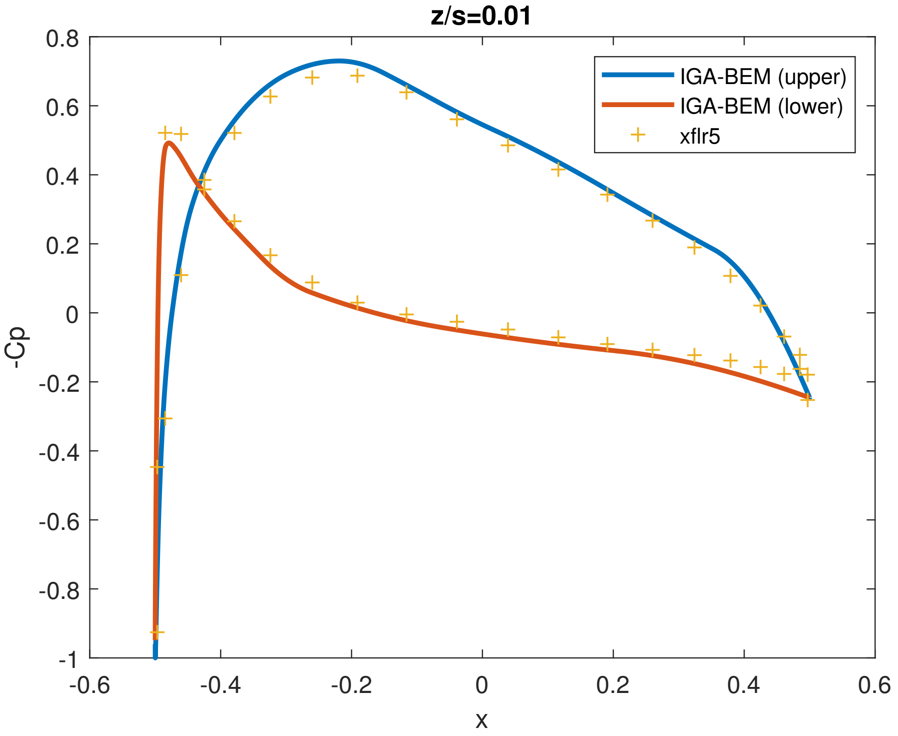


(a) Suction side

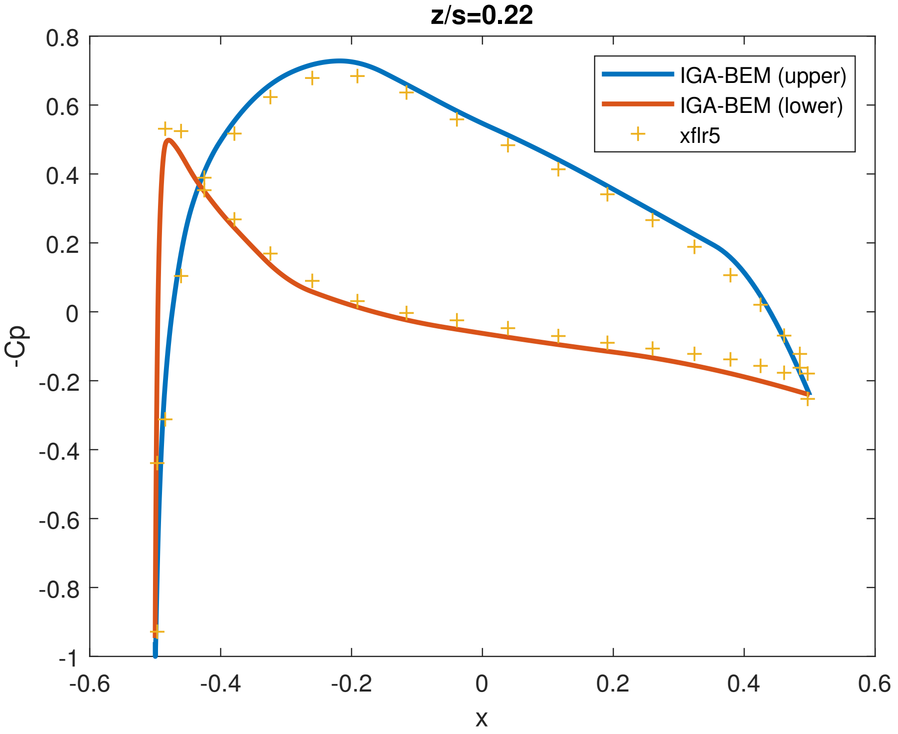


(b) Pressure side

Figure 6.23: Pressure coefficient distribution for the cambered wing



(a)



(b)

Figure 6.24: Pressure Coefficients for the Cambered NACA 4412 Wing

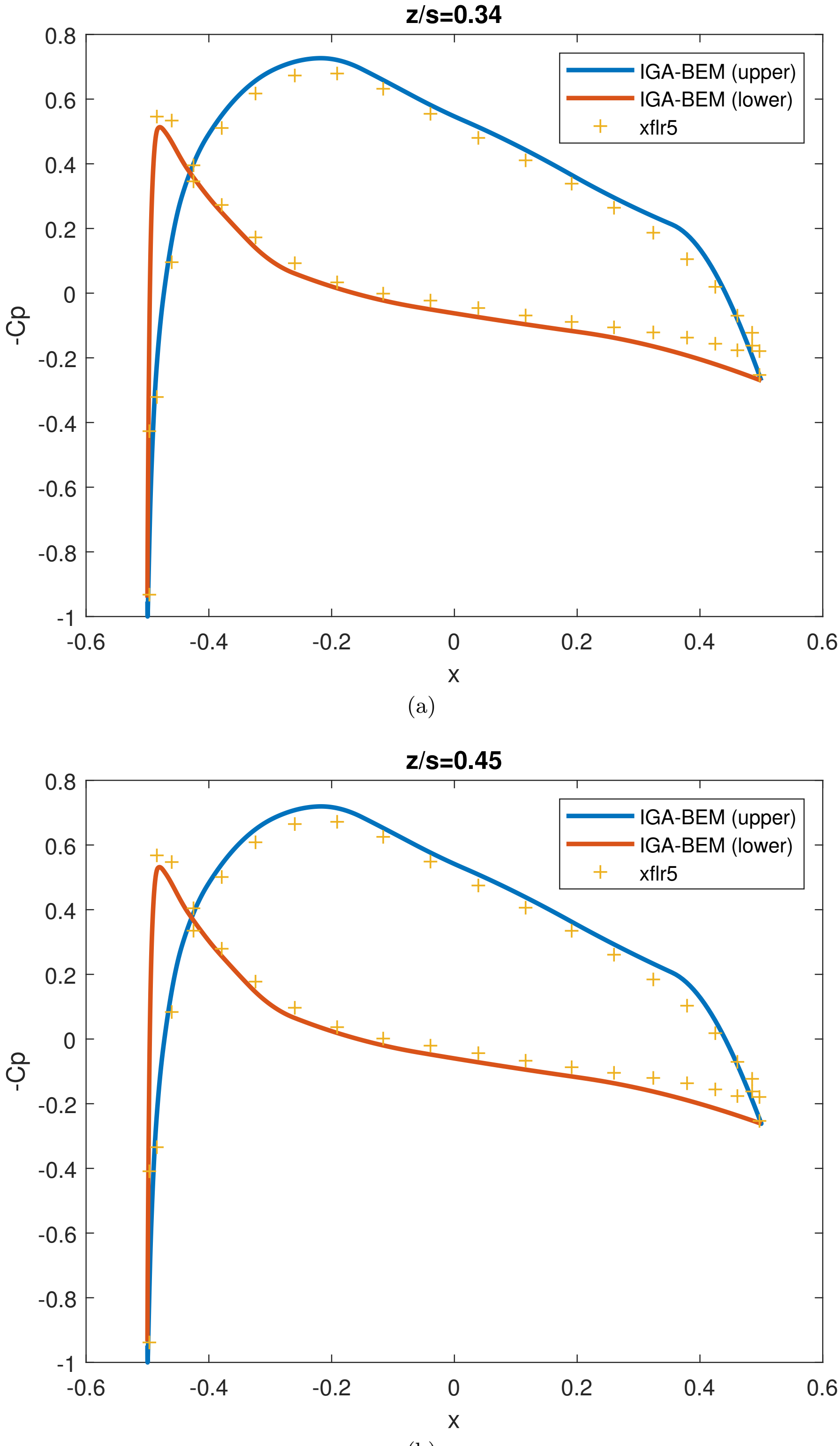
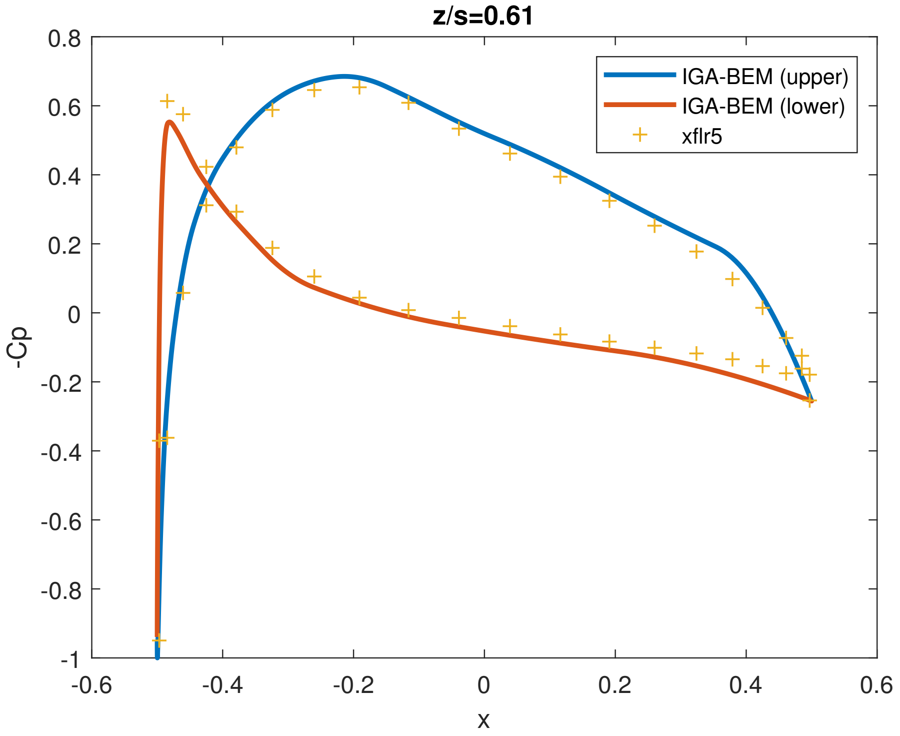
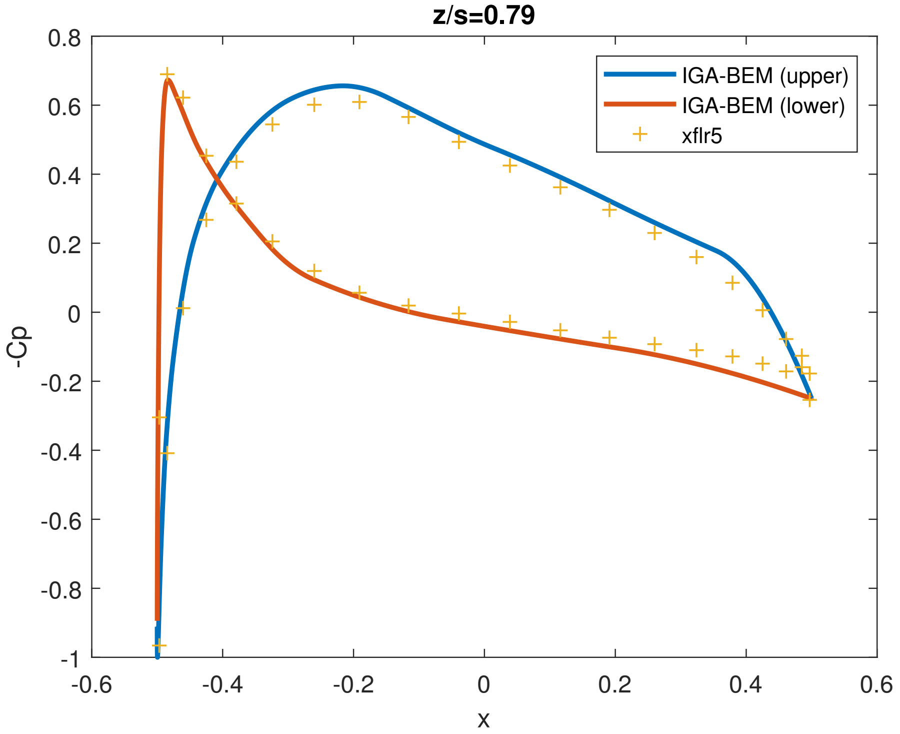


Figure 6.25: Pressure Coefficients for the Cambered NACA 4412 Wing

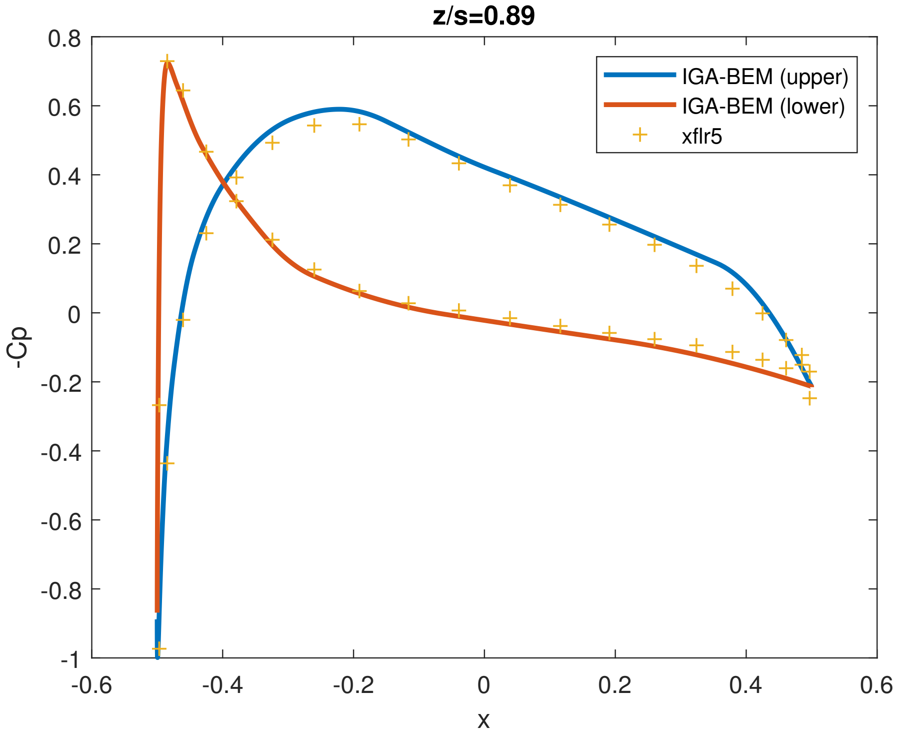


(a)

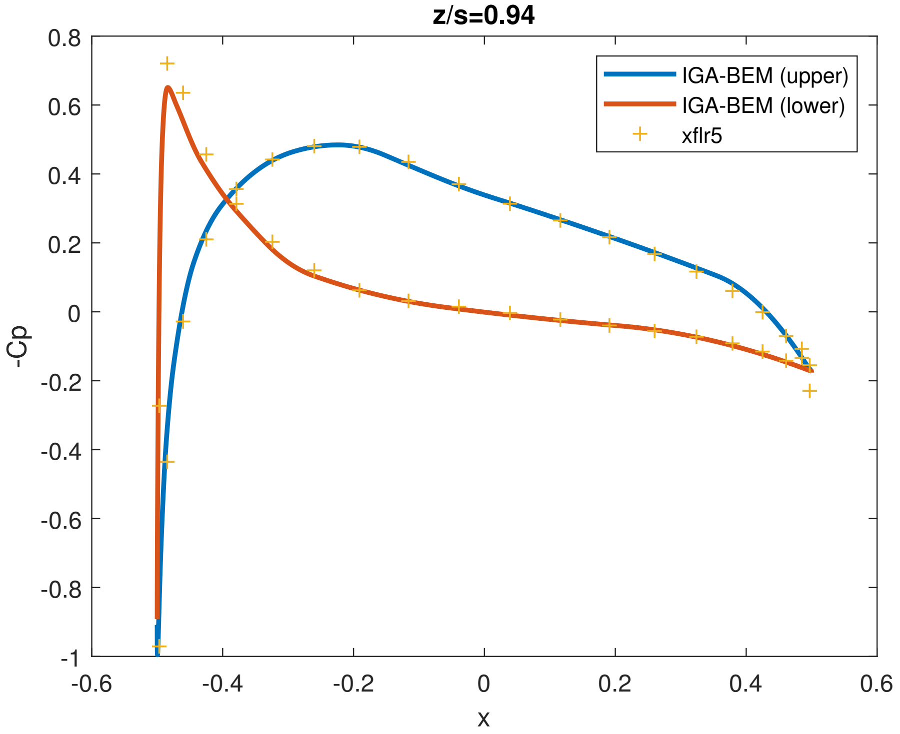


(b)

Figure 6.26: Pressure Coefficients for the Cambered NACA 4412 Wing



(a)



(b)

Figure 6.27: Pressure Coefficients for the Cambered NACA 4412 Wing

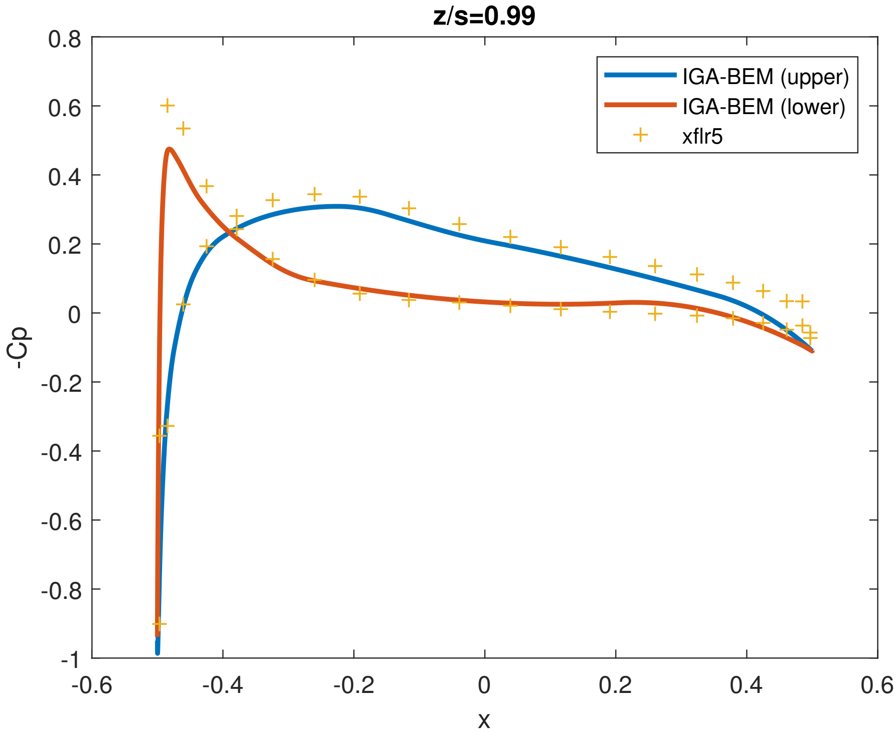


Figure 6.28: Pressure Coefficients for the Cambered NACA 4412 Wing

3 IGA-based Kutta Condition and Flow near the Trailing Edge

In this section the behaviour of the IGA-based solution of the problem at the trailing edge (TE) and especially near its tip is investigated. In this connection, the accuracy and influence of Kutta condition, formulated as a zero pressure-jump condition on the TE (see (5.25)) and numerically implemented via (5.32), is assessed and discussed.

Figure 6.29 depicts the 0^{th} order iterations, i.e, initial estimations, employed, for different levels of refinement, in the Newton-Raphson scheme used for solving the non-linear system in (5.32). It is clear from this figure that there is an increasing error in the estimation of the actual zero pressure-jump requirement, especially near the wing tip of the TE, for an increasing number of DoFs. It is also worth noticing that a small but constant error persists along the whole TE. This deviation can be explained by the fact that the initial estimation corresponds to the solution of the non-linear system (5.32) with the so-called Morino's condition. This condition states that the potential jump on the wake can be approximated by the difference between the potential values on the upper and lower parts of the wing on the TE. The linear nature of Morino's condition prevents the 0^{th} order iteration from revealing the observed span-wise cross flow, especially near the tip.

Figure 6.29 also indicates that the deviation of the pressure jump from zero becomes more severe as the DoFs of the problem increase. This behaviour is related to the existence of a singularity of the velocity $\nabla\phi$ at the intersection of the trailing edge with the tip, where the flow encounters two sharp edges with dihedral angles greater than π. There has been extensive research in the general field of potential theory as well as in aerodynamic lifting flows for non-smooth domains including sharp edges; see, e.g., [84], [10], and [39], [51]. This analysis concludes that in the vicinity of an edge the

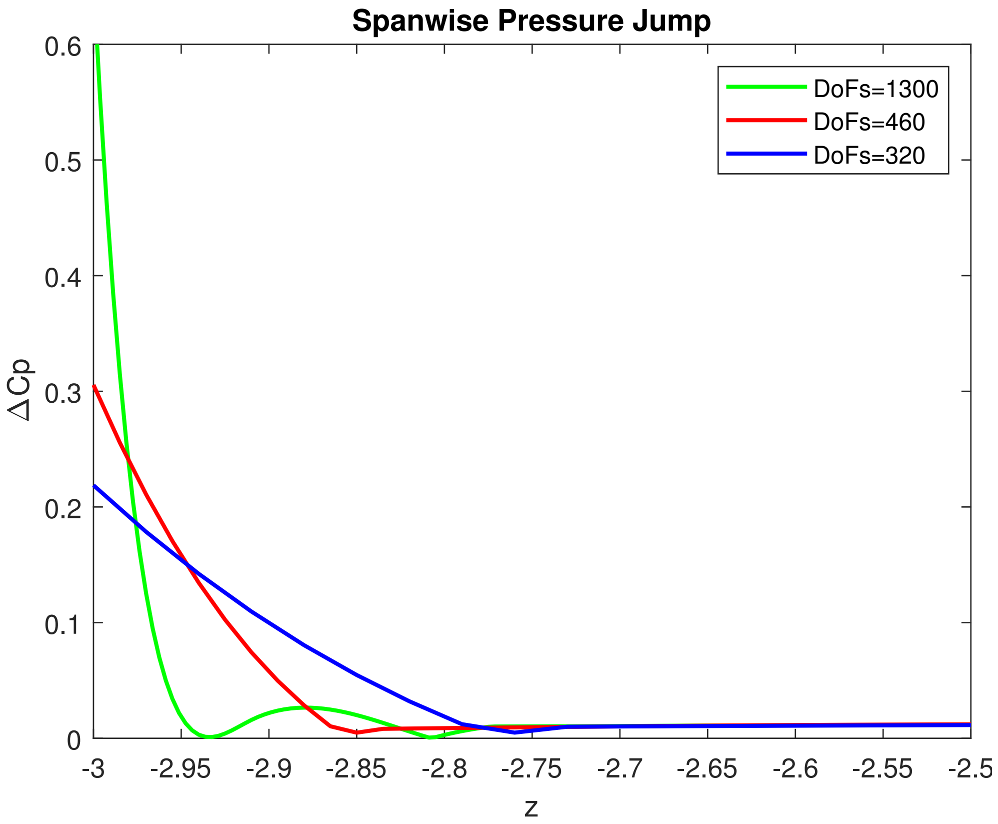


Figure 6.29: Pressure jump along the trailing edge at the 0th iteration

gradient of the solution $\phi(\mathbf{P})$ of 3D Laplace equation exhibits the following asymptotic behaviour:

$$\nabla\phi(\mathbf{P}) = O\big(r^{-(1-\pi)/\alpha}\big), \tag{6.1}$$

where r denotes the Euclidean distance of the field point $\mathbf{P}$ from the edge and α is the magnitude of the exterior of the dihedral angle formed at the edge of the body.

Figure 6.30 illustrates the configuration for the wing case by taking a plane $\mathcal{P}(x)$, which intersects vertically both edges where the wing surface meets the wing cap. For the wing geometries considered in this paper (see Figure 6.3) it is evident that, as long as the longitudinal coordinate x of $\mathcal{P}(x)$ is different from that of the trailing edge, $\alpha = 3\pi/2$ at both the suction and the pressure edge of the wing-cap. Then, (6.1) gives:

$$\nabla\phi(\mathbf{P}) = O(r^{-1/3}),\ x{\neq}\mathrm{TE}_x. \tag{6.2}$$

However, when $\mathcal{P}(x)$ reaches the trailing edge, $x = \mathrm{TE}_x$, a discontinuity occurs: the two equal dihedral angles at the pressure and suction intersection collapse to a single one equal to $\alpha = 2\pi$, leading to the estimate:

$$\nabla\phi(\mathbf{P}) = O(r^{-1/2}),\ x{=}\mathrm{TE}_x, \tag{6.3}$$

which indicates that the singularity of the velocity at the TE-tip is stronger than that along the remaining cap-edge.

On the basis of the above discussion it seems legitimate to expect that a higher number of DoFs will have the capacity to reveal the velocity singularity at the tip of TE more accurately. As a result, the error of the initial estimation increases with DoFs, yielding

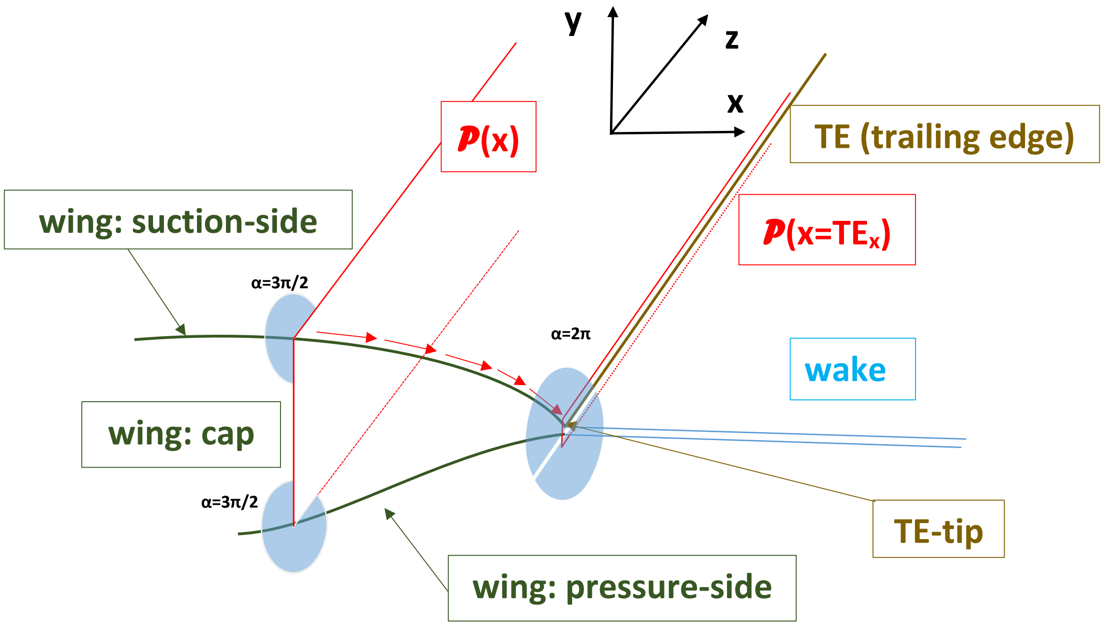


Figure 6.30: An intersecting plane $\mathcal{P}(x)$ sliding towards the trailing edge TE

slower convergence in the employed iterative scheme; see Figure 6.31.

The effect of tip singularity described in (6.2) is of local character, as it is clearly seen in Figure 6.29. This local character of the tip singularity is also depicted in Figures 6.32 to 6.35 where the chordwise distribution of C_p at the tip location $(z/s = 1)$ and in three sections near the tip $(z/s = 0.99, 0.95, 0.9)$ are presented. However, starting from the section at $z/s = 0.95$ and for the remaining wing sections, the initial (0^{th}) and final iterations are almost identical.

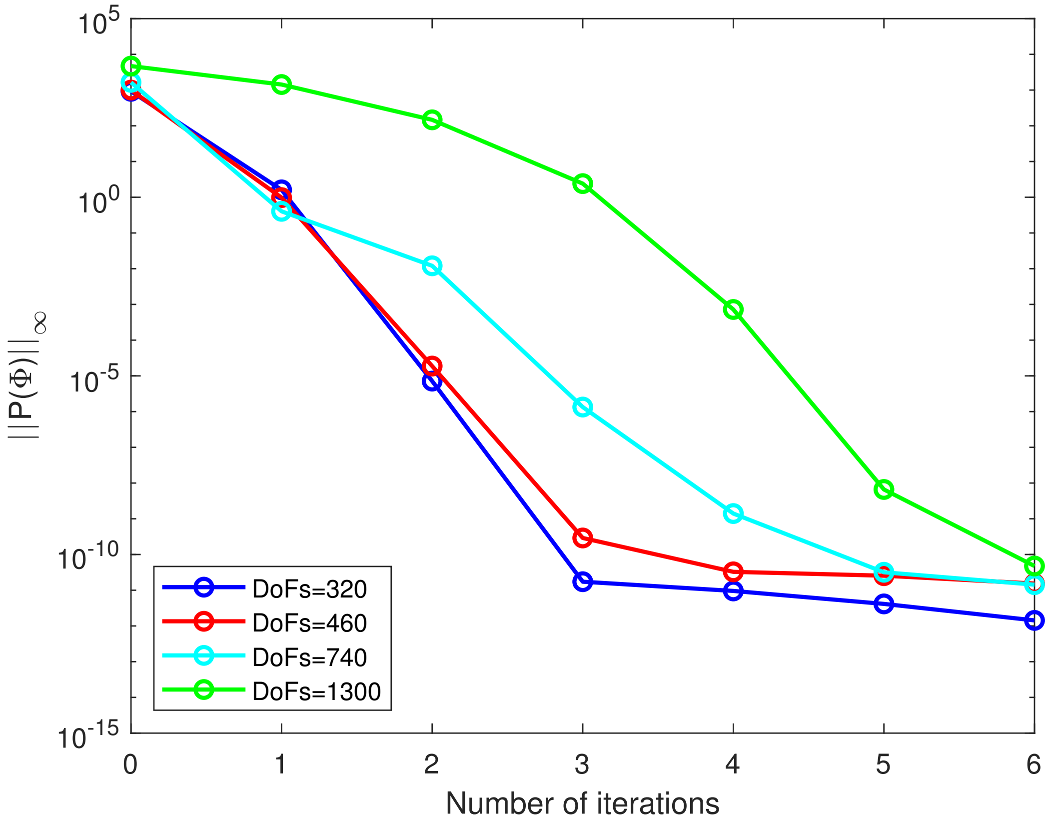
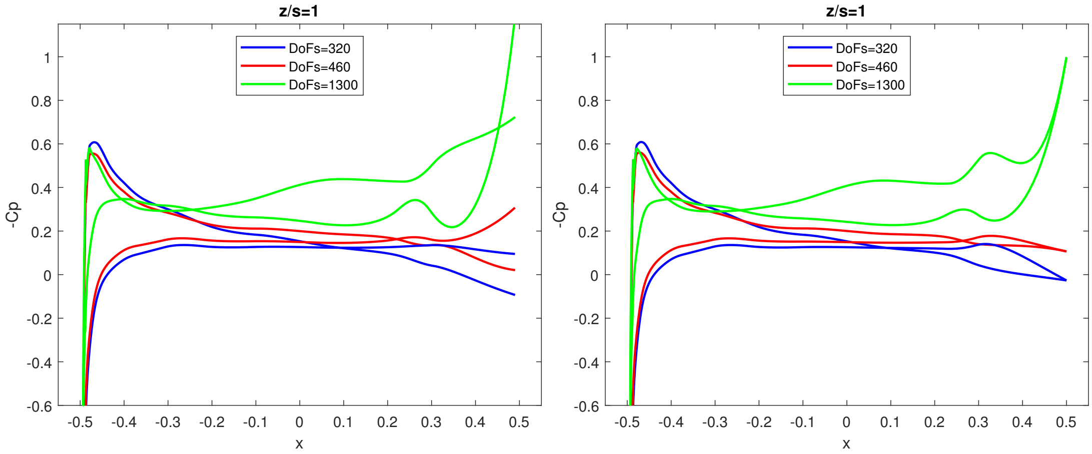


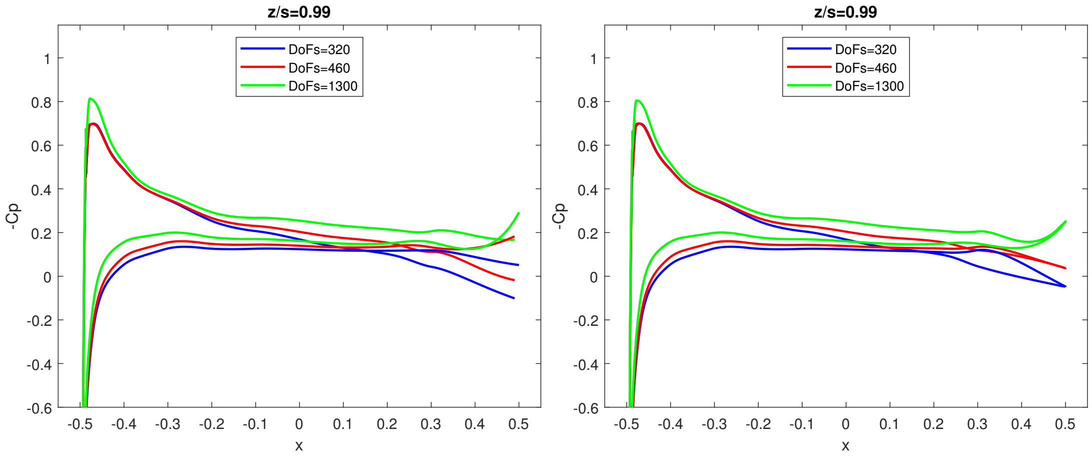
Figure 6.31: Newton-Raphson scheme convergence for different refinement levels



(a) 0^{th} iteration (Morino)

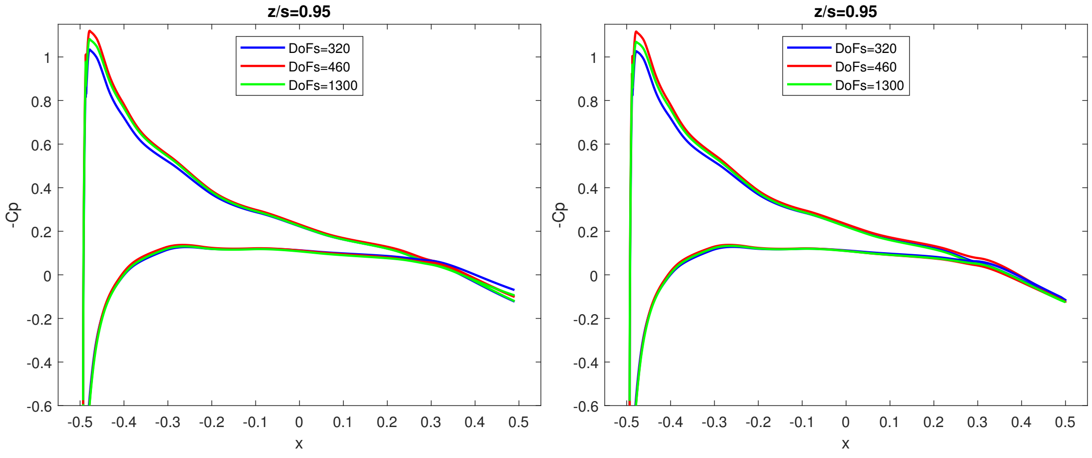
(b) After convergence

Figure 6.32: Tip chordwise C_p distribution for different refinement levels



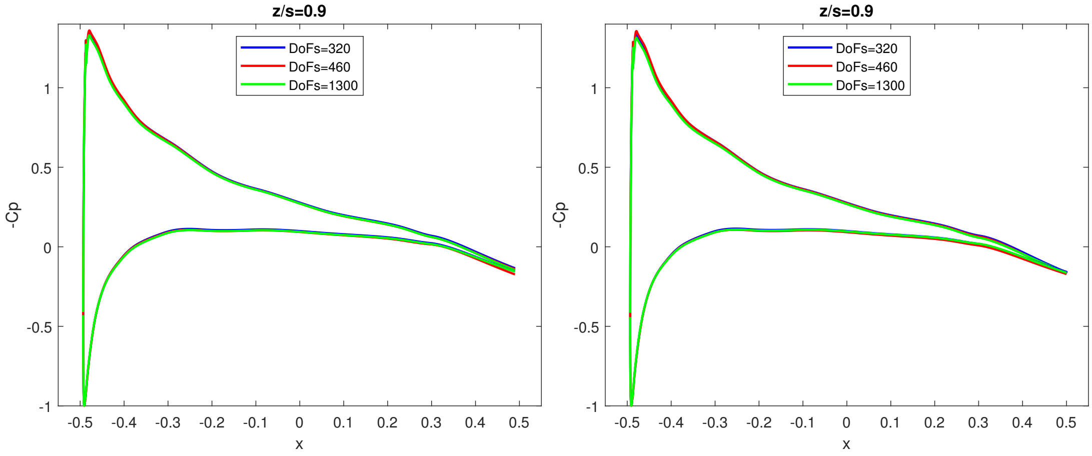
(a) 0^{th} iteration (Morino) (b) After convergence

Figure 6.33: $z/s = 0.99$ chordwise C_p distribution for different refinement levels



(a) 0^{th} iteration (Morino) (b) After convergence

Figure 6.34: $z/s = 0.95$ chordwise C_p distribution for different refinement levels



(a) 0^{th} iteration (Morino)

(b) After convergence

Figure 6.35: $z/s = 0.9$ chordwise C_p distribution for different refinement levels

In the light of the singular character of the solution in the vicinity of the tip, another critical parameter to be assessed for its influence on the accuracy of Kutta condition is the position of the collocation points, especially those near the tip. To assess the performance of Kutta condition we use L^1 and L^∞ norms, defined as:

$$||\Delta cp(\eta)||_{L^1} = \int_{\eta_{mid}}^{\eta_{tip}} \mid \Delta cp(\eta) \mid d\eta \tag{6.4}$$

$$\|\Delta cp(\eta)\|_{L^\infty} = max\{\Delta cp(\eta), \quad \eta_{tip} \leq \eta \leq \eta_{mid}\} \tag{6.5}$$

where η_{tip} and η_{mid} are the parametric values of the trailing edge curve that correspond to the tip and mid sections respectively. The wing used for this analysis is the original configuration, shown in Figure 6.3a, but with local refinement at the trailing edge region, as shown in Figure 6.36.

Results for the pressure jump on the trailing edge are presented when the last collocation point is on the tip ($z = -3$) and after shifting it by $0.01, 0.04, 0.07$ and 0.09; see Figure 6.37. The rest of the collocation points are also shifted in a way that prevents

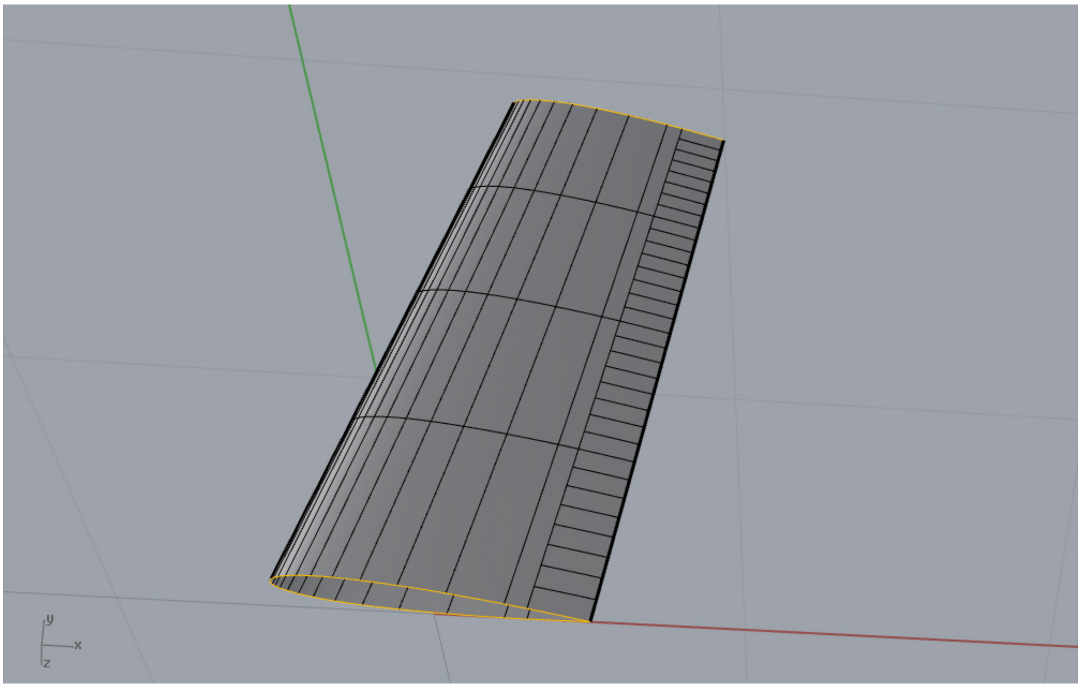
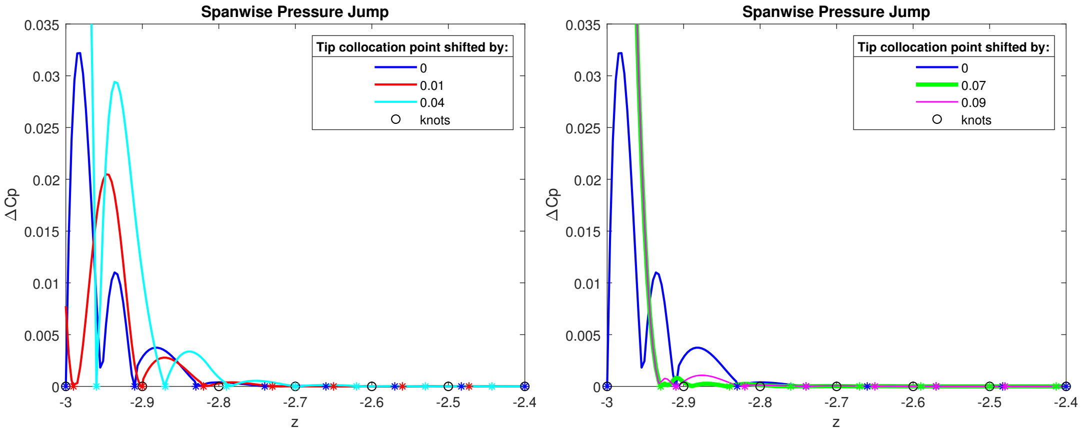


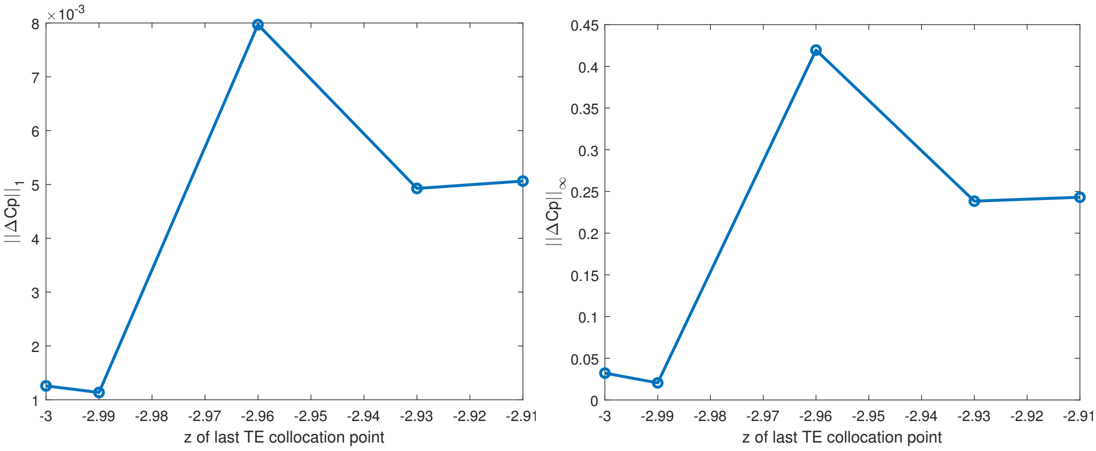
Figure 6.36: Wing under investigation with TE DoF=35

adjacent ones from being too close to each other, in the sense that the distance between them should not exceed a minimum threshold, while they still remain inside the support of each basis function of the TE curve. The corresponding L^1 and L^∞ norms are presented in Figure 6.38. It is clear that, regarding the accuracy of Kutta condition, the best practice is a small shifting of the last collocation point (on the tip); in our case, the best shifting is 0.01. This result remains valid, independently of the level of refinement of the trailing edge region.



(a) Closest to tip shifted points (b) Further from tip shifted points

Figure 6.37: Pressure jump on the TE for various positions of the last collocation point. $*$ symbols correspond to positions of collocation points.



(a) L^1 (b) L^∞

Figure 6.38: Norms of the pressure jump on the TE for various positions of the last collocation point.

Chapter 7

Conclusions and Future Steps

In this work a new Boundary Element Method (BEM) for steady lifting flows, based on Isogeometric Analysis (IGA) was presented. T-splines were used for the representation of all involved geometries and the approximation of the associated field (trace of the velocity potential on the wing). Adopting IGA enables the enforcement of Kutta condition (zero-pressure jump) exactly on the trailing edge (TE) as well as revealing the singular behaviour of the solution at the TE tip and the wing cap.

The method was applied for three different test cases:

- A rectangular wing with a NACA0012 profile.
- A swept wing with a sweep angle of 20 degrees and a NACA0012 profile.
- A cambered rectangular wing with a NACA4412 profile.

The calculated pressure coefficients show good agreement with experiments and superior behaviour when compared to a low order panel method in terms of the required DoFs (Degrees of Freedom) for a given level of accuracy. The iterative scheme, that was applied for the solution of the resulting quadratic system, exhibits fast convergence (5-6 iterations) for different levels of refinement. Furthermore, a numerical investigation was carried out regarding the effect of the tip singularity on Kutta condition for different refinement levels and different locations of the collocation points along the trailing edge. This study revealed that, as the number of DoFs increases, the unboundedness of pressure near the tip becomes evident. Moreover, the zero pressure jump along the trailing edge shows a low L^1 error, especially by slightly shifting the last collocation point from the tip. Note that, since T-splines were used, local refinement has been employed in the trailing edge region in an effort to investigate the behaviour of Kutta condition without introducing redundant DoFs.

This work is, to the author's best knowledge, the first attempt in employing Isogeometric Analysis for the problem of three dimensional potential lifting flows. The main benefits of IGA that have been discussed in previous works in other applications are also present here:

- Higher accuracy for fewer DoFs when compared to traditional BEM methods is achieved.
- The original geometry of the CAD model is maintained for analysis and the refinement process is as simple as using knot insertion on the underlying spline basis. No re-meshing is ever required.

There are also benefits that are specific to this application:

- The projection of ϕ on the spline space of the underlying geometry allows the enforcement of Kutta condition exactly on the trailing edge.
- The use of Greville abscissae as collocation points creates a well-conditioned matrix and no numerical instabilities occur. Not even near the trailing edge which has been a region that requires special treatment in traditional BEM (see for example [51].

Lastly, it's worth noting that the choice of T-splines for the underlying wing geometry offered some significant advantages:

- Geometric modelling of the wing was achieved with much fewer control points than using NURBS. For example, the cap surface of the NACA4412 wing was created with a single T-spline surface with only 83 control points. The same model would consist of 459 control points and 3 NURBS patches if NURBS were used.
- Local refinement was applied without introducing too many redundant DoFs. In this work, it was mainly employed to investigate the behaviour of Kutta condition on the trailing edge.

For all the above reasons, the developed method tightly integrates CAD and analysis. A next reasonable step would be to use the solver within a shape optimisation loop for wing geometries. Due to the nature of IGA, no re-meshing would be required for each new shape that is generated. This is expected to make the resulting tool highly efficient.

Another thing to consider is that the aforementioned method was based on the assumption of an a-priori known plane wake geometry, which served as a good approximation for the cases that were investigated (wings in uniform flows). A potential future step would be to extend this method to treat propeller flows. The main challenge there is that a-priori wake approximations do not work equally well due to the nature of the flow; trailing wakes shed from the blades are at a small distance from them and can affect their pressure distributions. This is especially important when the advance ratio is quite low which means that one of the blades could operate within the flow of the wake of another blade, as stated for example in [92].

This would call for the development of an IGA-based wake alignment scheme, similar to the ones described in Chapter 2. The scheme could use an initial approximation of the wake surface (i.e. a helical surfaces based on the propeller's geometric characteristics) and then correct it at each iteration step based on the solution field of the previous step. The iterative process would stop when the wake geometry converges. More details of the implementation of such a scheme in the IGA setting are not known right now. However, it would be safe to assumed that the geometric flexibility offered by T-splines could prove pivotal in representing complex wake surfaces. Moreover, choosing Greville's abscissae as points where the wake alignment condition is enforced would be a good starting point.

Lastly, it would be interesting to extend this method to include more complex configurations, such as multiple wings that interact with each other, wing-fuselage configurations and time-dependent problems.

Bibliography

[1] A.I. Ginnis, K.V. Kostas, C.G. Politis, P.D. Kaklis, K.A. Belibassakis, Th.P. Gerostathis, M.A. Scott, T.J.R. Hughes. Isogeometric boundary-element analysis for the wave-resistance problem using T-splines. *Comput. Methods Appl. Mech. Engrg.*, 279:425–439, 2014.

[2] B.A. Finlayson. The Method of Weighted Residuals and Variational Principles. *Academic Press, New York*, 1972.

[3] M. Jockovic , G. Radenkovi , M. Nefovska-Danilovi , M. Baitsch. Convergence of different wake alignment methods in a panel code for steady-state flows. *Applied Mathematical Modelling*, 71:152–172, 2019.

[4] B.H.Nguyen , H.D.Tran , C.Anitescu , X.Zhuang , T.Rabczuk. An isogeometric symmetric Galerkin boundary element method for two-dimensional crack problems. *Comput. Methods Appl. Mech. Engrg.*, 306:252–275, 2016.

[5] C. Giannelli, B. Juettler, H. Speleers. THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29:485–498, 2012.

[6] C.G. Politis , A.I. Ginnis , P.D. Kaklis, K.A. Belibassakis , C. Feurer. An isogeometric BEM for exterior potential-flow problems in the plane. *SIAM/ACM Joint Conference on Geometric and Physical Modeling, SPM09*, pages 349–354, 2009.

[7] C.G. Politis, A. Papagiannopoulos, K.A. Belibassakis, P.D. Kaklis, K.V. Kostas, A.I. Ginnis, T.P. Gerostathis. An Isogeometric BEM for Exterior Potential-Flow

Problems Around Lifting Bodies. *Proceedings, 11th World Congress on Computational Mechanics (WCCM XI)*, 2014.

[8] D. Forsey, R. Bartels. Hierarchical B-Spline Refinement. *SIGGRAPH: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 205–212, 1988.

[9] D. Greeley, J. Kerwin. Numerical Methods for Propeller Design and Analysis in Steady Flow. *Trans SNAME, vol 90*, 1982.

[10] D. Lesnic, L. Elliott, D.B. Ingham. Treatment of singularities in exterior fluid domains with corners using the boundary element method. *Computers and Fluids*, 23:817–827, 1994.

[11] E. Cervera, J. Trevelyan. Evolutionary structural optimisation based on boundary representation of NURBS: Part I: 2D algorithms. *Computers and Structures*, 83:1902–1916, 2005.

[12] E. Cervera, J. Trevelyan. Evolutionary structural optimisation based on boundary representation of NURBS: Part II: 3D algorithms. *Computers and Structures*, 83:1917–1929, 2005.

[13] E. J. Evans, M. A. Scott, X. Li, D. C.Thomas. Hierarchical analysis-suitable T-splines: Formulation, Bezier extraction amd application as an adaptive basis for isogeometric analysis. *Comput. Methods Appl. Mech. Engrg.*, 284:1–20, 2014.

[14] E.O. Suciu, L. Morino. A nonlinear finite-element analysis of wings in steady incompressible flows with wake roll-up. *14th Aerospace Sciences Meeting*, 1976.

[15] F. Auricchio , L.B. Da Veiga , T.J.R. Hughes , A. Reali , G. Sangalli. Isogeometric collocation methods. *Math. Models Methods Appl. Sci.*, 20:2075–2107, 2010.

[16] F. Auricchio , L.B. Da Veiga , T.J.R. Hughes , A. Reali , G. Sangalli. Isogeometric collocation for elastostatics and explicit dynamics. *CES,Report 12-07*, 2011.

[17] G. Farin. *Curves and Surfaces for Computer-Aided Geometric Design.* Academic press, 1990.

[18] F.T. Johnson, E.N. Tinoco, P. Lu, M.A. Epton. Three-Dimensional Flow over Wings with Leading-Edge Vortex separation. *AIAA Journal*, 18, 1987.

[19] G. Beer, I. Smith, C. Duenser. *The Boundary Element Method with Programming.* Springer Wien New York, 2008.

[20] G.-D. Kim, B.-K. Ahn, J.-H. Kim, C.-S. Lee. Improved Hydrodynamic Analysis of Marine Propellers Using a B-Spline-Based Higher-Order Panel Method. *J Mar Sci Technol*, 20:670–678, 2015.

[21] G.K. Politis. Simulation of unsteady motion of a propeller in a fluid including free wake modeling. *Engineering Analysis with Boundary Elements*, 28:633–653, 2004.

[22] G.K. Politis. Application of a BEM time stepping algorithm in understanding complex unsteady propulsion hydrodynamic phenomena. *Ocean Engineering*, 38:699–711, 2011.

[23] G.K. Politis. Unsteady Wake Rollup Modeling Using a Mollifier Based Filtering Technique. *Development and Applications of Oceanic Engineering DAOE*, 5, 2016.

[24] H. Glauert. The elements of airfoil and airscrew theory. *Cambridge University Press*, 1926.

[25] H. Ashley, S. Widnall , M.T. Landahl. New Directions in Lifting Surface Theory. *AIAA Journal*, 3:3–16, 1965.

[26] H. Lee, S.A. Kinnas. Application of a boundary element method in the prediction of unsteady blade sheet and developed tip vortex cavitation on marine propellers. *Journal of Ship Research*, 48:15–30, 2004.

[27] T. Hoshino. Hydrodynamic Analysis of Propellers in Steady Flow Using a Surface Panel Method. *Spring Meeting of Naval Architects of Japan*, 1989.

[28] H.W.M. Hoeijmakers. Methods for Numerical Simulation of Leading Edge Vortex Flow. *Springer Verlag*, 1985.

[29] J. Hess, A.M.O. Smith. Calculation of non-lifting potential flow about arbitrary three-dimensional bodies. *Journal of Ship Research*, 8, 1964.

[30] J. Kiednl , Y. Bazilevs , M.C. Hsu , R. Wuechner , K.U. Bletzinger. The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches. *Comput. Methods Appl. Mech. Engrg*, 199:2403–2416, 2010.

[31] J.A. Cottrell, A. Reali, Y. Bazilevs. Isogeometric analysis of structural vibrations. *Comput. Methods Appl. Mech. Engrg.*, 195:5257–5296, 2006.

[32] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs. Isogeometric Analysis: Toward Integration of CAD and FEA. *Wiley, Chichester*, 2009.

[33] J.E. Kerwin, C.S. Lee. Prediction of Steady and Unsteady Marine Propeller Performance by Numerical Lifting Surface Theory. *TRANS. SNAME*, 86:1–30, 1978.

[34] J.E. Kerwin, J.B. Hadler. The principles of naval architecture: Propulsion. *The Society of Naval Architects and Marine Engineers*, 2010.

[35] J.E. Kerwin, S.A. Kinnas, J. Lee, W. Shih. A Surface Panel Method for the Hydrodynamic Analysis of Ducted Propellers. *TRANS. SNAME*, 95, 1987.

[36] J.F.C Telles. A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary element integrals. *Int. J. Numer. Methods.*, 24:959–973, 1987.

[37] J.F.C Telles, R.F. Oliveira. Third degree polynomial transformation for boundary element integrals: further improvements. *Eng. Anal. Boundary Elem.*, 13:135–141, 1994.

[38] J.L. Hess. Calculation of Potential Flow about Arbitrary Three-Dimensional Bodies. *Report No. MDC J5679-01*, 1972.

[39] J.L. Hess. Review of integral-equation techniques for solving potential flow problems with emphasis on the surface source method. *Comput. Methods Appl. Mech. Engrg.*, 5:145–196, 1975.

[40] J.T. Lee. A Potential Based Panel Method for the Analysis of Marine Propellers in Steady Flow . *PhD Thesis, Department of Ocean Engineering, MIT*, 1987.

[41] K. Kostas , A. Ginnis , C. Politis , P. Kaklis. Shape-optimization of 2D hydrofoils using an Isogeometric BEM solver. *Computer-Aided Design*, 82:79–87, 2017.

[42] K.A. Belibasakis, G.K. Politis. A Boundary Integral Equation Formulation of the Neumann Problem for a Vector Field in R^3 with Application to Potential Lifting Flows. *Eng. Anal. Boundary Elem.*, 16:5–17, 1995.

[43] K.A. Belibasakis, G.K. Politis. A Non-Linear Velocity Based Boundary Element Method for the Analysis of Marine Propellers in Unsteady Flow. *International Shipbuilding Progress*, 45:93–133, 1998.

[44] K.A. Belibassakis, Th.P. Gerostathis, K.V. Kostas, C.G. Politis, P.D. Kaklis, A.I. Ginnis, C. Feurer. A BEM-isogeometric method for the ship wave-resistance problem. *Ocean Engineering*, 60:53–67, 2013.

[45] K.V. Kostas, M.M. Fyrillas , C.G. Politis , A.I. Ginnis , P.D. Kaklis. Shape optimization of conductive-media interfaces using an IGA-BEM solver. *Comput. Methods Appl. Mech. Engrg. 340 600614*, 2018.

[46] K.V. Kostas, A. Amiralin, S. Sagimbayev, T. Massalov, Y. Kalel, and C.G. Politis. Parametric model for the reconstruction and representation of hydrofoils and airfoils. *Ocean Engineering*, 199:107020, 2020.

[47] M.M. Fyrillas , T. Leontiou , K.V. Kostas. Optimum interfaces that maximize the heat transfer rate between two conforming conductive media. *International Journal of Thermal Sciences*, 121:381–389, 2017.

[48] K.V. Kostas, A.I. Ginnis, C.G. Politis, P.D. Kaklis. Ship-hull shape optimization with a T-spline based BEM-isogeometric solver. *Comput. Methods Appl. Mech. Engrg.*, 284:611–622, 2015.

[49] L. Morino. Boundary Integral Equations in Aerodynamics. *Appl. Mech. Rev. Aug*, 46:445–466, 1993.

[50] L. Morino, C. Kuo. Subsonic potential aerodynamics for complex configurations: A general theory. *AIAA journal*, 12, 1974.

[51] L. Morino, G. Bernardini. Singularities in BIEs for the Laplace equation; Joukowski trailing-edge conjecture revisited. *Engineering Analysis with Boundary Elements*, 25:805–818, 2001.

[52] A.-T. Luu, N.-I. Kim , J. Lee. 2D vibration analysis of circular arches with constant double symmetric cross-sections using isogeometric approach. *KSCE Journal of Civil Engineering*, 21:2751–2763, 2017.

[53] L.P. Yip, G.L Shubert. Pressure Distributions on a 1- by 3-Meter Semispan Wing at Sweep Angles From 0° to 40° in Subsonic Flow. *NASA TN D-8307*, 1976.

[54] M. Abdel-Maksoud, H.-J Heinke. Scale effects on ducted propellers. *Proceedings, The 24 th Symposium on Naval Hydrodynamics, July 813, Fukuoka, Japan*, 2002.

[55] M. Scott. T-splines as a Design-Through-Analysis Technology. *PhD Thesis, The University of Texas at Austin*, 2011.

[56] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sedeberg, T.J.R. Hughes. Isogeometric finite element data structures based on Bezier extraction of T-splines. *ICES report 10-45*, 2011.

[57] M.A. Scott, R.N. Simpson, J.A. Evans, S. Lipton, S.P.A. Bordas, T.J.R. Hughes, T.W. Sederberg. Isogeometric boundary element analysis using unstructured T-splines. *Comput. Method Appl. Mech. Engrg.*, 254:197–221, 2013.

[58] M.A. Scott, T.J.R. Hughes, T.W. Sederberg, M.T. Sederberg. An integrated approach to engineering design and analysis using the autodesk t-spline plugin for rhino 3d. *ICES Report 14-33*, 2014.

[59] M.A. Scott, X. Li, T.W. Sederberg, T.J.R Hughes. Local refinement of analysis-suitable T-splines. *Comput. Methods Appl. Mech. Engrg.*, 213:206–222, 2012.

[60] M.J. Borden , M.A. Scott, J.A. Evans , T.J.R. Hughes. Isogeometric finite element data structures based on Bzier extraction of NURBS. *Int. J. Numer. Meth. Engng 87:1547*, 2010.

[61] M.T. Landahl, V.J.E. Stark. Numerical Lifting-Surface Theory - Problems and Progress. *AIAA Journal,*, 3:2049–2060, 1968.

[62] M.M. Munk. Elements of the wing section theory and of the wing theory. *NACA Rept. 191*, 1924.

[63] Z. An, T. Yu , T.Q. Bui , C. Wang N.A. Trinh. Implementation of isogeometric boundary element method for 2-D steady heat transfer analysis. *Advances in Engineering Software*, 116:36–49, 2018.

[64] J.N. Newman. Marine hydrodynamics. *MIT Press*, 1977.

[65] N.T. Frink, S.Z. Pirzadeh, P.C Parikh, M.J. Pandya, M.K. Bhat. The NASA tetrahedral unstructured software system (TetrUSS). *The Aeronautical Journal*, 104:491–499, 2000.

[66] P. Hunter, A. Pullan. *FEM/BEM notes.* Department of Engineering Science, University of Auckland, 2001.

[67] P. Phung-Van, M. Abdel-Wahab, K.M. Liew, S.P.A Bordas , H. Nguyen-Xuan. Isogeometric analysis of functionally graded carbon nanotube-reinforced composite plates using higher-order shear deformation theory. *Composite Structures*, 123:137–149, 2015.

[68] P.R. Johnston, D. Elliott. A generalisation of Telles method for evaluating weakly singular boundary element integrals. *Journal of Computational and Applied Mathematics*, 131:223–241, 1999.

[69] L. Prandtl. Applications of modern hydrodynamics to aeronautics. *NACA Rept. 116*, 1921.

[70] K. Li, X. Qian. Isogeometric analysis and shape optimization via boundary integral. *Computer Aided Design*, 43:14271437, 2011.

[71] R.N. Simpson , Z. Liu , R. Vasquez , J.A. Evans. An isogeometric boundary element method for electromagnetic scattering with compatible b-spline discretizations. *Journal of Computational Physics*, 362:264–289, 2018.

[72] R.N. Simpson, S.P.A. Bordas, J. Trevelyan , T. Rabczuk. A two-dimensional Isogeometric Boundary Element Method for elastostatic analysis. *Comput. Method Appl. Mech. Engrg. 209-212:87-100*, 2012.

[73] S. Gaggero, S. Brizzolara. Exact Modeling of Trailing vorticity in Panel Method for Marine Propeller. *2nd International Conference on Marine Research and Transportation*, 2007.

[74] S. Pyo. Numerical Modeling of Propeller Tip Flows with Wake Sheet Roll-up in Three Dimensions . *PhD Thesis, Department of Ocean Engineering, MIT*, 1995.

[75] S. Pyo, S.A. Kinnas. The Flow Adapted Grid (FLAG) Applied to the Analysis of Propeller Tip Flows. *Propeller/Shafting '94 Symposium*, 1994.

[76] H. Sohngen. Bestimmung der Auftriebsverteilung fur beliebige instationare Bewegungen (Ebenes Problem). *Luftfahrtforschung, Bd 17, Nr 11 und 12*, 1940.

[77] H. Sohngen. T Zur Theorie der endlichen Hilbert-Transformation,. *Math Zeitschr, Bd 60*, 1954.

[78] Y. Wang, M. Abdel-Maksoud, B. Song. Convergence of different wake alignment methods in a panel code for steady-state flows. *Journal of Marine Science and Technology*, 21:567–578, 2016.

[79] Y. Wang, M. Abdel-Maksoud, B. Song. A fast method to realize the pressure Kutta condition in boundary element method for lifting bodies. *Ocean Engineering*, 130:398–406, 2017.

[80] T. Dokken, T. Lyche, K. Pettersen. Polynomial splines over locally refined box-partitions. *, Computer Aided Geometric Design*, 30:331–356, 2013.

[81] T. van Opstal, E. Fonn, R. Holdahl, T. Kvamsdal, A.M. Kvarving, K.M. Mathisen, K. Nordanger, K.M. Okstad, A. Rasheed, M. Tabib. Isogeometric Methods for CFD and FSI-Simulation of Flow around Turbine Blades. *Energy Procedia*, 80:442–449, 2015.

[82] T.J.R. Hughes,J.A. Cottrell,Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Engrg.*, 194:4135–4195, 2005.

[83] T.W. Sederberg, J. Zheng, A. Bakenov, A. Nasri. T-splines and T-NURCCs. *ACM Transactions on Graphics*, 22:477–484, 2003.

[84] V.A. Kondrat'ev, O.A. Oleinik. Boundary-value problems for partial differential equations in non-smooth domains. *Russian Mathematical Surveys*, 38:3–76, 1983.

[85] W. Geissler. Nonlinear Unsteady Potential Flow Calculations for Three-Dimensional Oscillating Wings. *AIAA Journal*, 16, 1978.

[86] X. Li, J. Zheng, T.W. Sederberg, T.J.R Hughes, M.A. Scott. On Linear Independence of T-spline Blending Functions. *Computer Aided Geometric Design*, 29:63–76, 2012.

[87] X. Li, M. A. Scott. Analysis-suitable T-splines: characterization, refineability and approximation. *Mathematical Models and Methods in Applied Science, 24(06):1141-1164*, 2014.

[88] Y. Bazilevs , C. Michler , V.M Calo , T.J.R. Hughes. Isogeometric variational multiscale modelling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes. *Comput. Methods Appl. Mech. Engrg.*, 199(13-16):780–790, 2010.

[89] Y. Bazilevs , I. Akkerman. Large eddy simulation of turbulent TaylorCouette flow using isogeometric analysis and residual-based variational multiscale method. *Journal of Computational Physics*, 229:3402–3414, 2010.

[90] Y. Bazilevs , J.R. Gohean , T.J.R. Hughes , R.D. Moser , Y. Zhang. Patient-specific isogeometric fluidstructure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device. *Comput. Methods Appl. Mech. Engrg.*, 198(45-46):3534–3550, 2009.

[91] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, T.W. Sederberg. Isogeometric analysis using T-splines. *Comput. Methods Appl. Mech. Engrg.*, 199(5-8):229–263, 2010.

[92] Y. Tian, S.A. Kinnas. A wake model for the prediction of propeller performance at low advance ratios. *Int J Rotating Mach*, 2012:1–11, 2012.

[93] Z. ling, Y. Sasaki, M. Takahashi. Analysis of Three-Dimensional Flow Around Marine Propeller by Direct Formulation of Boundary Element Method. *Journal of the Society of Naval Architects of Japan, Vol. 157*, 1985.

[94] Z. Liu, M. Majeed, F. Cirak, R.N. Simpson. Isogeometric FEM-BEM coupled structural-acoustic analysis of shells using subdivision surfaces. *International Journal for Numerical Methods in Engineering, 00:1-29*, 2016.

Appendices

Appendix A

Fundamentals of Vortex Sheets

The wake surface is regarded as a force-free vortex sheet that emanates from the trailing edge and extends to infinity. This appendix includes an introduction to vortex sheets and some interesting properties of them are presented here. The proofs of why the potential jump on the wake varies only along its span-wise direction and why its value is determined by the value on the trailing edge of the wing (see Chapter 3) are also produced.

1 Force-Free Vortex Sheet

A vortex sheet is a thin surface with a distribution of vorticity. Let a vortex sheet exist inside an irrotational, incompressible and inviscid fluid field. Then for a point P on the sheet the velocities on the upper and lower parts of the sheet$(\mathbf{V_u}, \mathbf{V_l})$ must be tangent to the sheet surface but not necessarily equal in either magnitude or direction. If $\mathbf{V_d}$ is the difference between the velocities and $\mathbf{V_m}$ is the mean velocity then the following relations occur:

$$\mathbf{V_m} = \frac{1}{2}(\mathbf{V_u} + \mathbf{V_l})$$
$$\mathbf{V_d} = \frac{1}{2}(\mathbf{V_u} - \mathbf{V_l})$$

When the difference between the upper and lower velocities is not equal to zero a vortex sheet exists with strength [34] given by:

$$\gamma = 2 \cdot [\mathbf{n} \times \mathbf{V_d}] \tag{A.1}$$

where **n** is the vector normal to the vortex sheet. The pressure jump between the two sides of the vortex surface is given by Bernoulli's law:

$$\Delta p = \frac{1}{2}\rho({\mathbf{V_u}}^2 - {\mathbf{V_l}}^2) \tag{A.2}$$

If δ is the angle between the mean velocity and the vorticity vector then by using law of cosines (A.2) becomes:

$$\Delta p = \frac{1}{2}\rho \mathbf{V_m} \gamma sin\delta \tag{A.3}$$

The vorticity vector can be analysed into two components along and normal to the mean flow, $\gamma_b = \gamma sin\delta, \gamma_f = \gamma cos\delta$ which are called *bound* and *free* vorticity respectively. When the mean velocity is parallel to the vorticity vector then the pressure jump becomes zero (Figure A.1) and only free vorticity exists. In this event the vortex sheet is called *force-free.*

Now let an airfoil be represented as a vortex sheet and its wake as a force-free vortex sheet. This is shown in Figure A.2. The vorticity vectors on the foil and wake can be also seen in Figure A.2. Only bound vorticity exists on the foil and only free vorticity on the wake. According to Kelvin's theorem the circulation around a closed contour inside a potential flow field is equal to zero. Following the approach of [34] and using the contour of Figure A.2:

$$\Gamma_1(s_2 = s_W) = \int_{s_L}^{s_T} \gamma_b(s_1; s_2 = s_W) ds_1 \tag{A.4}$$

and for the wake:

$$\Gamma_2(s_2 = s_W) = -\int_{s_W}^{s_E} \gamma_f(s_2) ds_2 \tag{A.5}$$

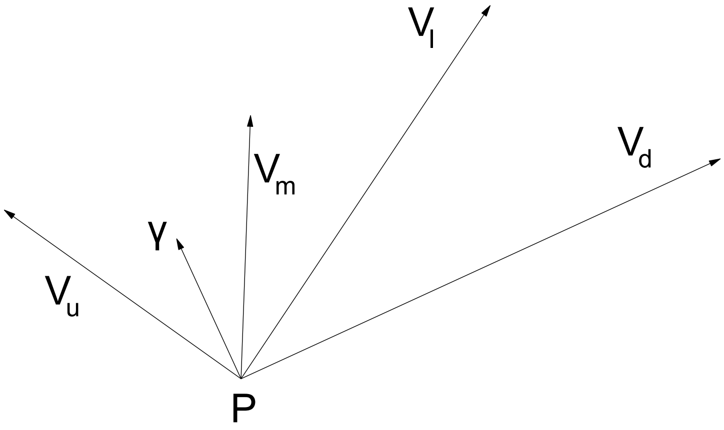
where s_1 is a coordinate system parallel to the mean flow, s_2 is normal to s_1, s_W is the intersection between s_1 and s_2 and s_L, s_T and s_E correspond to the leading edge, trailing edge and the boundary of the wake respectively.

The part of the contour which consists of the two lines parallel to the wake does not create any circulation since both lines are tangent to the vorticity vector. Then Kelvin's theorem gives:

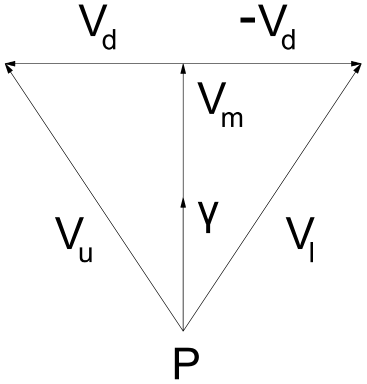
$$\Gamma_1 + \Gamma_2 = 0 \Rightarrow \Gamma_1 = \int_{s_W}^{s_E} \gamma_f(s_2) ds_2 \tag{A.6}$$

2 Circulation and Potential Jump

The wake of a wing can be regarded as a force-free vortex sheet surface. The strength of the free vorticity γ_f is connected to the circulation around the wing. It is proven [34] that:



(a) General case, $V_m \nparallel \gamma$



(b) Force-free condition, $V_m \parallel \gamma$

Figure A.1: Velocities on a force-free vortex sheet

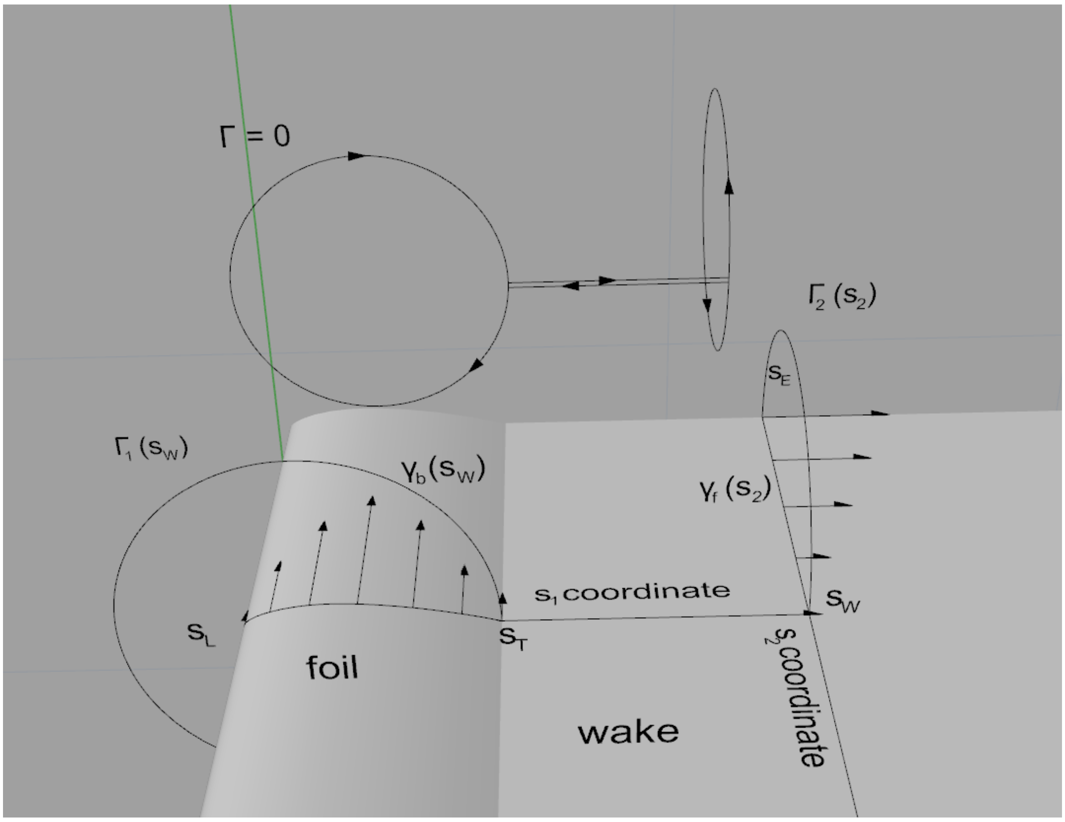


Figure A.2: 3D Foil and Wake Vortex Sheets

$$\frac{d\Gamma_1}{ds_W} = \gamma_f(s_W) \tag{A.7}$$

where s_W is the spanwise curvilinear coordinate (which is normal to the chordwise coordinate and parallel to the mean flow) on the wake and $\Gamma(s_W)$ is the circulation around the corresponding airfoil. But since the wake is force free:

$$\frac{d\Gamma_1}{ds_W} = 2V_d = V_{u,s_W} - V_{l,s_W} \tag{A.8}$$

Under the assumption that the flow is irrotational, incompressible and inviscid the fluid velocity can be written as a function of the velocity potential ϕ:

$$\mathbf{V} = \Big(\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y}, \frac{\partial\phi}{\partial z}\Big) \tag{A.9}$$

Then by substituting the velocity values of (A.8) with respect to the potential (A.8) becomes:

$$\frac{d\Gamma_1}{ds_W} = \frac{\partial\phi_u}{\partial s_W} - \frac{\partial\phi_l}{\partial s_W} \Rightarrow \Gamma(s_W) = \Delta\phi(s_W) \tag{A.10}$$

where $\Delta\phi(s_W) = \phi_u - \phi_l$ and ϕ_u, ϕ_l are the velocity potential values on the upper and lower sides of the wake respectively. This means that the potential jump $\Delta\phi(s_W)$ is constant along a line of constant mean flow (mean streamline) and equal to the circulation around the corresponding hydrofoil and varies only along the span of the wake.

Appendix B

Evaluating Weakly Singular Integrals

The kernel of (5.4) is weakly singular and special treatment of the integrals is required when $\mathbf{P}_j$ and $\mathbf{Q}$ lie significantly close to each other. In this appendix the approach to tackle this issue is presented.

1 Types of Integrals

Three different types of integrals may be considered depending on the distance between these two points:

- Far-field case: Let $\mathbf{Q}$ lie on a Bézier patch $\mathbf{x}^e$ with $conv_{\mathbf{x}^e}$ denoting the convex hull of its control points . When $\mathbf{P}_k$ and $\mathbf{Q}$ do not lie on the same surface element and the Euclidean distance $d(\mathbf{P}_k, conv_{\mathbf{x}^e})$ between $\mathbf{P}_k$ and $conv_{\mathbf{x}^e}$ is greater than two times the diagonal of $conv_{\mathbf{x}^e}$, i.e., $d(\mathbf{P}_k, conv_{\mathbf{x}^e}) > 2diag(conv_{\mathbf{x}^e})$, Gauss - Kronrod quadrature with 15 quadrature points is applied.
- Near-field case: When $\mathbf{P}_k$ and $\mathbf{Q}$ do not lie on the same surface element but

$d(\mathbf{P}_k, conv_{\mathbf{x}^e}) < 2diag(conv_{\mathbf{x}^e})$, Telles' transformation is applied as introduced in [36], [37], with the preimage of $\tilde{\boldsymbol{\xi}}(\mathbf{P}_k)$ of $\mathbf{P}_k$ regarded as the singular point of the transformation. Then Gauss - Kronrod quadrature with 31 quadrature points is applied on the transformed integral afterwards.

- In-field case: When $\mathbf{P}_k$ and $\mathbf{Q}$ lie on the same surface element the corresponding parametric domain is partitioned into 4 sub-domains from I to IV as demonstrated in Fig. B.1. Telles transformation is applied again for each sub-domain with $\tilde{\boldsymbol{\xi}}(\mathbf{P}_k)$ being the singular point of the transformation for each of them. Afterwards, Gauss - Kronrod quadrature with 31 quadrature points is applied on each transformed sub-domain.

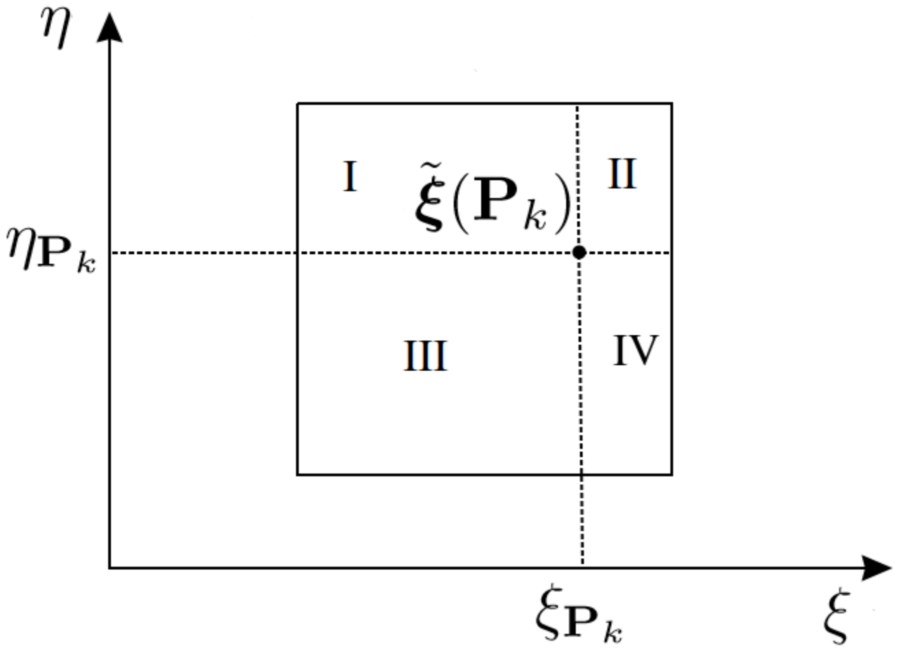


Figure B.1: Domain partition for the in field case

2 Telles' Transformation

A linear transformation of variables is applied on the in field and near field cases:

$$\int_{\eta_{WB}}^{\eta_B}\int_{\xi_{WB}}^{\xi_B}\mathbf{f}(\xi,\eta)d\xi d\eta=\int_{-1}^{1}\int_{-1}^{1}\mathbf{f}_*(t_1,t_2)dt_1dt_2,\quad \mathbf{f}_*=\mid d\xi/d\mathbf{t}\mid\mathbf{f} \tag{B.1}$$

Then, Telles' non-linear Transformation is applied for each variable of the resulting integrals:

$$\xi(s_1)=\alpha_1 s_1^3+b_1 s_1^2+c_1 s_1+d_1, \tag{B.2}$$

$$\xi(s_2)=\alpha_2 s_2^3+b_2 s_2^2+c_2 s_2+d_2, \tag{B.3}$$

where $\alpha_k=Q_k$, $\quad b_k=-3s_k^PQ_k$, $\quad c_k=3(s_k^P)^2Q_k$, $\quad d_k=-b_k$, $\quad Q_k^{-1}=1+3(s_k^P)^2$ for $k=1,2$ and the point (s_1^P,s_2^P) corresponds to the nearly singular point P.

Consequently, the right hand side of equation (B.1) becomes:

$$\int_{-1}^{1}\int_{-1}^{1}\mathbf{f}_*(\xi_1(s_1),\xi_2(s_2))(9\alpha_1\alpha_2(s_1-s_1^P)^2(s_2-s_2^P)^2)ds_1ds_2 \tag{B.4}$$

Then the integral of (B.4) can be calculated by using standard quadrature techniques.

Telles' transformation has a lumping effect on integration points approaching the singularity and produces superlinear convergence at the very least. However, according to [68], it works best when the singular point lies on the endpoints or midpoint of the integration interval while significant errors may be encountered when the singular point lies anywhere else. In order to counter this the interval of the in field case is partitioned in the way described in the previous section to ensure that P always lies on the endpoints of the four sub-intervals shown in Figure (B.1)

Appendix C

IGA-BEM Solver Software

This appendix serves as a user manual for the in-house IGA-BEM solver used in this work. The solver was developed in a C++11 programming environment. It can be run in parallel processes utilising a Message Passing Iinterface (MPI) standard and occupies all the available memory of the machine it is run on by employing shared memory matrices.

The following libraries are required as dependencies in order to compile the solver:

- Open MPI: *"The Open MPI Project is an open source Message Passing Interface implementation that is developed and maintained by a consortium of academic, research, and industry partners."*, (from https://www.open-mpi.org/)
- Global Arrays by Pacific Nothwest National Laboratory: *"Global Arrays (GA) is a Partitioned Global Address Space (PGAS) programming model. It provides primitives for one-sided communication (Get, Put, Accumulate) and Atomic Operations (read increment). It supports blocking and non-blocking primtives, and supports location consistency.*,
  (from https://hpc.pnl.gov/globalarrays/)
- GoTools by SINTEF: *"GoTools is a collection of C++ libraries related to geome-*

try targeting Computer Aided Design (CAD), Isogeometric Analysis (IgA) and big data approximation and analysis." , (from https://www.sintef.no/en/software/gotools/)

- SINTEF Spline Library (SISL) by SINTEF: *"SISL is a comprehensive NURBS library for the modeling and interrogation of curves and surfaces. It is implemented in C and has been under continuous development over three decades."*, (from https://www.sintef.no/en/software/sisl/)

Once compiled the solver can run by providing the following input files:

- A wing.iga file that includes all the required information about the main wing T-spline surface.
- A cap.iga file that includes all the required information about the wing cap T-spline surface.
- A wake.iga file that includes all the required information about the wake T-spline surface.
- A wake_ids.dat file that contains the Bézier element ids of the wake in matrix form.
- A input.dat file that includes the fluid velocity under investigation.
- A folder location where all the output files will be created.

The .iga file format is created by the T-spline plug-in for Rhinoceros 3D and contains control point and Bézier extraction information required for creating a T-spline class in the solver. The structure of .iga files may be found in [58]. The wake_ids.dat file is required in order to match the wake elements with their corresponding ones on the trailing edge of the wing.

The user can start the solver from the command window by opening the directory of the solver executable and using the following line:

```
./iga_bem_solver TSPLINE wing.iga wake.iga cap.iga wake_ids.dat input.dat /path-to-results
```

or if the user wishes the program to run in multiple cores:

```
mpirun -np x iga_bem_solver TSPLINE wing.iga wake.iga cap.iga wake_ids.dat input.dat /path-to-results
```

where x is the number of parallel processes. The application will then run in batch mode until it solves the problem and will produce the following output files:

- A.dat: Contains the A matrix of the non-linear system.
- b.dat: Contains the right hand side vector of the non-linear system.
- phi_on_surface.csv: Contains the field value of ϕ for a set of output points with x,y,z coordinates on the main wing surface
- phi_on_surface_cap.csv: Contains the field value of ϕ for a set of output points with x,y,z coordinates on the cap surface
- phiCPs.dat: Contains the solution of the non-linear system (values of all the initially unknown coefficients ϕ_i and $\Delta\phi_i$) including all steps of the Newton-Raphson iterative scheme
- pressure.csv: Contains the field value of pressure coefficient C_p for a set of output points with x,y,z coordinates on the main wing surface
- pressure_cap.csv: Contains the field value of pressure coefficient C_p for a set of output points with x,y,z coordinates on the cap surface
- velocity_on_body.csv: Contains the velocity components V_x, V_y, V_z for a set of output points with x,y,z coordinates on the main wing surface
- velocity_on_cap.csv: Contains the velocity components V_x, V_y, V_z for a set of output points with x,y,z coordinates on the cap surface

The number of output points is equal to 100 points per Bézier element for each surface. The output .csv files can be used to visualise the resulting field quantities. In this work *ParaView* (https://www.paraview.org/) is used for creating all the colour maps.