# University of Southampton

Faculty of Engineering and Physical Sciences

School of Engineering

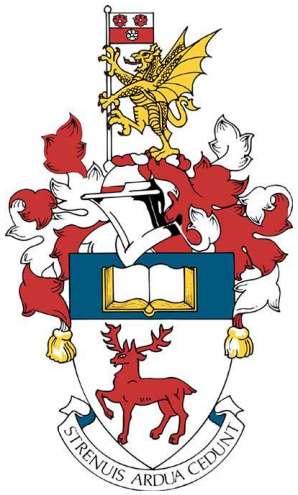
**Adjoint System-Based Optimisation of Hydrofoils Using Algorithmic Differentiation**

by

**Rafael Andreas Maximilian Tannenberg**

Thesis for the degree of Doctor of Philosophy

June 2025



# University of Southampton

## Abstract

Faculty of Engineering and Physical Sciences

School of Engineering

Doctor of Philosophy

Adjoint System-Based Optimisation of Hydrofoils Using Algorithmic Differentiation

by

Rafael Andreas Maximilian Tannenberg

Hydrofoils are the biggest performance differentiators on modern racing yachts. They significantly affect the speed, the state and the trim of the yacht system. This requires detailed optimisation and the modelling of the whole system during the optimisation. However, due to the high computational cost, hydrofoils are still optimised in isolation of the yacht system with most studies using only a small number of design variables. This introduces inaccuracies and leaves considerable potential inaccessible.

Therefore, a stationary physics model of an entire yacht is developed. It includes an advanced lifting line method for hydrofoil force prediction and a detailed parametric model of the hydrofoil with 68 design variables. As this is too computationally expensive to optimise with conventional strategies, the physics model is integrated into a gradient-based optimisation routine, where the gradient is computed with the adjoint method. The adjoint method can compute the gradient at small cost, independent of the number of design variables, and is the key to detailed design. The adjoint method is only applied to the bottleneck of the physics model using algorithmic differentiation. The remainder of the model is differentiated with the easier to implement finite difference method. The combined gradients are provided to an optimisation algorithm.

The framework is used to optimise the hydrofoils of an AC75 America's Cup yacht. It is validated with a parametric study and literature. A single-condition optimisation of the hydrofoil is performed for upwind $V_{\mathrm{MG}}$. The optimum foil shows a maximum anhedral angle, a low chord and a high span together with a detailed twist distribution that leads to the desired elliptical lift distribution. Aft sweep, a convex leading edge and raked wing tips are used to further increase performance. The optimised foil is 1.58 kts or 6.6 % faster than the initial foil. This results in a race-dominating time saving of 84 s over a typical race. This highly detailed design and the resulting performance gains have only been made possible through the use of the adjoint method. The framework is extended to account for additional courses and wind speeds in a multi-condition optimisation. The optimal multi-condition foil is similar to the single-condition foil but with a planform that shifts the center of lift further aft. This is favorable for the included high-speed cases as it produces more righting moment. This finding has only been enabled through the use of the full physics model. The final multi-condition foil is 7.2 % faster than the initial foil and saves 82 s over a race. The multi-condition foil is compared to foils optimised for the individual conditions. The individual foils perform best in their respective niches, whereas the multi-condition foil performs best across the range of conditions.

This study has undertaken the first adjoint, system-based optimisation of a hydrofoil. The use of the entire physics model eradicates inaccuracies and the adjoint method allows highly detailed design. The method solves in 1.5 h on a standard desktop PC, demonstrating its significant efficiency. Due to this efficiency, more physics and any number of design variables can be included. It can be applied to any foiling craft and is also suitable for similar applications in maritime and aerospace technology.

*“You are probably frightened at times, scared, worried, you hate it, you absolutely despise the fact that you are involved, but when you get to the finish you know why, because there is nothing like it, it gets in your blood, and you can’t get rid of it”*

***Sir Peter Blake on the Whitbread Round the World Yacht Race***

*Dedicated to my parents,*

*Johanna and Dr. Hubert Tannenberg.*

*Thank you for everything! I love you!*

# Table of Contents

# Table of Tables

# Table of Figures

Table of Figures

# Research Thesis: Declaration of Authorship

Print name: Rafael Andreas Maximilian Tannenberg

Title of thesis: Adjoint System-Based Optimisation of Hydrofoils Using Algorithmic Differentiation

I declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

Main publications:

- *Tannenberg, R., Turnock, S. R., Hochkirch, K. and Boyd, S. W. (2023) 'VPP-Driven Parametric Design of AC75 Hydrofoils', Journal of Sailing Technology 2023, vol. 8, no. 1, pp. 161 – 181.* https://doi.org/10.5957/jst/2023.8.9.161

- *Tannenberg, R., Hochkirch, K., Walther, A., Turnock, S. R. and Boyd, S. W. (2025) 'Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation', under review at Optimisation and Engineering.*

- *Tannenberg, R., Hochkirch, K., Turnock, S. R. and Boyd, S. W. (2025) 'Multi-Condition Hydrofoil Optimisation Using an Adjoint Velocity Prediction Program', under review at Ocean Engineering.*

Other publications:

- *Tannenberg, R., Turnock, S. R., Hochkirch, K. and Boyd, S. W. (2023) 'VPP-Driven Parametric Study on AC75 Hydrofoil Design',6th InnovSail Conference. Lorient, France.*

- *Tannenberg, R., Hochkirch, K., Walther, A., Turnock, S. R. and Boyd, S. W. (2024) 'Development of an Adjoint VPP-Driven Hydrofoil Optimisation Framework', 8th High Performance Yacht Design Conference. Auckland, New Zealand.*

- *Tannenberg, R., Hochkirch, K., Turnock, S. R. and Boyd, S. W. (2025) 'Adjoint Optimisation of Hydrofoils for Multiple Conditions', 25th Chesapeake Sailing Yacht Symposium, Annapolis, Maryland, USA.*

- *Melis, F. M., Tannenberg, R., Boyd, S. W., Scharf, M. and Abdel-Maksoud, M. (2024) 'AC75 Aerodynamic Force Prediction Using a 3D Panel Code', 8th High Performance Yacht Design Conference. Auckland, New Zealand.*

- *Melis, F. M., Tannenberg, R., Boyd, S. W. and Abdel-Maksoud, M. (2024) 'AC75 Aerodynamic Performance Prediction via BEM, Journal of Sailing Technology 2024, vol. 9, no. 1, pp. 143 - 174.* https://doi.org/10.5957/jst/2024.9.1.143

Signature: ..... Date: 6th of June 2025

# Acknowledgements

I am incredibly thankful to the University of Southampton, the School of Engineering and to all the people that have worked with me during my PhD. This whole project was an amazing challenge and I'm very grateful I have been given the opportunity. Tackling this challenge would have not been possible without you all. The biggest thank you

To Prof. Stephen Boyd for just everything. From finding the funding for this self-proposed project, over academic guidance, to motivating me in hard times. Above all for being a true friend. I'm very grateful that you picked up my idea and made all this possible. This has been a big dream of mine and thanks to you I got the chance to pursue it. I will never forget this.

To Prof. Stephen Turnock for all the good advice, for reminding me to think outside of my project and for organising the funding. Thank you for giving me this opportunity.

To Dr. Karsten Hochkirch for allowing me to use his software FS-Equilibrium and giving me access to its source code. For all the help finding my way in the code and the endless meetings working in the code trying to make it produce stable results in the optimisation. Thank you for being a mentor.

To Prof. Andrea Walther for letting me use her algorithmic differentiation tool ADOL-C and the very valuable advice on how to use it to differentiate our code. Thank you very much for welcoming me in your group in Berlin.

To Dr. Heikki Hansen for picking up my idea from the DNV side. Thank you for answering Michele's and my many questions on FS-Equilibrium.

To Dr. Martin Fischer for helping me develop the idea for this project, for all the insights into the America's Cup world that are otherwise hidden and the long discussions on my results. Thank you for being a mentor.

To my good friends and colleagues Michele, Chris and Salvo for working with me on this project for countless days, nights and weekends.

To my good friend Andi for rendering the AC75 and for designing all the other yachts with me over the last decade almost.

To my good friends Konrad, Aaron, Felix, Milia, Umberto and Tom for the good time we had in Southampton.

To Becky Walford, the King's Harbour Master of Portsmouth, for taking me out to all the regattas on Brenda's J.

Above all to my family. Leo, Johanna and Dr. Hubert Tannenberg. Thank you for everything!

I hope I can give it all back to you one day!

# Definitions and Abbreviations

| | |
|---|---|
| $A_i$ | Area of segment $i$ [m$^2$] |
| $A_F$ | Foretriangle area [m$^2$] |
| $A_J$ | Jib area [m$^2$] |
| $A_M$ | Mainsail area [m$^2$] |
| $A_N$ | Nominal area [m$^2$] |
| $A_R$ | Aspect ratio |
| $C_D$ | Drag coefficient |
| $C_{\mathrm{Di}}$ | Induced drag coefficient |
| $C_{\mathrm{Dpj}}$ | Jib profile drag coefficient |
| $C_{\mathrm{Dpm}}$ | Mainsail profile drag coefficient |
| $C_H$ | Heeling force coefficient |
| $C_L$ | Lift coefficient |
| $C_{\mathrm{Lj}}$ | Jib lift coefficient |
| $C_{\mathrm{Lm}}$ | Mainsail lift coefficient |
| $C_R$ | Driving force coefficient |
| $c$ | Chord length [m] |
| $c_{ro}$ | Root chord [m] |
| $D$ | Drag force [N] |
| $D$ | Induced drag [N] |
| $\boldsymbol{dv}$ | Hydrofoil design variables [m, °] |
| $d_z$ | Sink (ride height) [m] |
| $E_{\mathrm{HM}}$ | Mast height above shear [m] |
| $F$ | VPP Function [kts] |
| $F_A$ | Average shear height [m] |
| $\mathrm{FAE}_{x1}$ | Foil arm extension station 1 x-position [m] |
| $\mathrm{FAE}_{c1}$ | Foil arm extension station 1 chord [m] |
| $\mathrm{FAE}_{x2}$ | Foil arm extension station 2 x-position [m] |
| $\boldsymbol{F_i}$ | Force at control point $i$ [N] |
| $\mathrm{FWP}_{t15}$ | Foil wing portside station 15 twist [°] |
| $f$ | Function |
| $\boldsymbol{fm}$ | force- and moment vector [N, Nm] |
| $f_i$ | Function $i$ |
| $f_x$ | Force in $x$-direction [N] |
| $f_y$ | Force in $y$-direction [N] |
| $f_z$ | Force in $z$-direction [N] |
| $h$ | Submersion [m] |
| $L$ | Lift force [N] |
| $l_i$ | Lagrangian multiplier $i$ |
| $\boldsymbol{l_i}$ | Length of segment $i$ [m] |
| $l_{ext}$ | Extension lengths [m] |
| $l_{spa}$ | Semi span [m] |
| $N_{\mathrm{i,p}}(u)$ | Nonrational B-spline basis function in *u*-direction |
| $N_{\mathrm{j,q}}(v)$ | Nonrational B-spline basis function *v*-direction |
| $m_x$ | Moment around $x$-axis [Nm] |

| | |
|---|---|
| $m_y$ | Moment around $y$-axis [Nm] |
| $m_z$ | Moment around $z$-axis [Nm] |
| $\boldsymbol{P}_{\mathrm{i,j}}$ | Control points of NURBS-Surfaces |
| $p, q$ | Degrees of NURBS surfaces in *u*- and *v*-direction |
| $\boldsymbol{p}$ | Point a function is evaluated for |
| $R_F$ | Frictional resistance [N] |
| $R_T$ | Total resistance [N] |
| $r_{tap}$ | Taper ratio |
| $\boldsymbol{r_{i_0 j}}$ | Vector from node $i_0$ to control $j$ |
| $\boldsymbol{r_{i_1 j}}$ | Vector from node $i_1$ to control $j$ |
| $r_{i_0 j}, r_{i_1 j}$ | Magnitudes of $\boldsymbol{r_{i_0 j}}$ and $\boldsymbol{r_{i_1 j}}$ |
| $\boldsymbol{S}$ | NURBS surface |
| $s$ | Span [m] |
| $t$ | Thickness [m] |
| $u, v$ | Directions of NURBS surfaces |
| $\boldsymbol{u}_\infty$ | Unit vector in the direction of the freestream |
| $\boldsymbol{V_i}$ | Local velocity at control point $i$ [m s$^{-1}$] |
| $V_{\mathrm{MG}}$ | Velocity made good [kts] |
| $V_S$ | Boat speed [kts] |
| $V_{\mathrm{TW}}$ | True wind speed [kts] |
| $\boldsymbol{V}_\infty$ | Free-stream velocity [m s$^{-1}$] |
| $\boldsymbol{v_{ji}}$ | Influence of horseshoe vortex $j$ on control point $i$ [m$^{-1}$] |
| $w$ | Downwash |
| $w_{\mathrm{i,j}}$ | Weights of control points of NURBS surfaces |
| $x_i$ | Input variable $i$ |
| $(1+k)$ | Form factor |

| | |
|---|---|
| α | Angle of attack/incidence [°] |
| $\beta_a$ | Apparent wind angle [°] |
| $\beta_t$ | True wind angle [°] |
| $\Gamma_j$ | Strength of horseshoe vortex $j$ [m$^2$ s$^{-1}$] |
| $\delta_e$ | Elevator rake [°] |
| $\delta_f$ | Flap angle [°] |
| $\delta_r$ | Rudder angle [°] |
| $\delta_{\mathrm{ra}}$ | Rake angle [°] |
| $\delta_{win}$ | Wing rake angle [°] |
| ϵ | Downwash angle [°] |
| $\theta$ | Pitch angle [°] |
| λ | Leeway angle [°] |
| $\boldsymbol{\xi}$ | Free variable vector |
| $\rho$ | Density [kg m$^{-3}$] |
| τ | flat parameter |
| $\varphi$ | Heel angle [°] |

| | |
|---|---|
| as | Automatically spaced |
| AVL | Athena Vortex Lattice |
| CAD | Computer-aided design |

| | |
|---|---|
| CFD | Computational fluid dynamics |
| CVS | Circolo della Vela Sicilia |
| CoG | Centre of gravity |
| ETNZ | Emirates Team New Zealand |
| FEA | Finite element analysis |
| FSI | Fluid-structure interaction |
| GUI | Graphical user interface |
| IMS | International measurement system |
| IPOPT | Interior Point Optimizer |
| LLT | Lifting line theory/method |
| LRPP | Luna Rossa Prada Pirelli |
| NURBS | Non-uniform rational B-Splines |
| NSGA2 | Non-Dominated Sorting Genetic Algorithm |
| PSQP | Preconditioned Sequential Quadratic Programming |
| RANS | Reynolds-averaged Navier-Stokes |
| RANSE | Reynolds-averaged Navier-Stokes equations |
| RNZYS | Royal New Zealand Yacht Squadron |
| RYS | Royal Yacht Squadron |
| SLSQP | Sequential Least Squares Programming |
| SNOPT | Sparse non-linear optimiser |
| TWS | True wind speed |
| VPP | Velocity prediction program |

# Chapter 1 Introduction

## 1.1 Motivation

The numerical optimisation of an engineering component requires three things: a parametric model that describes the component as a function of design variables, a physics model that computes the performance of the component based on the current set of design variables and an optimisation algorithm that identifies the best performing set of design variables in an iterative process. This is true for many optimisation problems. In some cases however, it is not the component performance itself that is of interest, but rather the performance of a system of which the component is a part. One such example are hydrofoils (components) on racing yachts (system). Hydrofoils are lifting surfaces operated underwater to lift the hull of a yacht out of the water and are used for example in the America's Cup, the Formula 1 of sailing. The aim in the America's Cup is to have the best performing yacht on the racecourse, so the system performance is paramount. If in such cases an isolated component optimisation is performed, two problems occur. First, the optimal component performance might not directly translate to the optimal system performance, as entire aspects of how the component influences the system are not considered. Secondly, if the influence of a component on the state of the system is so large that it significantly affects the state of the system, the isolated optimisation does not account for the changes in system state during the optimisation. This introduces inaccuracies and falsifies the optimal solution.

In hydrofoil optimisation both effects are prominent. Only modelling the component performance, for example in terms of lift to drag ratio, neglects important aspects such as the total lift and the righting moment generated which significantly affect the yacht. Constraints can be used to ensure a foil delivers the required forces and moments, but these requirements are then fixed and were derived from some initial design. Therefore, the optimisation does not have the possibility to identify a foil design with, for example, a higher righting moment leading to higher performance. Furthermore, the design of the hydrofoils has a significant influence on the state and the trim (i.e. control) of the yacht. Optimising the component in isolation means that every foil is assessed for an initial state and trim which is not updated throughout the optimisation (e.g. leeway angle and rake angle). This ignores the changes in state and trim associated with changing foil designs and introduces inaccuracies. The significance of this was shown by Paulin et al. (2015) for an America's Cup test boat. Two promising hydrofoils were compared using a stationary physics model of the yacht. The model predicted differences in boat speed of more than 20%. Other state variables, such as the leeway angle, changed by a factor of 2.2 and the control-variable rake by a factor of 2. Nevertheless, the current practice is to optimise hydrofoils

in isolation of the yacht system neglecting these significant changes in yacht state and trim. The reason for this is the lower complexity in the development and validation of the physics model and the lower computational requirements. Modelling a whole system requires significantly more computational resources than modelling a single component.

Components of a system, which have a strong influence on the system, also promise large potential performance gains. To fully extract this potential, detailed optimisation with a high number of design variables is needed. A detailed optimisation of the spineform, the planform and the twist distribution of an America's Cup foil requires roughly 70 variables. This extends to hundreds of variables if section design is to be included. However, the computational time of most optimisation strategies scales poorly with the number of design variables and makes detailed design impossible. For example, a parametric study with 70 variables and five values tested each would require $5^{70}$ model evaluations, making it practically incomputable. This is the reason why published hydrofoil optimisations usually only employ between two to ten design variables. The only option for detailed optimisation is gradient-based optimisation which scales better with the number of design variables. However, also gradient-based methods can become prohibitively expensive if the gradient is approximated using the typical finite difference method. For the approximation of a single gradient with 70 design variables, 71 evaluations of the physics model are needed, and the gradient must be computed numerous times during the optimisation. Other differentiation techniques exist but are similarly as costly. The only exception is the adjoint method which is a reverse application of the chain rule to a function. By differentiating in reverse, the gradient can be computed at small computational cost independent of the number of input variables. This allows the optimisation of thousands of design variables and is the key to detailed design. However, applying the chain rule to a function in the form of computer code is complex and requires significant implementation efforts compared to the relatively simple finite difference method.

System performance is often also heavily influenced by external factors and the optimal component design might be sensitive to these factors. If the factors vary across the operation of the system, this must be accounted for in the optimisation. America's Cup yachts are raced on upwind- and downwind courses in different wind speeds. The different wind directions and speeds significantly affect the performance and the state of the system. The optimal design of the hydrofoil is thereby sensitive to these conditions. Therefore, optimising the hydrofoil for a single condition neglects this dependency and multi-condition optimisation should be conducted instead. This adds another layer of complexity. In published hydrofoil optimisations, this variability is mostly not considered.

In summary, this requires a sophisticated hydrofoil optimisation strategy with a model of the entire system to deliver the correct optimum solution and many variables to fully extract the immense potential. The only way a full physics model of the entire yacht system can be integrated into a detailed optimisation routine is a gradient-based strategy, where the adjoint method is used to compute the gradient. The method must also account for varying wind speeds and directions.

## 1.2 Aims and objectives

*The aim of this research is to integrate a full physics model of an entire yacht and a detailed parametric model into a computationally efficient hydrofoil optimisation framework to obtain accurate yacht performance predictions and the ability to fully exploit the design space, with the ultimate goal to gain a competitive advantage in sailing races like the America's Cup.*

To achieve this aim the following projects are defined with their respective objectives:

**I. Reviewing the current state of scientific knowledge in hydrofoil optimisation**

a. Reviewing the literature relevant to the topic of hydrofoil optimisation
b. Gaining insights into the hydrofoil optimisation methodologies of high-performance sailing teams

**II. Development and first applications of the AC75 physics model**

a. Development of a parametric model of the AC75 hydrofoil relating design variables to the hydrofoil geometry
b. Development of an advanced lifting line model of the AC75 hydrofoil that enables the computation of the forces and moments produced by a certain hydrofoil geometry
c. Development of the full physics model of the AC75 using the VPP FS-Equilibrium to enable accurate performance assessment of the AC75 for a given hydrofoil geometry
d. Validation of the VPP model with real life performance data recorded during the last America's Cup
e. VPP-driven parametric study on the five main hydrofoil design variables to gain first insights into their effect on boat performance and to have a benchmark for the first optimisation results

**III. Development of an adjoint system-based hydrofoil optimisation framework**

a. Preparation for and differentiation of the parametric- and the lifting line model using the algorithmic differentiation tool ADOL-C
b. Development of a finite difference-based routine to approximate the gradients of the physics model excluding the lifting line method
c. Combination of the adjoint gradients from the lifting line method and the parametric model with the finite difference-based gradients from the remainder of the physics model
d. Integration of the models and differentiation routines into a system-based single-objective optimisation
e. Validation of the approach using simple optimisation case studies
f. VPP-driven optimisation with the aim to maximise the velocity made good of the yacht for a single condition

**IV. Extension of the framework to multi-condition optimisation**

a. Development of a routine to compute the gradients of multiple conditions and averaging them to achieve a single objective
b. Integration of the gradient averaging method into the optimisation routine
c. Multi-condition optimisation with the aim to maximise the average velocity made across multiple conditions

## 1.3 Report structure and publications

In this section, the structure of the report is presented. Three journal papers form the main body of the work and are accompanied by this introduction, a chapter on the literature review, a conclusionary chapter and an appendix.

**Chapter 1 - Introduction** has introduced a class of optimisation problems where a component of a system, that has a strong influence on the system performance and state, is to be optimised. One such problem are hydrofoils. This requires the development of an optimisation algorithm capable of accurate system performance analysis and detailed optimisation which was set out as the aim of this research. A set of objectives has been defined to achieve this aim. The objectives have been grouped into projects that correspond to the following chapters.

**Chapter 2 - Literature review** corresponds to objectives I a. – b. It presents and discusses the literature relevant to achieving the aim of the thesis. This includes an introduction to the yacht system, to hydrofoils, to hydrofoil performance models and to optimisation strategies. It pays special emphasis on optimisation approaches for problems with a high number of design

parameters, i.e. gradient-based methods, and approaches for gradient computation. Furthermore, it presents published hydrofoil optimisation methods and the approach of one America's Cup team.

**Chapter 3 - VPP-Driven Parametric Design of AC75 Hydrofoils (Paper 1)**

> *Tannenberg, R., Turnock, S. R., Hochkirch, K. and Boyd, S. W. (2023) 'VPP-Driven Parametric Design of AC75 Hydrofoils', Journal of Sailing Technology 2023, vol. 8, no. 1, pp. 161 – 181.* https://doi.org/10.5957/jst/2023.8.9.161

This paper corresponds to objectives II a. – e. In the paper, the development of the full stationary physics model of the AC75 in the VPP FS-Equilibrium is presented. This includes the detailed parametric model of the foil geometry and the hydrodynamic model of the foil computing the hydrofoil forces and moments. The models accurately predict the performance of the yacht on the racecourse with a given foil. The hydrodynamic model and the entire physics model are validated. Using the developed physics model, a parametric study, systematically varying the main hydrofoil design variables, is conducted. The parameters include the anhedral angle, the span, the chord, the taper ratio and the rake of the foil. This allows first insights and forms a good benchmark for a first, simple test of the optimisation algorithm developed later. The parametric study predicts a difference of 13.3% $V_{\mathrm{MG}}$ from the slowest to the fastest foil highlighting the immense potential of hydrofoil optimisation. To fully extract this potential, detailed optimisation is required. The parametric study as a conventional optimisation tool is not feasible for detailed design. Therefore, a gradient-based optimisation approach, where the gradient is computed using the adjoint method is proposed. The paper won the Best Student Paper Award from the Journal of Sailing Technology in 2023 and the Mandles Prize for Hydrofoil Excellence in 2024.

**Chapter 4 - Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation (Paper 2)**

> *Tannenberg, R., Hochkirch, K., Walther, A., Turnock, S. R. and Boyd, S. W. (2025) 'Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation', under review at Optimisation and Engineering.*

This paper corresponds to objectives III a. – f. Chapter 2 has highlighted the need for accurate performance prediction capabilities which was achieved through the development of the AC75 physics model in Chapter 3. It has also shown that detailed design is required to fully extract the potential of foil design and that conventional methods cannot cope with the high numbers of design variables required for detailed optimisation. Hence, in this chapter an innovative optimisation approach is developed. It allows the use of the physics model to evaluate the performance of a candidate and the optimisation of potentially hundreds of design variables. This

is achieved through the differentiation of the physics model with a combination of finite differences and the adjoint method. The adjoint method is used for the bottleneck of the routine. Finite differences are used to differentiate the parts of the program that are not affected by the high number of input variables. The two gradient parts are combined and provided to the optimisation algorithm IPOPT. The resulting method is first benchmarked against the parametric study and finds the same optimal values for the five variables. Thereafter, the method is used to optimise the twist distribution of a simple foil, which successfully achieves the desired elliptical lift distribution. Finally, a full optimisation with 68 design variables is performed. It optimises the planform, the spineform and the twist distribution of the foil in detail. The optimisation is executed for a single upwind condition and delivers a feasible, smooth and high-performance foil with a trust-worthy convergence history.

**Chapter 5 - Multi-Condition Hydrofoil Optimisation Using an Adjoint VPP (Paper 3)**

> *Tannenberg, R., Hochkirch, K., Turnock, S. R. and Boyd, S. W. (2025) 'Multi-Condition Hydrofoil Optimisation Using an Adjoint Velocity Prediction Program', under reiview at Ocean Engineering.*

This paper corresponds to objectives IV a. – c. Hydrofoils are operated across a wide range of conditions. In yacht racing, this includes different wind speeds and different courses. The optimal design is thereby sensitive to these conditions. However, the previous optimisations have been performed for a single condition only. Therefore, the framework is extended to optimise the hydrofoil for multiple conditions. This is achieved by computing and then averaging the $V_{\mathrm{MG}}$ and gradients for every condition resulting in a multi-condition, but single-objective optimisation approach. The most-likely conditions for an America's Cup race are determined from historical weather data for a specific time and venue resulting in three wind speeds (8, 11 and 14 kts true wind speed $V_{\mathrm{TW}}$) and a respective up- and downwind course, so six conditions in total. The optimal multi-condition foil is compared to the individual optimal foils.

**Chapter 6 – Conclusions** summarises the key findings of this study and draws conclusions. It discusses the novelty and the limitations of the approach. A section on further work proposes areas of future work to increase the capabilities of the method. The general approach is not limited to hydrofoils and yacht racing but can be applied to any similar optimisation problem. Potential other applications are presented.

**Appendix A – Algorithmic differentiation of the lifting line method and the parametric model** presents the implementation work that was required to differentiate the bottleneck of the AC75 physics model with the algorithmic differentiation tool ADOL-C. It is presented for the reader who wishes to get an insight into the implementation process. The appendix introduces the original

implementations, presents the general steps required for differentiation with ADOL-C and their application to the parametric and the lifting line model.

## 1.4 Author contributions

The three journal papers have been co-authored by Prof. Stephen W. Boyd and Prof. Stephen R. Turnock from the University of Southampton, as well as Dr. Karsten Hochkirch from the DNV Ship Performance Center. In the following, they are referred to as S.W.B., S.R.T. and K.H, respectively. The second paper has also been co-authored by Prof. Andrea Walther from the Humboldt-Universität zu Berlin, she is referred to as A.W. Rafael Tannenberg is referred to as "the candidate".

**Paper 1:** In this paper, the candidate has developed and validated the VPP model of the AC75 yacht, as well as the parametric models of the hydrofoil and its integration into the VPP. He has written the script conducting the parametric study, has produced and analysed the results. He also wrote the manuscript. The VPP model is set-up in the software FS-Equilibrium, which was developed by K.H. General guidance and supervision was provided by S.W.B. and S.R.T. All authors reviewed the manuscript.

**Paper 2:** In this paper, the candidate has developed the adjoint system-based optimisation strategy and implemented it. He produced and analysed the results. He also wrote the manuscript. The system-model is based on the software FS-Equilibrium, which was developed by K.H. The adjoint version of the bottleneck of the system-model was obtained through ADOL-C, an algorithmic differentiation tool partly developed by A.W. Advice on how to use ADOL-C to differentiate FS-Equilibrium was provided by K.H. and A.W. General guidance and supervision was provided by S.W.B. and S.R.T. All authors reviewed the manuscript.

**Paper 3:** In this paper, the candidate developed and implemented the multi-condition approach. He produced and analysed the results. He also wrote the manuscript. The system-model is based on the software FS-Equilibrium, which was developed by K.H. General guidance and supervision was provided by S.W.B. and S.R.T. All authors reviewed the manuscript.

## 1.5 Further publications

Other publications have been published as part of this project but are not part of the thesis. This includes the three following conference papers:

1. *Tannenberg, R., Turnock, S. R., Hochkirch, K. and Boyd, S. W. (2023) 'VPP-Driven Parametric Study on AC75 Hydrofoil Design', 6th InnovSail Conference. Lorient, France.*

2. *Tannenberg, R., Hochkirch, K., Walther, A., Turnock, S. R. and Boyd, S. W. (2024) 'Development of an Adjoint VPP-Driven Hydrofoil Optimisation Framework', 8th High Performance Yacht Design Conference. Auckland, New Zealand.*

3. *Tannenberg, R., Hochkirch, K., Turnock, S. R. and Boyd, S. W. (2025) 'Adjoint Optimisation of Hydrofoils for Multiple Conditions', 25th Chesapeake Sailing Yacht Symposium, Annapolis, Maryland, USA.*

In addition, a student project has been initiated to develop an accurate force model of the AC75 sail plan based on a boundary element method. The project was undertaken by Michele Melis in the form of a Master thesis with the Technische Universität Hamburg-Harburg and the University of Southampton. The project has led to a conference paper co-authored by the candidate and a publication in the Journal of Sailing Technology. A second master project was started developing a structural model of the AC75 foil. This project was conducted by Christian Hülß. Both projects were initiated and co-supervised by the candidate. These are:

4. *Melis, F. M., Tannenberg, R., Boyd, S. W., Scharf, M. and Abdel-Maksoud, M. (2024) 'AC75 Aerodynamic Force Prediction Using a 3D Panel Code', 8th High Performance Yacht Design Conference. Auckland, New Zealand.*

5. *Melis, F. M., Tannenberg, R., Boyd, S. W. and Abdel-Maksoud, M. (2024) 'AC75 Aerodynamic Performance Prediction via BEM, Journal of Sailing Technology 2024, vol. 9, no. 1, pp. 143 - 174. https://doi.org/10.5957/jst/2024.9.1.143*

6. *Hülß, C.A. (2023). Deformation Model for AC75 foil optimisation. Master Thesis, University of Southampton.*

# Chapter 2 Literature review

## 2.1 The yacht system

Modern sailing racing yachts are incredibly complex systems. They feature sophisticated sail plans, hull geometries and hydrofoils and are made from advanced materials. Complex control systems and electronics are used to assist the sailors. The design of the yachts is a multi-disciplinary and highly technological endeavour. The most advanced yachts are sailed in the America's Cup, the Formula 1 of sailing. Similar to Formula 1, the teams can develop the boat within a certain rule. This makes the competition a design race. The yacht type currently sailed in the America's Cup is the AC75. The AC75 rule was written for the 36th America's Cup (Royal New Zealand Yacht Squadron and Circolo Della Vela Sicilia, 2020) and updated for the 37th (Royal New Zealand Yacht Squadron and Royal Yacht Squadron, 2023). The AC75 is shown in Figure 2.1.

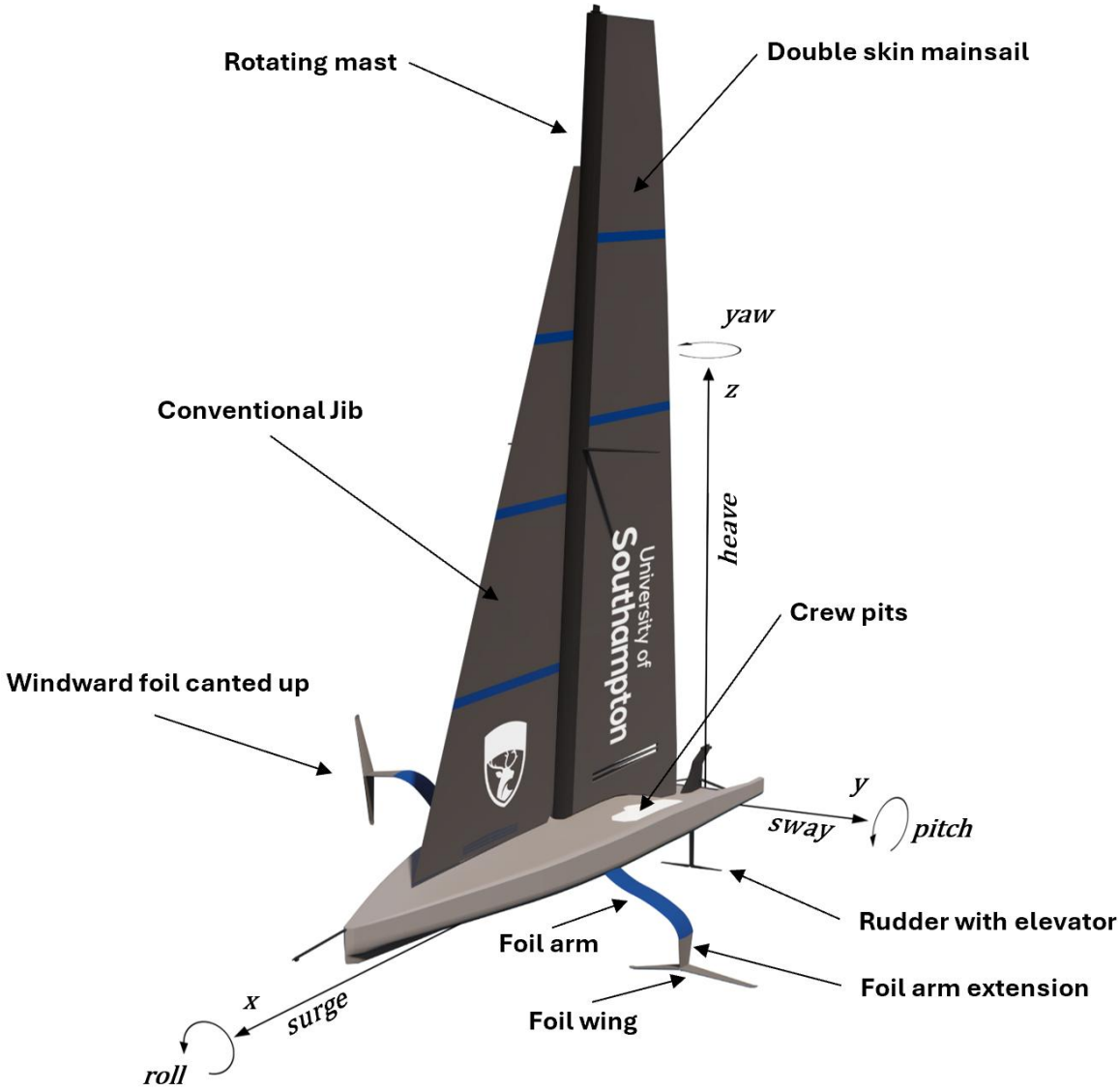



Figure 2.1: AC75 type yacht with its six degrees of freedom

The yacht has six degrees of freedom. These are the three translational degrees surge, sway and heave, and the three rotational degrees roll/heel, pitch and yaw. The forces and moments acting on the yacht system are usually resolved into these six degrees of freedom. They correspond to $f_x, f_y, f_z, m_x, m_y$ and $m_z$. If all forces and moments are in equilibrium in all six degrees of freedom, the boat is sailing in a steady state. If they are not in equilibrium, the yacht behaves dynamically. This is for example the case in manoeuvres. The forces acting on the AC75 are generated by its components. The five main components of the AC75 are:

**Sail plan –** The AC75 features a conventional jib paired with a double skin mainsail on a rotating mast. The double skin mainsail has higher aerodynamic efficiency than conventional sails and includes complex control systems. These allow a high degree of shape control and enable, for example, deep camber for high lift coefficients. The rotating mast permits alignment of the leading edge with the flow. This increases efficiency further. The sails provide the drive force ($f_x$) that propels the yacht. They also produce less desirable side force ($f_y$) and the resulting heeling- ($-m_x$), pitching- ($m_y$) and yawing moments ($m_z$). The sail plan is controlled by the sail trimmers using hydraulic control systems.

**Hull –** The hull of the AC75 is mainly an aerodynamic component since the yachts usually have a flight time of 100% during a race. Nevertheless, their take-off behaviour is very important, especially in light winds. The hull accommodates most of the control systems, the crew and batteries that power the hydrofoil control. The hull mainly produces aerodynamic drag (-$f_x$), side force ($f_y$) and gravitational force (-$f_z$) plus the resulting moments.

**Crew –** The AC75 is sailed by a crew of 11 (or 8 according to the new rule) that are responsible for steering, trimming and “flying” the yacht. Parts of the crew are so called “grinders” or “cyclers” generating hydraulic pressure. This hydraulic pressure is used to control the sails. The weight of the crew produces down force (-$f_z$), righting- ($m_x$) and pitching moment ($m_y$).

**Rudder –** The rudder is used to steer the yacht but also provides pitch stability with the horizontal elevator attached to its end. The rudder can be raked forward or aft to change the amount of pitching moment the elevator produces. The rudder and elevator produce drag (-$f_x$), side force ($\pm f_y$) and vertical lift or downforce ($\pm f_z$) plus the accompanying moments ($\pm m_x$, $\pm m_z$, $\pm m_z$). The rudder and elevator are adjusted by the crew.

**Hydrofoils –** The AC75 is equipped with two main hydrofoils, one on each side of the yacht. The leeward foil is canted into the water and provides vertical lift ($f_z$) to support the weight of the yacht. Additionally, it generates the side force (-$f_y$) to counter the side force from the sails and the hull. It also generates righting (+$m_x$), pitching (-$m_y$) and yawing moments (-$m_z$). The windward foil is canted out of the water to reduce hydrodynamic drag (-$f_x$) and generate righting moment

($+m_x$) based on the foil's weight. Each foil consists of the foil arm, the extension of the arm and the foil wings. The foil wings and the extension are open to development within certain rules. This is discussed in detail in Paper 1. The foil wings are equipped with one or two flaps allowing the flight controller to regulate the forces produced.

Hydrofoils, and especially T-foils on canting arms, are a relatively new technology in the America´s Cup. The results are a steep learning curve and a large potential for performance gains. The AC75 rule was written to specifically promote hydrofoil development and therefore, the hydrofoils have been assigned a large design space compared to other components. This increases the potential for performance gains and makes the hydrofoils the biggest performance differentiators on the AC75. According to Dan Bernasconi, the Chief-Designer of Team New Zealand, the foils "…may well be the area which decides the next America's Cup" (America's Cup, 2020a). This requires sophisticated hydrofoil design and optimisation techniques.

## 2.2 Hydrofoils

Hydrofoils are lifting surfaces comparable to the wing of an airplane but operated underwater to lift a boat partially or fully out of the water. They can also be used to produce side force and increase righting moment. Hydrofoils have a very effective lift to drag ratio at semi-displacement and planning speeds and can therefore significantly reduce the overall drag of a watercraft. This is shown in Figure 2.2. Hydrofoils have been applied to sailing yachts, fast ferries and military vessels in the past, see for example Gilruth (1951), Buermann and Hoerner (1964) and Baron Von Schertel (1973), but have remained a niche technology due to their complexity. This changed with the introduction of hydrofoils to the 34$^{th}$ America's Cup in 2013 which has led to a renewed and widespread interest. They have been applied to many sailing yachts, surf boards and power boats since and are an area of active research. They are seen as a key technology in the decarbonisation of small and fast watercraft (e.g. Tyde, 2024).

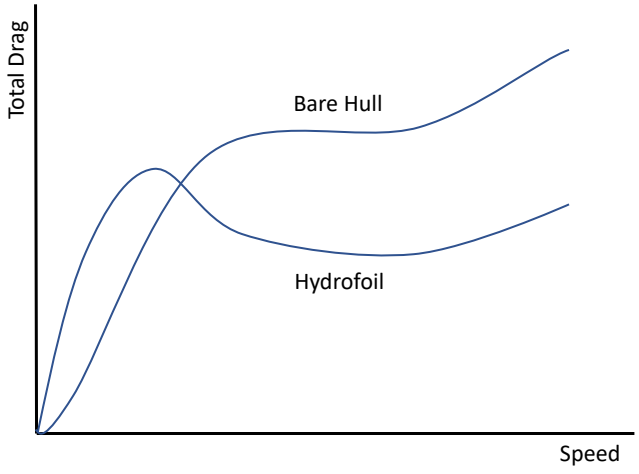



Figure 2.2: Differences in resistance of a boat with and without hydrofoils

Hydrofoils can have a variety of shapes and can be mounted to a vessel in different configurations. The types used in high performance sailing are mainly L-foils or T-foils. L-foils have the strut of the foil attached to one end of the foil, whereas T-Foils have it attached to the middle of the foil. Y-Foils are a variation of T-Foils and have their wing arranged at an anhedral angle. This is shown in Figure 2.3. In addition, hydrofoils can be tapered, swept, raked and twisted or exhibit more complex shapes. The "front-view" is called the spineform, the "top-view" the planform.

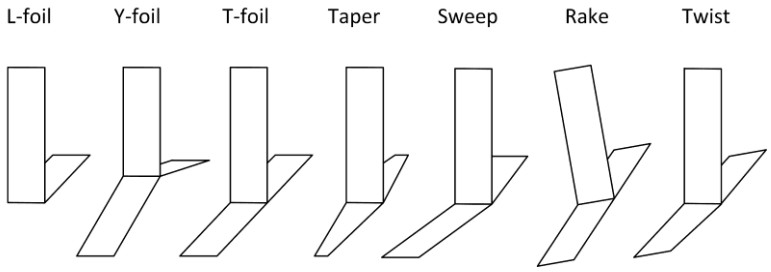



Figure 2.3: Hydrofoil design aspects

The shape of the hydrofoils significantly affects the vessel they are applied to. It influences the performance but also the whole state and trim of the boat. This has been demonstrated by Paulin et al. (2015) at the example of a C-Class catamaran. The study compared the two L-foils shown in Figure 2.4a for their performance in multiple conditions. The performance was assessed with a full stationary physics model of the catamaran developed in the VPP FS-Equilibrium. Both foils were considered promising but were found to exhibit very different performances. This is shown in Figure 2.4b. Foil "g2b" for example shows a 16% higher velocity made good for a true wind angle $\beta_t$ = 49° and a true wind speed $V_{\mathrm{TW}}$ = 9 m/s. Foil "sls03" on the other hand leads to a 20% higher boat speed ($V_S$) sailing downwind ($\beta_t$= 123° and $V_{\mathrm{TW}}$ = 6 m/s). This shows the immense importance of hydrofoil design and optimisation in a game where fractions of a second can decide a race. The influence of the foil design is so big that not only the boat speed changes but also the whole state and the trim of the boat. For example, the state variable "leeway" changed up to 3°, the rake of the foil changed 2° (Figure 2.5a and b). This significant influence on the state and trim of the yacht must be considered in hydrofoil design and optimisation.

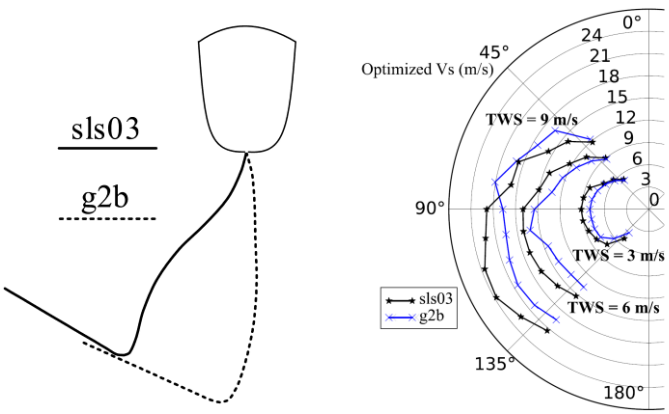



Figure 2.4: Spineform (a) and boat speed (b) of two foils for a C-Class yacht (Paulin et al., 2015)

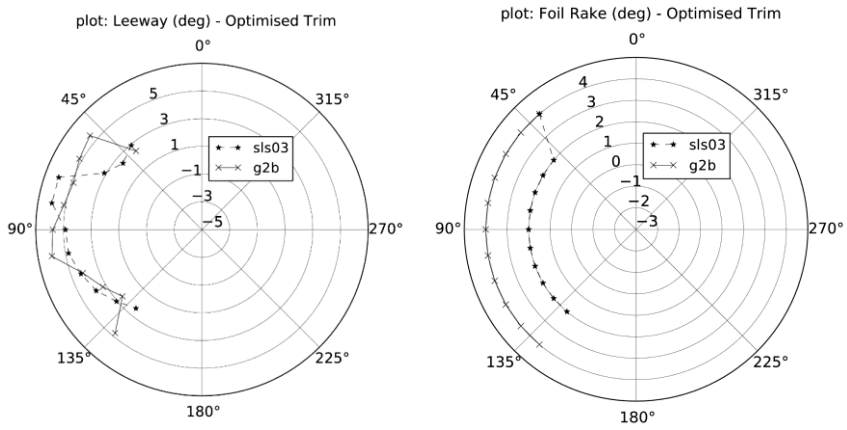



Figure 2.5: Leeway angle (a) and rake angle (b) of two foils for a C-Class yacht (Paulin, 2013)

Hydrofoils exhibit complex physics. These involve two-dimensional and three-dimensional effects, as well as cavitation and free-surface effects. A short introduction to these physics is given in the following, for a detailed description see Molland and Turnock (2022) and Faltinsen (2005). Hydrofoil sections are used to bend the streamlines of a flow. This is either achieved by orienting them at some angle of incidence α with respect to the free-stream direction or by using a cambered section shape. The streamline curvature causes a pressure drop on one side of the foil and a pressure increase on the other side. The consequence is the force shown in Figure 2.6a. This force is commonly decomposed in the lift force, that acts normal to the free-stream direction, and the drag force, that acts in the direction of the free-stream. The higher the angle of attack (or the camber), the more lift is produced. Initially, the amount of lift varies roughly linearly with the angle of attack up to a point called stall. Here, the flow starts to separate from the upper surface of the foil causing a reduction in lift. This is shown on Figure 2.6b where the lift is presented in terms of the non-dimensional lift coefficient $C_L$. The profile drag of the foil section (or the sectional profile drag coefficient $C_D$) varies parabolically with the angle of incidence and is a sum of the skin friction drag and the viscous pressure drag of the foil section.

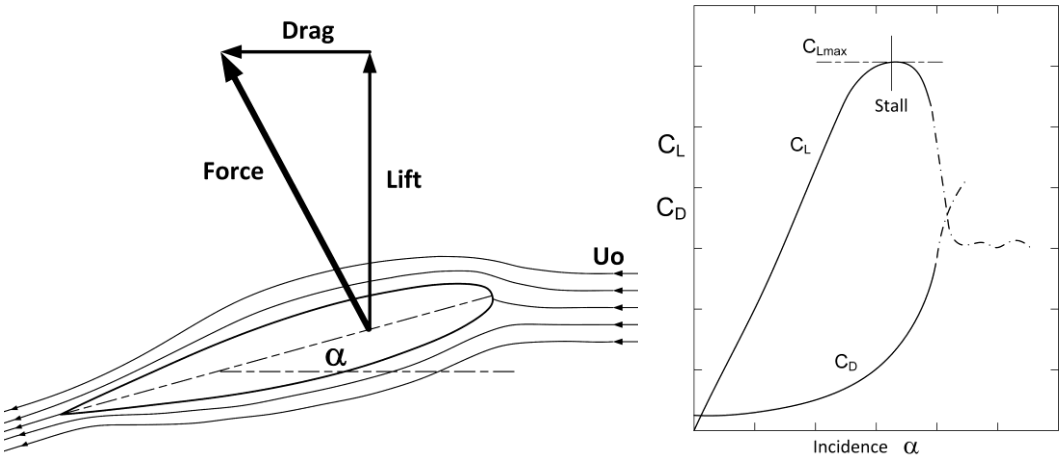



Figure 2.6: (a) Hydrofoil section at an angle of incidence, (b) lift- and drag characteristics of the section with respect to the angle of incidence; (Molland and Turnock, 2022)

Extending the foiling problem to three dimensions introduces additional effects. While 2D foils only experience flow in the chordwise direction, 3D-foils are also subject to spanwise flow. The pressure differential between the upper- and the lower face of the foil leads to a flow across the tip trying to equalise this differential. This results in the formation of tip vortices as shown in Figure 2.7a. The tip vortices cause a downward flow component commonly referred to as “downwash”. The downwash $w$ rotates the local velocity downwards by an angle $\epsilon$, reducing the effective angle of attack of the foil and rotating the lift force backwards ($L_0$). The part of the lift now pointing in the original free-stream direction is additional drag and called lift induced drag or short induced drag ($D_i$). This phenomenon is shown in Figure 2.7b.

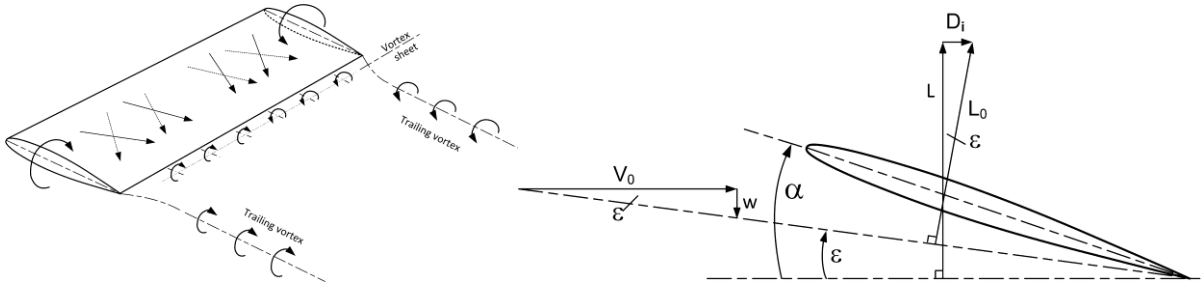



Figure 2.7: (a) tip vortices , (b) change in effective angle of attack due to downwash; (Molland and Turnock, 2022)

Prandtl (1921) has shown that for a constant span wing the induced drag is minimal when the downwash is constant across the span of the foil. This is achieved if the lift distribution along the span is elliptical as shown in Figure 2.8. The lift distribution is elliptical for a straight wing of elliptical planform. A rectangular planform causes a more rectangular lift distribution and a triangular planform a more triangular distribution. Additionally, the twist distribution and the sweep of the foil can be employed to modify the lift distribution. For complex foil geometries the downwash of a foil can also become negative in some parts of the wing. This is called “upwash”. Apart from upwash and downwash, complex 3D-foils can also experience interaction effects between different parts of the foil. This is the case for foils with anhedral angle and sweep for example.

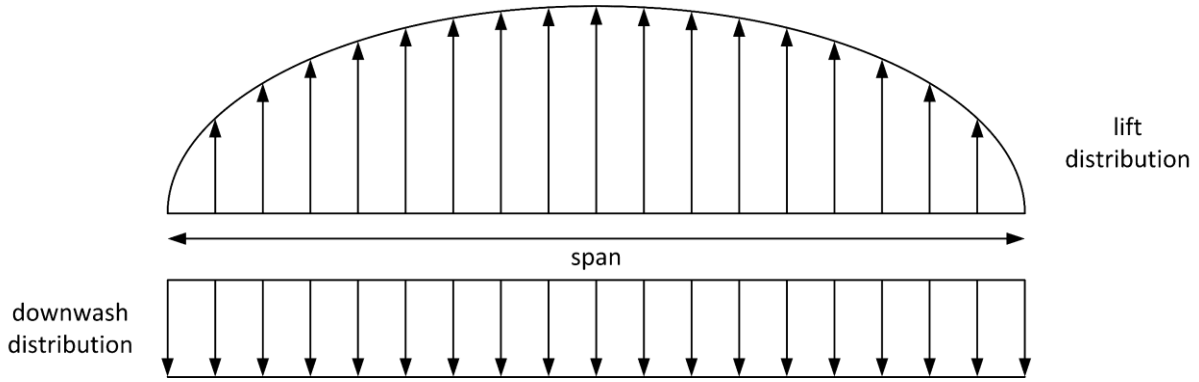



Figure 2.8: Lift and downwash distribution across the span of a wing

A further effect, not limited to 3D-foils, is cavitation. Cavitation is a formation of voids in the liquid occurring when the local pressure in the fluid drops below the vapour pressure. This can for example be caused by a hydrofoil producing very low pressure on the suction side. The result is a sudden loss of lift and an increase in drag. Furthermore, when the cavities reach a region with higher pressure again, they collapse and create shockwaves that can damage the foil. Cavitation is a concern for hydrofoils at the higher end of the speed range and in manoeuvres.

Free-surface effects are caused by the presence of the interface of the water and the atmosphere. Operating a hydrofoil close to this interface has several consequences. If the foil is operated at very low speeds, the free-surface acts like a wall and increases the performance of the foil (wing-in-ground effect). However, if the foil is operated at higher speeds, the free-surface causes a reduction of lift and an increase in drag due to wave-making. This is explained in detail in Section 3 of Paper 1. Additional drag can arise from water particles being projected in the air (spray). A further concern is ventilation. Very similar to cavitation it arises when the local pressure becomes very low. The proximity to the free surface allows air to be sucked down into the low-pressure region. This aeration also causes a sudden loss of lift. For further information see Faltinsen (2005).

## 2.3 Performance modelling

To understand and efficiently optimise the performance of hydrofoils, their physics must be modelled mathematically to some extent. Different models for hydrofoil force prediction are available. These can be used for isolated analysis or as part of VPPs or simulators. They vary in accuracy and complexity and require the correct trade-off to be made. Semi-empirical models exist for simple foil geometries, for example Whicker and Fehlner (1958), but complex shapes require numerical models. These include the simpler and less computationally expensive potential flow methods and the more expensive and accurate viscous methods. Available potential flow methods are for example the lifting line method (LLT), the vortex lattice method and the panel method. For a detailed introduction to those see Katz and Plotkin (1991). Viscous methods are commonly referred to as computational fluid dynamics (CFD) and include Reynolds-averaged Navier-Stokes- and higher-level methods. The different methods are presented in Table 2.1. A “+” indicates direct modelling capability, a “+-“ indirect or partial modelling capability and a “–“ no modelling capability. The order of computational time required for a single solve of a method on a normal PC is given in the last column.

Table 2.1: 3D fluid dynamic models for wings

| Method | Complex foils | Flow separation & stall | Bulbs | Free-Surface effects | Cavitation | Ventilation | Computational time |
|---|---|---|---|---|---|---|---|
| **Prandtl´s LLT** | - | - | +- | - | - | - | milliseconds |
| **Advanced LLT** | + | +- | +- | +- | - | - | milliseconds |
| **Vortex Lattice** | + | - | +- | +- | - | - | seconds |
| **Panel code** | + | - | + | +- | - | - | minutes |
| **RANS** | + | + | + | +- | +- | +- | hours |
| **Higher level CFD** | + | + | + | +- | +- | +- | Days/weeks |

Table 2.1 shows that viscous methods are the most capable approaches and can model complex foil geometries with bulbs even in fully stalled cases. With some addition they are also capable of modelling cavitation and free-surface effects including ventilation. However, a single solve of such a method requires potentially hours of computational effort. Hence, RANS methods provide a good option for detailed assessment of the flow surrounding a hydrofoil, but they are less suitable for use within VPPs or simulators. They are used for validation of lower-fidelity methods (e.g. Rousselon, 2020) and isolated hydrofoil optimisations as discussed in Section 2.5. Higher-level CFD methods are not common in hydrofoil design and optimisation.

In contrast, potential flow methods are much cheaper to solve computationally and are frequently used in VPPs and simulators. Except for Prandtl's original lifting line method, they are all capable of modelling complex hydrofoils. Prandtl's method cannot model sweep, anhedral or the operation of foils at a leeway angle due to the use of 2D-vortices to model the lift of the sections of the foil. Advanced lifting line methods exist that make use of 3D-vortices and can model such features. Non-linear lift-curve slopes and iterative solvers can be used to model the onset of stall. Viscous lift- and drag coefficients can be included to account for the sectional shape of the foil. One example is the method of Phillips and Snyder (2000). A transformation of a second foil across the free-surface plane can be used to model the loss of lift caused by the presence of the free-surface. This is called the "biplane-analogy". For more information see Section 3 of Paper 1. Similarly capable are vortex lattice methods. These can additionally model the camber of a foil directly but cannot account for stall. Both methods however assume thin, foil-like geometries and cannot directly model features such as bulbs. Options to include bulbs are for example empirical models or slender body theory. Panel codes can directly model "thick" geometries but cannot model stall and require significantly more computational time to solve

when compared to lifting line methods. Potential flow methods have frequently been used for hydrofoil force modelling in the America's Cup (Wilkins; Rousselon, 2020; Fischer, 2021).

Hydrodynamic models of hydrofoils can compute the forces and moments the foils produce and measures of efficiency such as lift to drag ratio. However, they do not model how a hydrofoil affects the yacht system. This requires a physics model of the entire yacht. These models are usually developed in velocity prediction programs. They require models of all aero- and hydrodynamic components of the yacht, as well as gravity models. The first VPP was developed at Massachusetts Institute of Technology by Kerwin (1978). It relied on data from wind tunnel- and towing tank investigations for each individual yacht. Later, semi-empirical models were used based for example on the ORC sail coefficients (Offshore Racing Congress, 2016) and the Delft Systematic Yacht Hull Series (Keuning and Sonnenberg, 1978). Additionally, numerical models have been introduced such as potential flow codes (Caponnetto et al., 1999) and fully integrated RANS solvers for both the aero and hydro modules (Böhm, 2014; Artemis Technologies, 2021; Robin et al., 2023). Furthermore, simplified models for the fluid-structure interaction between the air flow and the sails (Roux et al., 2008) and the water flow and hydrofoils have been presented (Horel and Durand, 2019).

To solve for the velocity on a given course at a given true wind speed, the VPPs search for a state where all forces provided from the force modules equate to zero. The forces can be assessed for a minimum of two degrees of freedom (surge and roll) up to all six. To find the equilibrium, an iterative approach or an optimisation routine can be used. Usually, the state variables are defined to balance the forces (for example boat speed for $f_x$ and leeway for $f_y$) and trim variables are used to optimise the performance (for example crew position or the flattening of the sails). These VPPs are called stationary. A typical solver set-up for a conventional yacht is presented in paper 1. This traditional solving strategy is valid for conventional and stable vessels. Hydrofoiling boats like the AC75 are inherently unstable and require a different set-up. This is also discussed in paper 1. Another approach is to assess the dynamic behaviour of a yacht by computing the forces and their effects on the boat in a time-series. These programs are referred to as dynamic velocity prediction programs (DVPPs) and can be controlled by scripted autopilots or by human interaction (sailing simulators).

Several commercial VPPs exist, like for example WinDesign (Wolfson Unit, 2024), the North VPP (North Sails, 2024) and the ORC VPP (Offshore Racing Congress, 2024). The most sophisticated velocity prediction algorithms are used by the America's Cup teams and in other high-performance yacht design applications. The programs usually have both a stationary and a dynamic mode. Examples are FS-Equilibrium used in the Oracle Team USA campaigns (Hochkirch, 2018) and Gomboc from Emirates Team New Zealand (SumToZero, 2021). Both FS-

Equilibrium and Gomboc incorporate a lifting-line approach for hydrofoil modelling. In addition, FS-Equilibrium incorporates a vortex lattice method and Gomboc a panel code. Both also employ a simplified fluid-structure interaction method for the hydrofoils based on an Euler beam-bending/twist model coupled to the potential flow models. Apart from the commercial VPPs, proprietary codes have also been used to model the performance of foiling boats, for example for the AC72 (Hagemeister and Flay, 2019), the Nacra 17 (Graf et al., 2020; Knudsen et al., 2023) and the AC75 (Patterson and Binns, 2022; Rodriguez et al., 2022).

For the optimisation of hydrofoils, a suitable performance model must be chosen. Typically, this is either a potential flow model or a RANS method. These however only model the foil in isolation. Paulin et al. (2015) have shown the immense influence of hydrofoils on the yacht state and trim, so an isolated optimisation does not account for such changes. Therefore, a full, stationary physics model of the yacht is chosen for this study. The physics model is developed in the VPP FS-Equilibrium. FS-Equilibrium has been employed for various projects in the past. This includes the simulation of manoeuvres of an IMS yacht (Richardt et al., 2005), the training of starting manoeuvres for America's Cup helmsmen (Binns et al., 2008) and the performance evaluation of different wind-assisted propulsion technologies (Hollenbach et al., 2020; Hochkirch and Bertram, 2022). FS-Equilibrium has also been applied to foiling boats such as the C-Class Catamaran Groupama (Paulin et al., 2015), the International Moth (Boegle et al., 2012; Eggert et al., 2020), the AC50 (Hansen et al., 2019) and the QFX Lake Racer (Melis et al., 2022). For a detailed description of FS-Equilibrium see paper 1.

The physics model of the entire yacht must include a force model of the hydrofoil. Solving for the steady sailing state of the yacht requires numerous hydrofoil force predictions and an optimisation requires numerous sailing states to be solved. This means that the computational time required for a foil force prediction heavily affects the time required for the optimisation. During an America's Cup campaign many optimisations must be run to first validate the approach and then use it throughout the campaign. This includes for example optimisations for different starting points, different discretisations and updating the optimal foil design, if advancements in other areas requires this (e.g. new foil control systems). Therefore, the time for an optimisation cannot be more than a couple of days on a standard PC at maximum (Fischer, 2021). Viscous methods for foil force prediction in combination with the VPP would result in an optimisation time of years. This could be reduced with the use of computing clusters but would still require weeks and significant cost. Therefore, potential flow methods must be used. FS-Equilibrium features an advanced lifting line method and a vortex lattice method. For this project, the lifting line method is chosen. It is cheaper to evaluate, can directly employ viscous profile lift and drag coefficients and can be made non-linear to account for the onset of stall.

## 2.4 Optimisation

This section briefly introduces optimisation techniques and methods for gradient computation. For a detailed introduction to optimisation see Martins and Ning (2022) and Keane and Prasanth (2005). For a detailed introduction to gradient computation see Griewank and Walther (2008). Furthermore, examples of gradient-based optimisations, where the gradient is computed using the adjoint method, are presented for various fields. Approaches to hydrofoil optimisation are discussed in detail in Section 2.5.

### 2.4.1 Optimisation strategies and algorithms

Optimisation in the mathematical context is the process of finding the minimum or maximum of a function with respect to input variables. The function to optimise is called the objective function. It can for example represent a biological, financial or an engineering system/component. The input or design variables describe this system/component. They must be independent and can be bounded to be within a feasible range. Additional constraints can be introduced via constraint functions to ensure structural soundness of an engineering component for example.

Different approaches exist for identifying the set of input variables that results in the minimum or maximum of the function. Simple approaches sample the design space in a regular- (e.g. parametric studies) or a random manner (e.g. Latin Hypercubes) with the best sample chosen. Sampling is simple to set-up and allows the use of the objective function in a black box manner but requires high numbers of objective function evaluations. Therefore, the methods are good for identifying trends but are not suitable for detailed optimisation. As discussed in the Introduction, a parametric study with 70 design variables and five values tested per variable would require $5^{70}$ function evaluations. This is practically incomputable, even for cheap functions. More sophisticated optimisation strategies/algorithms use an iterative approach. The information of a current function evaluation is used to inform the point/s evaluated in the next iteration/s. Generally, there are two types of optimisation algorithms, gradient-free and gradient-based approaches. Gradient-free or zeroth-order algorithms use no other information than the objective values. They are easy to set-up and the objective function is again used as a black box. Examples are genetic- or particle swarm methods. Gradient-based algorithms additionally use gradient information (first-order information). The gradient of the function with respect to the input variables informs locally as to how to change the variables to get closer to the (local) optimum. This additional information helps the algorithms to converge more efficiently, especially for high numbers of design variables. Figure 2.9 shows how the number of function evaluations required scales with the number of input variables for gradient-free and gradient-based methods. For this

example, the evaluations required by the gradient-based method grew from 67 to 206, the number of function calls required by the gradient-free method grew from 103 to 32,000 (Martins and Ning, 2022). This makes gradient-free methods prohibitively expensive for detailed optimisation and means that gradient-based algorithms are the only option for such problems.

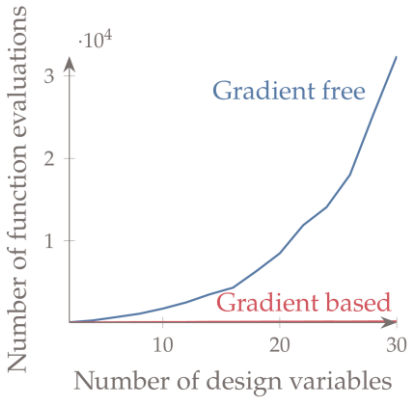



Figure 2.9: Number of function evaluations required for optimisation vs number of input variables from Martins and Ning (2022)

The downside of gradient-based methods is that they only search locally. This means that an identified minimum is only guaranteed to be the global minimum if the function is convex. If a function has multiple extrema and the optimiser is started in the vicinity of a local extremum, it will only find this local optimum as opposed to the global one. Starting the optimiser from multiple locations reduces this risk significantly (Multi-start). A preceding search with a global optimisation algorithm is another possibility but this is usually too computationally costly for problems with large numbers of input variables. In general, gradient-based strategies require the objective function and potential constraint functions to be differentiable.

A further technique is reducing the cost of the objective function by creating a meta- or surrogate model. This means several function evaluations are used to train a response surface representing the objective function. The optimisation is then performed on this much cheaper to evaluate response surface. Suitable meta models are polynomials, splines, kriging models or even artificial neural networks. For details see Forrester et al. (2008). However, meta-models are only of advantage if the number of function evaluations required for training them is lower than the number of function evaluations required in a normal optimisation. The function calls required to train the model thereby scale poorly with the number of design variables. This is described as the "curse of dimensionality" by Martins and Ning (2022) or as Keane and Prasanth (2005) put it: "as problem dimensions rise, the ability to construct any kind of accurate surrogate rapidly diminishes".

Optimisation cannot only be performed for multiple design variables but also for multiple objectives. For a sailing yacht for example, one might wish to optimise its performance in light- and strong winds. If it is unclear how to weigh these two objectives against each other, multiple optimisations with different weightings of the objectives can be performed. The different optimal designs form the "Pareto front" in a diagram that has the two objective values on the ordinates. The user can then pick the design that appears to be the best trade-off. This is known as the weighted sum approach. More efficient and capable ways of computing the Pareto front exist, but in any case, they require multiple optimisations to be run. If possible, the weighting should hence be determined beforehand (for example based on the likelihood of light and strong winds) to maintain a single objective which requires only one optimisation run.

This project aims to optimise high numbers of design variables to fully extract the immense potential of hydrofoils. Following the previous discussion, this is only possible with gradient-based methods. Several different gradient-based optimisation algorithms exist. Martins and Ning (2022) propose BFGS-methods (Broyden–Fletcher–Goldfarb–Shanno) for unconstrained problems and SQP- (Sequential quadratic programming) or IP-methods (Interior point) for constrained problems. While the optimisation problem in this project is unconstrained, future additions to the routine will likely require the introduction of constraint functions. Therefore, SQP- and IP-methods are considered. Lyu et al. (2014) have compared different optimisation algorithms for the optimisation of eight design variables defining the twist distribution of an aircraft wing. The results are shown in Figure 2.10. The gradient-free genetic algorithm NSGA2 required 12,757 iterations to identify the optimal twist distribution, the SQP-based optimisers SNOPT, PSQP and SLSQP required 27, 17 and 14, respectively. The IP-based optimiser IPOPT used 13. IPOPT has also been successfully used in conjunction with ADOL-C before (e.g. Gauger et al., 2012), so it is chosen as the optimiser for this project.

IPOPT is an interior point line search filter method for large-scale nonlinear optimisation problems developed by Wächter and Biegler (2006). The objective and potential constraint functions can be convex or non-convex but must be twice differentiable. Variables can be bounded with lower and upper bounds. IPOPT either uses supplied gradients or approximates the first order information with finite differences. For unconstrained problems, such as the problem in this thesis, it employs the BFGS-method. BFGS is a quasi-Newton method, which uses first and second order information of the function to find its minimum. In contrast to Newton-methods, the second order information is approximated with the first order information from the last two iterations. This means the second order information does not have to be computed, which can be computationally costly. Multiple starting points or a preceding global search should be considered to mitigate the risk of only identifying a local optimum. For a general introduction to interior point and BFGS methods see Martins and Ning (2022).

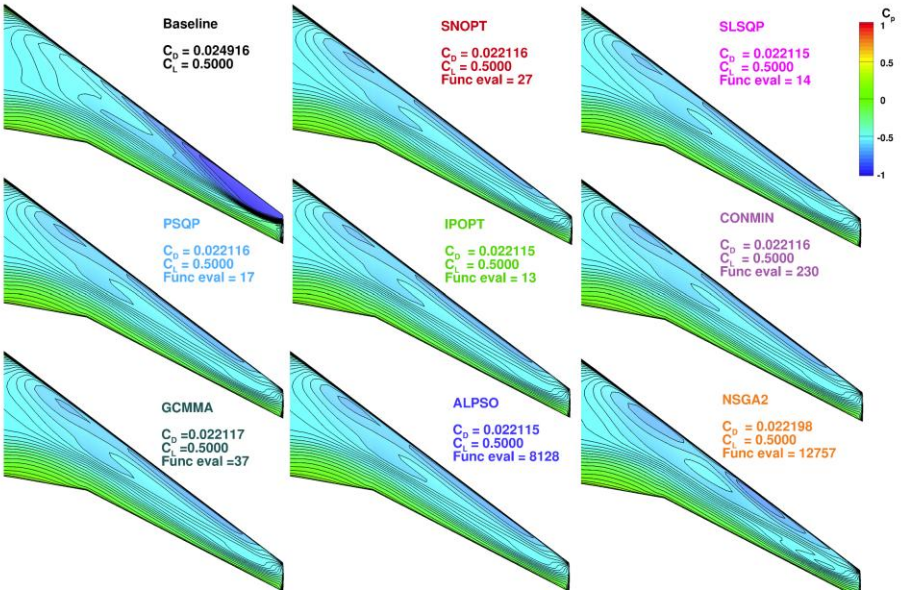



Figure 2.10: Performance of different optimisation algorithms for the optimisation of the twist distribution of an aircraft wing with eight design variables (Lyu et al., 2014); Gradients computed via the adjoint method

### 2.4.2 Gradient computation

Gradient-based optimisation requires the computation of the gradient of the objective function and potential constraint functions at every iteration of the optimisation process. The gradient is the direction and rate of fastest increase of a scalar-valued function $f$ with respect to its input variables $x_i$. If a function has more than one output the individual gradients are presented in the Jacobian matrix $J$, where the rows are the transposed gradients (Equation 1).

$$J_f = \begin{bmatrix} \nabla f_1{}^T \\ \vdots \\ \nabla f_m{}^T \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Equation 1: Jacobian matrix

Different methods for gradient computation exist. These include symbolic differentiation, finite differences, the complex step method, and algorithmic differentiation. While symbolic differentiation is widely used in calculus, it is unsuitable for numeric optimisation. Finite differences in contrast are extensively used in numerical optimisation. This approximation of the gradient is simple to implement and does not require anything but the function values. Therefore, they are the only option to obtain gradients of black box functions. Finite differences work by evaluating the function for an initial set of design variables, then perturbating the first design variable by a small step $h$ and then evaluating the function again. The difference in the output is

divided by the step size. This is shown in Equation 2 and must be repeated for any other design variable. The step is taken forwards which makes it a forward finite differencing scheme.

$$\frac{\partial f}{\partial x_1} = \frac{f(x_1 + h) - f(x_1)}{h}$$

Equation 2: Forward finite differencing scheme

While finite differences are simple to implement, they suffer from poor efficiency. A function with 70 input variables requires 71 function evaluations for gradient approximation. This is expensive. Furthermore, the accuracy of the gradients depends on the step size. A high step size is inaccurate due to a high truncation error. Therefore, the step size should be reduced maximally to reduce the truncation error. The smaller the step size however, the higher the error due to subtractive cancellation. This dilemma means that finite difference-based gradients can never be fully accurate and can hinder rapid convergence.

The complex step method is also an approximation of the gradient but uses a complex step as opposed to the real step in the finite difference method. Therefore, the inputs and outputs must be made complex variables, and the function must be solved using complex arithmetic. This requires the source code to be modified and means the function cannot be used as a black box. The method does not require a subtraction to be performed, so the only source of inaccuracy is the truncation error. This can be reduced by reducing the step size and vanishes if the step size is small enough. Similarly to finite differences, the time required to compute the gradient scales linearly with the number of input variables.

The third option for differentiation of complex functions (in the form of computer code) is called algorithmic differentiation. A computer code consists of a series of basic operations. These operations can be differentiated symbolically and then chained together using the chain rule. This results in gradients accurate to machine precision and can be performed manually or with algorithmic/automatic differentiation tools. Manual differentiation is time-consuming and error-prone, so automatic differentiation is to be preferred. Intuitively, the application of the chain rule is started from one input variable, differentiating to the output variables and then repeating this process for the remaining input variables. This is called forward differentiation and requires one partial derivative to be evaluated per input variable. Therefore, the number of function evaluations again linearly scales with the number of design variables, albeit less significantly when compared to the previous methods. This makes the method inefficient for problems with large numbers of input variables, but only few outputs. Conversely, the chain rule can also be applied in reverse, starting at one output, differentiating to the inputs. This requires the evaluation of one partial derivative per output variable, independent of the number of input variables. The evaluation of one partial derivative is thereby roughly as expensive as a forward function pass. This makes it

extremely powerful for sophisticated optimisation problems that usually have high numbers of input variables but only one or a few outputs. This method is known as reverse differentiation or the adjoint method and allows the optimisation of any number of design variables at small cost.

Two types of reverse algorithmic differentiation tools exist. These are based on either source transformation or on operator overloading. Source transformation tools transform the given code into the differentiated version of the code. This differentiated version can then be compiled and evaluated. Source transformation tools are for example ADIFOR (Bischof et al., 1992), OpenAD (Hovland et al., 2002), TAF (Giering et al., 2005) and TAPENADE (Hascoet and Pascual, 2013). In contrast, operator overloading methods differentiate the computer code at run time. This is achieved by recording all operations and resulting intermediate variables of the function during a normal evaluation. The information is stored on a so-called tape that depicts a simpler internal representation of the actual code. This internal representation can then be differentiated in reverse. The taping procedure is facilitated by replacing the typical floating point data type *double* with a specific data type. This datatype (often called *adouble* for active *double*) allows to save the intermediate values on the tape during a normal solve of the function. Operators such as +, * and sin are overloaded to record what operations were carried out. The overloaded operators and the special data type are provided by the algorithmic differentiation tool. The benefit of operator overloading is that a single code is maintained and developed, while source transformation requires the maintenance of the normal version and the differentiated version of the code. The downside of operator overloading is that it is slightly more computationally expensive. Exemplary operator overloading-based tools are JAX for Python (Schoenholz and Cubuk, 2020), CasADI for MATLAB (Andersson et al., 2012) and ForwardDiff.jl for Julia (Revels et al., 2016). The program to be differentiated in this project is FS-Equilibrium, which is written in C++. Several tools for the differentiation of C++ code exist. Examples are FAD (Aubert et al., 2001), CoDiPack (Sagebaum et al., 2019) and ADOL-C (Walther and Griewank, 2012). The most widely used tool for C++ programs is ADOL-C, so it was chosen for this project. For the full list of algorithmic differentiation tools see Bücker et al. (2024a).

ADOL-C is a program for **a**utomatic **d**ifferentiation by **o**ver**l**oading in **C**++. It is under sustained development at the Department of Mathematics at the Humboldt-Universität zu Berlin. The tool can compute derivatives of any order for a given source code in C or C++. It features both a forward and a reverse mode. For the differentiation with ADOL-C, several steps must be performed. First, the source code of the program must be fully available to ADOL-C, which means it cannot rely on pre-compiled libraries that are linked to the code. Parts of the code which have such a reliance must therefore be replaced beforehand. In a second step, every variable that influences the gradient must be declared as an active variable by assigning it the type *adouble* instead of the normal C++ double precision floating point datatype *double*. Variables that do not

influence the gradient can remain of the passive type. A special initialisation must be carried out to specify the input variables (also referred to as the independent variables) and the output variables (or dependent variables). The part of the computer program that should be differentiated (also called the active section) must be marked. Finally, the function to compute the gradient or the jacobian can be called. For a detailed introduction see Walther and Griewank (2022). For a simple example of differentiation with ADOL-C and the details of the differentiation of the lifting line method and the parametric model see Appendix A.

### 2.4.3 Algorithmic differentiation and adjoint optimisation examples

Algorithmic differentiation and adjoint optimisations have been performed in many fields including Biology (Hovland et al., 1997; ADIFOR), Medicine (Jee et al., 2005; ADOL-C), Physics (Kim et al., 2006; TAPENADE) and Engineering. For a data base of algorithmic differentiation applications see Bücker et al. (2024b). Examples in engineering range from electrical engineering (Hart et al., 2006; ADOL-C), over controller design (Röbenack, 2007; ADOL-C), to structural optimisation (Tadjouddine et al., 2006; ADIFOR). Most extensively however, it has been applied to aerodynamic shape optimisation. Potential flow-based adjoint optimisations have been performed with lifting line methods (Hodson et al., 2017), vortex lattice methods (Kontogiannis and Laurendeau, 2021) and panel codes (Sarikaya and Tuncer, 2022). Examples for CFD-based optimisations are Bischof et al. (1992)(ADIFOR), Schlenkrich et al. (2008)(ADOL-C) and Albring et al. (2015)(CoDiPack). Aerodynamic adjoint optimisations have also been performed with high level CFD models such as Large Eddy simulations (Roth and Ulbrich, 2013). Algorithmic differentiation has also been applied to CAD-tools (Computer-aided design) to allow the adjoint optimisation of geometries parameterised in full CAD-programs as opposed to small, tailored parametric models (Banović et al., 2018; Mykhaskiv et al., 2018; ADOL-C). This was extended to an industrial airfoil design tool including CAD- and CFD-models (Banović et al., 2020; ADOL-C). In the field of maritime engineering, adjoint optimisations have been applied for example to ship hulls (Lavimi et al., 2024), propeller blades (Lee et al., 2014) and hydrofoils. The applications to hydrofoils are discussed in detail in the next section.

## 2.5 Hydrofoil optimisation

Several hydrofoil optimisation approaches are discussed in the literature. Some of them consider section optimisation, others the optimisation of the spine- and the planform. In rare cases both are optimised together. Optimisation techniques include sampling, gradient-free optimisation, meta-modelling and adjoint approaches. This section introduces the published hydrofoil optimisation strategies with a focus on spine- and planform optimisation as section optimisation is not part of this study. It also discusses the approaches used in the America's Cup. These might be the most advanced approaches but are usually kept secret. Nevertheless, an interview with the America's Cup designer Martin Fischer has shone light on some of the strategies used by at least one America's Cup team.

Early hydrofoil optimisations have been performed mostly for section shapes with the aim to delay cavitation. A detailed description of this body of work is given by Garg et al. (2017). Examples are Brockett (1966), Eppler and Shen (1979), Shen and Eppler (1981) and Kinnas and Fine (1993). Brockett published design charts to identify optimum foils for different conditions in 1966. The underlying data was computed from steady two-dimensional flow investigations with an empiric correction for viscous effects. Eppler and Shen employed boundary layer and profile theory methods to find optimum shapes for delayed cavitation inception for symmetric sections in 1979 and asymmetric sections in 1981. Kinnas and Fine carried out analysis on hydrofoils with partially cavitating flows in 1993.

Section optimisations with gradient-free optimisation algorithms have been performed by Mishima and Kinnas (1996) and Zeng and Kuiper (2012). These were based on lower-fidelity methods and limited to five and ten design variables, respectively. A multi-objective section optimisation was performed by Kostas et al. (2017). It was based on a Boundary-Element method in conjunction with a genetic algorithm and optimised eight design variables. Djavareshkian and Esmaeili (2014) performed an optimisation of the thickness and camber of a foil section operated near the free-surface. The approach made use of an artificial neural network as surrogate model and a particle swarm optimiser. The foil section was modelled with Navier-Stokes and a volume of fluid method for the free-surface. Silva (2014) optimised the section for a C-Class catamaran with the aim to maximise lift, while preventing cavitation. The method used XFoil (Drela, 1989) and a gradient-free optimisation algorithm. Again eight design variables were optimised. Sacher et al. (2017) performed an optimisation of a hydrofoil section with deformable elements at the trailing edge for an AC50. The routine aimed to reduce the drag for four different sailing conditions with corresponding boat speeds and lift forces. Additionally, four constraint functions (for four different boat speeds) were implemented to prevent cavitation making use of criteria based on the cavitation number. XFoil in conjunction with a non-linear elasticity method was used to solve

the FSI problem. The optimisation was based on a derivative-free method. To reduce the computational time of the optimisation with 11 design variables, a surrogate-based approach was chosen. Peri (2023) has performed section optimisations for cases with no free-surface, with free-surface and with free-surface and cavitation for maximum lift to drag ratio with six design variables. The investigations were based on a RANS solver with a volume of fluid method for free-surface and a Schnerr-Sauer cavitation model (Schnerr and Sauer, 2001). The method used a meta model and regular sampling. A denser sampling was then performed around the area of the best performing foil from the first sampling. This process was repeated until the density of points was deemed sufficient.

Several methods have also been published regarding the design and optimisation of the spine- and planforms of hydrofoils. These optimisations are more specific to the type of hydrofoiling craft they are applied to and are performed for motor vessels as well as sailing yachts. Besnard et al. (1998) have performed an optimisation of single and biplane foils with varying numbers of struts for a large fast ferry. The global design variables were the foil depth, the thickness to chord ratio and the aspect ratio. These were optimised in a parametric study. However, each of these configurations also included a section optimisation aiming for maximum lift without cavitation and flow separation. The section optimisations were performed by a gradient-based optimisation algorithm with finite differences for gradient computation. The section modelling was performed with an interactive boundary layer approach, while the 3D foil geometries were analysed with a higher-order panel method. A finite element model was used to ensure structural integrity. Kandasamy et al. (2011) optimised the hydrofoil for a semi-foiling fast ferry with a parametric study on the foil rake, the foil position and the foil span. The study included the catamaran hull and was performed using unsteady RANS simulations. The aim was to reduce wave-making responsible for erosion in a narrow passage. The parametric study included four foil rake values and four longitudinal foil positions, as well as four demi-hull spacing values. The demi-hull spacing defines the span of the foil.

Parametric studies have also been applied to the hydrofoil design for sailing yachts. Ploe (2018) for example, performed a parametric study for a C-class catamaran hydrofoil. The approach was to design and numerically test families of foils in which only one parameter was altered. The four design variables described the spineform of the foil. 110 different foil geometries were generated and analysed using a hydrodynamic tool based on XFoil (Drela, 1989) and Athena Vortex Lattice (Drela and Youngren, 2021). The foil with the most favourable combination of heave stability and drag was considered the best performing. A comparable approach was employed by Guida et al. (2020) for the foils of the Nacra 17. Families of foils were created with varying cant and elbow angles, tip configurations and aspect ratios. Unsteady RANS simulations were performed for each individual foil using the commercial CFD software Star-CCM+ (Siemens Digital Industries

Software, 2020). The simulations were performed for a fixed value of the ride height, the leeway- and the rake angle. The foils were ranked according to their lift to drag- and side force to drag ratios.

Hydrofoils for sailing yachts have also been optimised using gradient-free optimisation algorithms. Meneghello et al. (2016) performed the optimisation of an L-foil for a sailing multihull. The hydrofoil performance was modelled using AVL (Athena Vortex Lattice) (Drela and Youngren, 2021). The aim was to minimise the drag of the foil while keeping side force and vertical lift constant. The optimisation was limited to seven design variables. The spineform of the foil was defined with a Bezier curve with four variables. A second curve with three parameters was used to describe the planform of the foil. The optimisation achieved an increase of lift to drag ratio by a factor of 2.3 compared to a baseline foil. The optimisation strategy was improved for better convergence by Alimohammadi et al. (2017).

Phelivan Solak et al. (2023) have performed a multi-fidelity surrogate-based optimisation for a kite foil type wing. The aim was to minimise drag while keeping vertical lift and side force constant. Ten design variables were used with the addition of the state variable heel. The construction of the surrogate model was conducted with many data points from a potential flow solver (796) and a small number of RANS simulations (4). The optimisation has required two weeks and returned an unfeasible result with a high prediction error.

A high fidelity surrogate-based optimisation was conducted by Ploe (2018). The approach was based on an unstructured finite volume Navier-Stokes solver. The optimisation was performed for two design variables, tip angle and the tip twist, and the state variable immersion. Rake and leeway angles were automatically adjusted to satisfy vertical lift and side force requirements. The method solved in two weeks on a large workstation.

Gradient-based optimisations have been performed by Tozzi (2004) for two-dimensional and three dimensional hydrofoils with potential flow models. The objective of the optimisations was the minimisation of the drag to lift ratio. Conjugate gradient and steepest descent algorithms were used to optimise a handful of design parameters. The gradients were computed using a central finite difference scheme. The author derived an adjoint formulation of the problem for future optimisation with large numbers of design variables.

Garg et al. (2017) have performed an adjoint hydrofoil optimisation. The approach used high-fidelity methods like Ploe but is considerably different. Garg's methodology is not limited to the hydrodynamics of the foil but also accounts for its structural behaviour. The FSI problem was solved using the CFD-solver ADflow (Kenway et al., 2019) coupled to the FEA-solver TACS (Kennedy and Martins, 2014). The adjoints were readily provided by the two solvers using

automatic differentiation. To optimise the foil geometry the gradient-based optimisation algorithm SNOPT (sparse non-linear optimiser) (Gill et al., 2005) was chosen. This allowed the optimisation of 210 design variables in a deformation-based geometric model. Constraint functions were utilised to account for cavitation, structural integrity and build tolerances. The routine was used for single- and a multi-point optimisation. The single-objective optimisation on 192 cores took 12h and achieved an overall efficiency gain of 12.4% compared to the baseline foil. The multi-point optimisation required 51h and attained an average efficiency gain of 8.53%. This lower gain is due to the foil having to perform across multiple points, each of which has a different optimum foil design.

Another multi-point adjoint hydrofoil optimisation was performed by Liao et al. (2022). The optimisation concerned a T-foil and had the aim to reduce drag. 198 design variables were used describing the planform (span, chord distribution and sweep), the twist distribution and the section shapes in several locations along the foil. Additionally, the rake of the foil was optimised. The foil performance was evaluated using ADflow (Kenway et al., 2019). 204 constraints were enforced to ensure for example the required amount of lift and the absence of cavitation. The optimal foil achieved a drag reduction of 18.9%. The optimisation required four days on 240 cores. A subsequent optimisation of a composite foil included the optimisation of the composite lay-up (Ng et al., 2025).

An adjoint optimisation was also performed by Luna Rossa for the hydrofoils of the AC75 during the 36$^{th}$ America's Cup (Fischer, 2021). The optimisation was based on a hydrostructural model consisting of a lifting line method and a finite element code. The adjoint versions of the models were implemented by hand. The aim of the optimisation was to minimise drag while enforcing the required amount of vertical lift and side force. Cavitation was considered with simplified empirical formulations. The multi-point optimisation was based on six conditions derived from VPP simulations with an initial hydrofoil (up- and downwind $V_{\mathrm{MG}}$-courses in three different wind speeds). These conditions remained constant throughout the optimisation. However, new foil candidates do lead to different conditions, e.g. different boat speeds and leeway angles, which is not accounted for and introduces inaccuracies. Therefore, a second VPP analysis was conducted after the first optimisation run was finished. The second VPP analysis was based on the optimal foil of the first run and was used to update the set of conditions. A second optimisation was then performed based on the updated conditions and started from the optimal foil of the first run.

## 2.6 Summary

Hydrofoils are the biggest performance differentiators on modern racing yachts such as the AC75. They significantly influence the speed of the yacht but also its state and trim. This calls for accurate and detailed optimisation.

Optimisation requires a performance model of the component/system to optimise and an optimisation strategy/algorithm. Different models for hydrofoil performance prediction are available. These range from potential flow-based methods for hydrofoil force modelling to dynamic simulators of the entire sailing yacht.

The large performance potential also calls for highly detailed design. High numbers of design variables should be optimised with gradient-based methods, where the gradient is computed using the adjoint method. The implementation of the adjoint method is best executed with algorithmic differentiation tools but requires access to and significant preparation of the source code.

Numerous hydrofoil optimisation approaches are proposed in the literature with different performance models and optimisation strategies. The most capable approaches are based on the adjoint method and either potential flow models or RANS methods. One such approach was developed by Luna Rossa. The optimisation was performed with an adjoint lifting line method. However, this isolated optimisation of the foil did not account for the changes in state and trim of the yacht during the optimisation. To reduce the resulting inaccuracies, the state and trim was updated after a first optimisation run and used as the base for a second.

The next step in hydrofoil optimisation must be to directly include a physics model of the entire yacht in an adjoint optimisation process. This eradicates the inaccuracies and opens the possibility for the optimiser to purposely identify foils that lead to higher performing sailing states.

# Chapter 3 VPP-Driven Parametric Design of AC75 Hydrofoils

*Tannenberg et al. (2023) "VPP-Driven Parametric Design of AC75 Hydrofoils", Journal of Sailing Technology 2023, volume 8, issue 1, pp. 161 – 181.*

https://doi.org/10.5957/jst/2023.8.9.161

# VPP Driven Parametric Design of AC75 Hydrofoils

**Rafael Tannenberg**
University of Southampton, UK, rafael.tannenberg@soton.ac.uk.

**Stephen R. Turnock**
University of Southampton, UK.

**Karsten Hochkirch**
DNV Ship Performance Center, Germany.

**Stephen W. Boyd**
University of Southampton, UK.

**Abstract.** Hydrofoils are a vital part of modern racing yachts such as the AC75, which was sailed in the 36th America's Cup and should hence be optimised thoroughly. The literature shows that hydrofoil design and optimisation usually focuses on the lift and drag characteristics in isolation of the yacht 'system'. Although these characteristics relate to hydrofoil performance, they do not directly translate to the performance of the yacht on the racecourse. In this paper we perform a parametric study of the main design variables of the hydrofoil that is based on a model of the entire yacht in the Velocity Prediction Program (VPP) FS-Equilibrium. The hydrofoil forces are modelled using an advanced lifting line method and empirical formulations for a bulb. This accurately captures the foil design influence on the boat´s performance. The VPP is coupled to a parametric model of the foil based on NURBS surfaces (Non-uniform rational B-Splines) which was used to systematically generate 72 different designs. The candidates were tested in three wind speeds for up and downwind performance. The best performing design has maximum span and anhedral angle, and minimum chord with some of the weight stored in a bulb. The study shows that the assessment of hydrofoils where the performance is measured in boat speed is an extremely valuable tool.

**Keywords:** Velocity Prediction; Hydrofoil; Parametric Study; Optimisation; AC75.

## Nomenclature

| | |
|---|---|
| $A_i$ | Area of segment $i$ [m$^2$] |
| $A_F$ | Foretriangle area [m$^2$] |
| $A_J$ | Jib area [m$^2$] |
| $A_M$ | Mainsail area [m$^2$] |
| $A_N$ | Nominal area [m$^2$] |
| $A_R$ | Aspect ratio |
| $C_D$ | Drag coefficient |
| $C_{\mathrm{Di}}$ | Induced drag coefficient |
| $C_{\mathrm{Dpj}}$ | Jib profile drag coefficient |
| $C_{\mathrm{Dpm}}$ | Mainsail profile drag coefficient |
| $C_H$ | Heeling force coefficient |
| $C_L$ | Lift coefficient |
| $C_{\mathrm{Lj}}$ | Jib lift coefficient |
| $C_{\mathrm{Lm}}$ | Mainsail lift coefficient |
| $C_R$ | Driving force coefficient |
| $c$ | Chord length [m] |

$d_z$ Sink (ride height) [m]
$E_{\mathrm{HM}}$ Mast height above shear [m]
$F_A$ Average shear height [m]
$\boldsymbol{F_i}$ Force at control point $i$ [N]
$f_i$ Function $i$
$f_x$ Force in $x$-direction [N]
$f_y$ Force in $y$-direction [N]
$f_z$ Force in $z$-direction [N]
$h$ Submersion [m]
$l_i$ Lagrangian multiplier $i$
$\boldsymbol{l_i}$ Length of segment $i$ [m]
$N_{\mathrm{i,p}}(u)$ Nonrational B-spline basis function in *u*-direction
$N_{\mathrm{j,q}}(v)$ Nonrational B-spline basis function *v*-direction
$m_x$ Moment around $x$-axis [Nm]
$m_y$ Moment around $y$-axis [Nm]
$m_z$ Moment around $z$-axis [Nm]
$p, q$ Degrees of NURBS surfaces in *u*- and *v*-direction
$P_{\mathrm{i,j}}$ Control points of NURBS-Surfaces
$R_F$ Frictional resistance [N]
$R_T$ Total resistance [N]
$S$ NURBS surface
$s$ Span [m]
$t$ Thickness [m]
$u, v$ Directions of NURBS surfaces
$\boldsymbol{V_i}$ Local velocity at control point $i$ [m s$^{-1}$]
$V_{\mathrm{MG}}$ Velocity made good [kts]
$V_S$ Boat speed [kts]
$V_{\mathrm{TW}}$ True wind speed [kts]
$\boldsymbol{V}_\infty$ Free-stream velocity [m s$^{-1}$]
$\boldsymbol{v_{ji}}$ Influence of horseshoe vortex $j$ on control point $i$ [m$^{-1}$]
$w_{\mathrm{i,j}}$ Weights of control points of NURBS surfaces
$(1 + k)$ Form factor

$\beta_a$ Apparent wind angle [°]
$\beta_t$ True wind angle [°]
$\Gamma_j$ Strength of horseshoe vortex $j$ [m$^2$ s$^{-1}$]
$\delta_e$ Elevator rake [°]
$\delta_f$ Flap angle [°]
$\delta_r$ Rudder angle [°]
$\theta$ Pitch angle [°]
$\lambda$ Leeway angle [°]
$\boldsymbol{\xi}$ Free variable vector
$\rho$ Density [kg m$^{-3}$]
$\tau$ flat parameter
$\varphi$ Heel angle [°]

ETNZ Emirates Team New Zealand
IMS International Measurement System
NURBS Non-uniform rational B-Splines
VPP Velocity prediction program

## 1. Introduction

The 36th America´s Cup saw the introduction of the first fully foiling monohull yacht class, the AC75, in Auckland in 2020. The yachts were sailed with great success and reached speeds above 53 kts that were not seen in the previous America's Cups. The AC75 was also appointed as the class for the 37th America´s Cup which will be held in Barcelona in 2024. The design class allows development in certain areas within the rules originally set out by the Royal New Zealand Yacht Squadron and the Circolo Della Vela Sicilia (2020). This includes the design of the hydrofoils which has proven to be extremely important as it has a huge influence on the performance of the yachts. Figure 3.1 shows the AC75 with its two main hydrofoils. The windward foil of the AC75 is canted out of the water to provide righting moment while the leeward foil is canted into the water to generate hydrodynamic lift. The foils are equipped with flaps that can be used to control the amount of hydrodynamic force generated. The cant angle of the leeward foil can be changed to adjust the ratio of generated side force to vertical lift as well as the amount of generated righting moment. The hydrofoil can be divided into two parts: the foil arm and the foil wing, both shown in Figure 3.1. The design of the foil arm is mainly fixed by the rule while the foil wing geometry is open to development within the `Foil box´ shown in Figure 3.2. The wing must be symmetric around the `Foil wing symmetry plane´ and must be placed within 10 and 12 m forward of the transom. In addition, the wing, the flaps and the internal control systems must weigh 921 kg to ensure the yacht has a reasonable amount of righting moment. To meet this weight requirement, while enabling a wing with low volume and low associated wetted surface area, the wings are made from a combination of steel and lead. The design space also allows a bulb to incorporate some of the weight. Emirates Team New Zealand (ETNZ) and American Magic chose to use bulbs and wings with smaller chord, while Luna Rossa and INEOS Team UK chose wings with larger chord and no bulb. Further differences included different taper ratios and anhedral angles. ETNZ's foil had no anhedral angle (T-Foil), while the other teams opted for Y-Foils as seen in Figure 3.1.

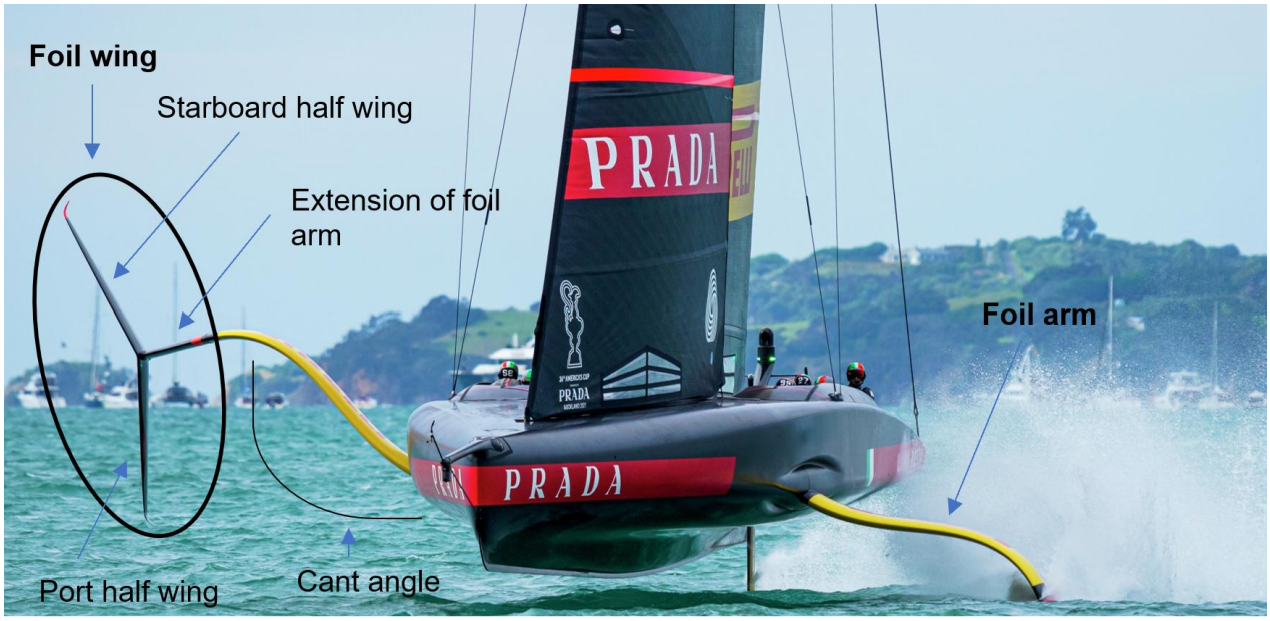



**Figure 3.1: AC75 Luna Rossa based on Gattini (2020)**

This work investigates 72 different hydrofoil designs with varying extension lengths (which translate to different anhedral angles), semi spans, root chords, taper ratios and wing rakes. The weight of every candidate is computed and a bulb is automatically sized to meet the weight requirement. The study includes design candidates that approximately represent the foils of the four teams in the last America´s Cup. The designs are evaluated in terms of their performance in up- and downwind conditions in the three true wind speeds ($V_{TW}$) 8, 11 and 14 kts using the VPP FS-Equilibrium. Although the AC75 rule has slightly changed for the next edition (Royal New Zealand Yacht Squadron and Royal Yacht Squadron, 2023), the

performance predictions in this study are based on the original rule to enable meaningful comparison with publicly available performance data from the last America´s Cup.

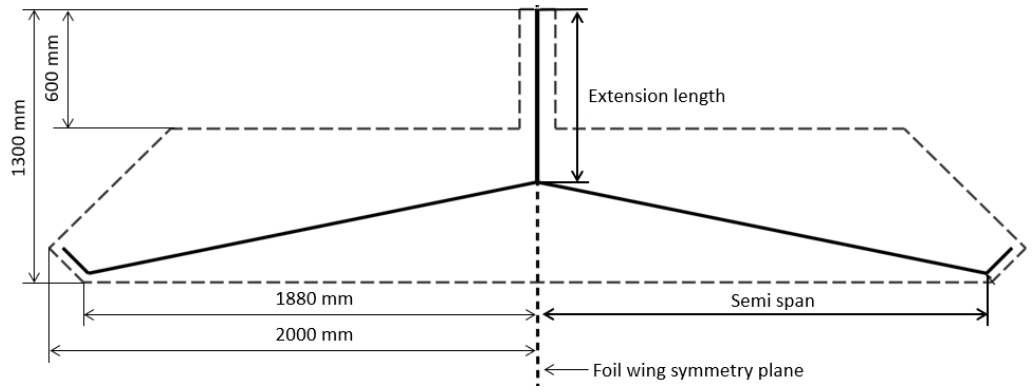



**Figure 3.2: Foil Box**

## 2. Velocity prediction model

FS-Equilibrium is a common workbench for the analysis of stationary and dynamic states of sailing yachts. The different forces acting on a yacht are modelled in so called 'force modules' for given conditions. A state for which the forces from the modules equate to zero in all degrees of freedom is called a valid steady sailing state. The program determines the steady sailing state by means of a Newton-Raphson method. The method alters the state variables of the yacht until force equilibrium is found. A Hooke-Jeeves algorithm can be used on top to maximise the performance of the yacht by optimising the trim variables. For example, common trim variables are the *flat*-parameter for the depowering of sails or the position of the crew. FS-Equilibrium has been successfully used for hydrofoiling boats in the past such as the C-Class Catamaran Groupama (Paulin et al, 2015), the International Moth (Eggert, 2018), the AC50 (Hansen et al, 2019) and the QFX Lake Racer (Melis et al, 2022). For more information on the VPP see Hochkirch (2018).

### 2.1 Solver set-up

The stationary mode of FS-Equilibrium was used to compute the steady sailing states of the AC75. It is limited to the states where the boat is fully foiling. This increases the robustness of the solver while it does not affect the evaluation of a foil´s performance because the only foil designs considered are those that can fully support the yacht in all evaluated conditions. However, the AC75 is sailed very differently compared to normal displacement yachts, which must be reflected in the set-up of the solver. In contrast to a traditional sailing yacht the trimming of the boat is much more dynamic, and a non-optimum trim might lead to a capsize instead of a valid sailing state which is just slightly slower. Table 3.1 shows the state variables (blue) and trim variables (grey) that the Newton-Raphson uses to balance the individual forces and moments for a conventional sailing yacht and the AC75. As for conventional yachts, the boat speed $V_S$, the leeway angle $\lambda$ and the rudder angle $\delta_r$ are employed to balance $f_x$, $f_y$ and $m_z$, respectively. $f_z$ however, must be balanced by a trim variable as the AC75 foils are not passively controlling the ride height. Hence, the state variable `ride height´ (or sink) $d_z$ has little influence on the heave forces as long as the yacht is fully foiling. The trim variable `flap angle´ $\delta_f$ must be employed instead. Furthermore, the moments around the x-axis ($m_x$) must be balanced by a trim variable. For normal yachts this balance is achieved by increasing the heel angle $\varphi$ which increases the righting moment from the keel and decreases the heeling moment from the sails. Increasing the heel angle of the AC75 however does not result in an increase in righting moment. Therefore, the moment must be balanced by a trim parameter controlling the forces from the rig which is the sail´s *flat*-parameter $\tau$. The pitching moment ($m_y$) of the

AC75 can either be balanced by the pitch angle $\theta$ as with normal yachts or with the elevator rake $\delta_e$. The elevator rake is constantly trimmed to maintain a target pitch angle and is hence better suited for balancing $m_y$ as it better reflects the sailing style of the AC75.

**Table 3.1: Different solver set-ups (state variables blue, trim variables grey)**

| | Conventional Yacht | AC75 |
|---|---|---|
| $f_x$ | boat speed $V_S$ | boat speed $V_S$ |
| $f_y$ | leeway angle λ | leeway angle λ |
| $f_z$ | sink $d_z$ | flap angle $\delta_f$ |
| $m_x$ | heel angle $\varphi$ | flat τ |
| $m_y$ | pitch angle $\theta$ | elevator rake $\delta_e$ |
| $m_z$ | rudder angle $\delta_r$ | rudder angle $\delta_r$ |
| | **Hooke-Jeeves** | **Fixed targets** |
| | flat τ | heel angle $\varphi$ |
| | crew x-position | pitch angle $\theta$ |
| | crew y-position | ride height (sink) $d_z$ |

The pitch angle as well as the ride height (sink) and the heel angle are however still relevant for solving the sailing state but are fixed targets and must hence not be optimised. The target values were estimated from race data from America's Cup (2020). It suggests that the boats are on average pitched bow down by around 2°, heeled to windward by around 1.5° and fly 0.8 m above the waterline (this results in the hull being clear of the water by about 0.4 m). This fixing of state variables does mean a slight simplification of the problem but greatly increases the robustness of the routine as the optimisation loop using the Hooke-Jeeves algorithm can be neglected completely and only the Newton-Raphson method has to be employed. The resulting problem is expressed using Lagrange multipliers ($l_i \neq 0$) within the functional

$$F = -V_S + l_0 \sum f_x + l_1 \sum f_y + l_2 \sum f_z + l_3 \sum m_x + l_4 \sum m_y + l_5 \sum m_z\,, \qquad (1)$$

where the first order condition for an extremum is

$$F \stackrel{!}{=} \min \rightarrow \nabla F \stackrel{!}{=} 0. \qquad (2)$$

This leads to a non-linear equation system with 12 equations:

$$\begin{aligned}
&f_0 = \frac{\partial F}{\partial V_S} \stackrel{!}{=} 0, \quad f_1 = \frac{\partial F}{\partial \lambda} \stackrel{!}{=} 0, \quad f_2 = \frac{\partial F}{\partial \delta_f} \stackrel{!}{=} 0,\\
&f_3 = \frac{\partial F}{\partial \tau} \stackrel{!}{=} 0, \quad f_4 = \frac{\partial F}{\partial \delta_e} \stackrel{!}{=} 0, \quad f_5 = \frac{\partial F}{\partial \delta_r} \stackrel{!}{=} 0,\\
&f_6 = \frac{\partial F}{\partial l_0} \stackrel{!}{=} 0, \quad f_7 = \frac{\partial F}{\partial l_1} \stackrel{!}{=} 0, \quad f_8 = \frac{\partial F}{\partial l_2} \stackrel{!}{=} 0,\\
&f_9 = \frac{\partial F}{\partial l_3} \stackrel{!}{=} 0, \quad f_{10} = \frac{\partial F}{\partial l_4} \stackrel{!}{=} 0, \quad f_{11} = \frac{\partial F}{\partial l_5} \stackrel{!}{=} 0.
\end{aligned} \qquad (3)$$

Using the following notation for the free variable vector

$$\boldsymbol{\xi} = (V_S, \lambda, \delta_f, \tau, \delta_e, \delta_r, l_0,\ l_1,\ l_2,\ l_3,\ l_4,\ l_5)^T, \tag{4}$$

the equation system in (3) can be written as follows:

$$f_i(\boldsymbol{\xi}) \stackrel{!}{=} 0 \qquad \text{for } i = 0 \dots 11. \tag{5}$$

Using a Taylor expansion as approximation for $f_i$ in the vicinity of a selected point $\sim\boldsymbol{\xi}_x$ and neglecting higher order terms yields a system of linear equations:

$$f_i(\boldsymbol{\xi}_x + \Delta\boldsymbol{\xi}) \approx f_i(\boldsymbol{\xi}_x) + \left.\frac{\partial f_i}{\partial \xi_0}\right|_{\boldsymbol{\xi}_x} \Delta\xi_0 + \cdots + \left.\frac{\partial f_i}{\partial \xi_{11}}\right|_{\boldsymbol{\xi}_x} \Delta\xi_{11} \qquad \text{for } i = 0 \dots 11, \tag{6}$$

where the partial derivatives of $f_i$ are computed using a forward finite differencing scheme. In each iteration a correction is calculated, and the procedure is stopped when a predefined accuracy is achieved, see Press et al. (1988) for more detail. A comparable approach was used by Patterson and Binns (2022), where the flap angle was balanced in a separate loop at every step of the Newton-Raphson.

## 2.2 Force modules

The forces of the AC75 are modelled with force modules for the gravitational forces, the aerodynamics of the sails and the hull, and the hydrodynamics of the foil, the rudder and the elevator. The gravity modules are used to model the forces and moments arising from the weight of the different components such as the hull and the crew. The weights of the components and crew were taken from the AC75 rule, while their lever arms were estimated from pictures. Inertia terms can be neglected completely for this stationary analysis.

**Table 3.2: Projected areas, Centres of effort and Drag coefficients of the AC75 hull**

| | Projected area [m²] | Centre of effort [m] | $C_D$ |
|---|---|---|---|
| **Front** | 7.4 | 8 | 0.25 |
| **Side** | 33.7 | 0 | 0.8 |
| **Top** | 79.8 | 0.5 | 0.8 |

The aerodynamic forces and moments arising from the hull are modelled with a simple force module based on drag coefficients in the three planes, the respective areas and centres of effort as shown in Table 3.2. The drag coefficients are taken from Hoerner (1965) for a comparable geometry whilst the areas and centres of effort are measured from a generic AC75 hull shown in the GUI of FS-Equilibrium in Figure 3.3.

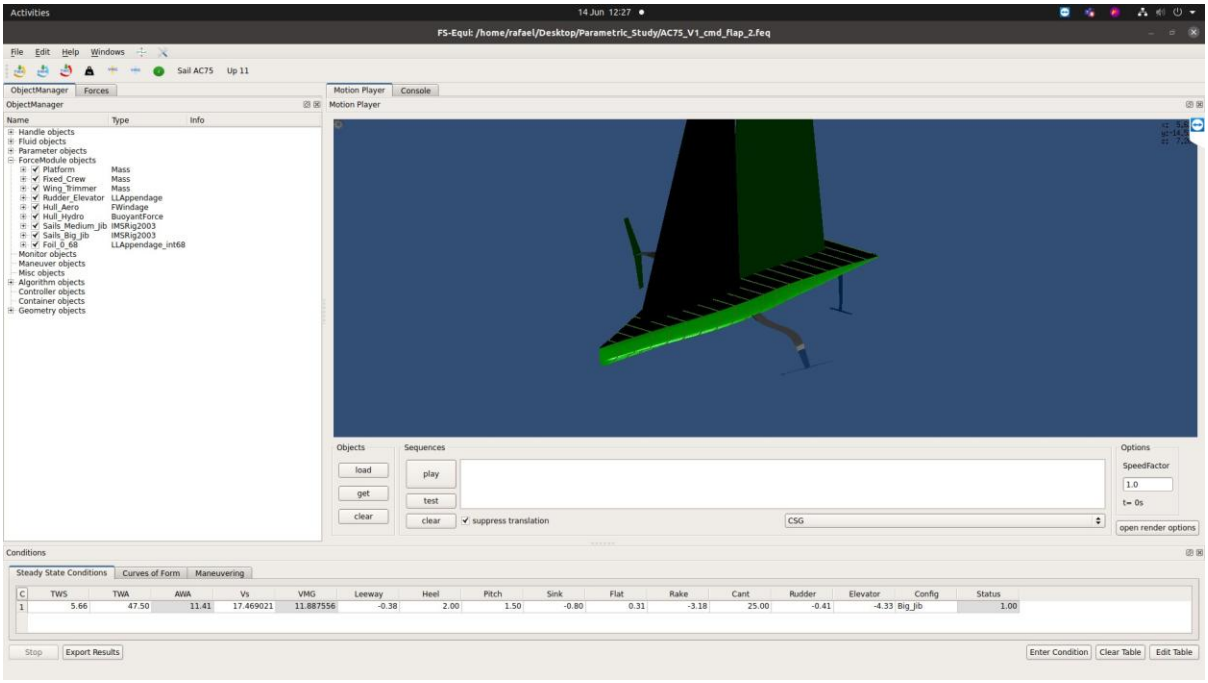



**Figure 3.3: Graphical User Interface of FS-Equilibrium showing the employed hull**

The sail plan of the AC75 consists of several conventional jibs and a soft double skin mainsail both end-plated to the deck. The double skin mainsail is a complex component that enables trimming of the sheeting angle, the twist, and the depth and position of the large possible maximum camber in several locations along the span of the sail. The variable camber allows the generation of high lift coefficients which are used at take-off and deep true-wind angles in lighter winds. The end-plated double skin sail in combination with the rotating mast is also significantly more efficient than a conventional mainsail in general. In addition, the top of the mainsail can be inverted to generate righting moment. To capture these advanced features and trim possibilities a sophisticated sail model is under development. However, at this stage, a simplified approach is taken by using the coefficient-based model for conventional rigs originated by Hazen (1980) presented below. The exact implementation follows that of the 2003 IMS VPP, that also accounts for the blanketing of the sails. A medium and a large Jib, in combination with the main sail, are modelled in so called 'configurations' and can be used according to the analysed conditions. The sail plan in this study corresponds to the sail plans of Luna Rossa, American Magic and INEOS Team UK which did not have the lower deck and extended luff like ETNZ. The details of the configurations are given in Table 3.3. The nominal area $A_N$ is computed as the sum of the mainsail area $A_M$ and the area of the foretriangle $A_F = 95.3$ m$^2$.

**Table 3.3: Configurations of the two sail plans**

| Configuration | Jib area $A_J$ $[m^2]$ | Main area $A_M$ $[m^2]$ | Nominal Area $A_N$ | Aspect Ratio $A_R$ |
|---|---|---|---|---|
| Medium Jib + Main | 64.2 | 142 | 237.3 | 4.37 |
| Big Jib + Main | 74.4 | 142 | 237.3 | 4.37 |

Based on the sail areas and the force coefficients of the individual sails (Jib: $C_{\mathrm{Lj}}$ and $C_{\mathrm{Dpj}}$, Main: $C_{\mathrm{Lm}}$ and $C_{\mathrm{Dpm}}$), which are depending on the apparent wind angle $\beta_a$, the lift and drag coefficients ($C_L$ and $C_D$) of the aggregate sail sets are computed with

$$C_L = \frac{C_{\mathrm{Lj}}\, A_J + C_{\mathrm{Lm}}\, A_M}{A_N}\, \tau \tag{7}$$

and

$$C_D = \frac{C_{\mathrm{Dpj}}\, A_J + C_{\mathrm{Dpm}}\, A_M}{A_N} + C_{\mathrm{Di}}. \tag{8}$$

$C_{\mathrm{Di}}$ is the induced drag coefficient:

$$C_{\mathrm{Di}} = {C_L}^2 \left(\frac{1}{\pi\, A_R} + 0.005\right), \tag{9}$$

where the aspect ratio is

$$A_R = \frac{(1.1\,(E_{\mathrm{HM}} + F_A))^2}{A_N}. \tag{10}$$

$E_{\mathrm{HM}}$ is the mast height above the shear line and $F_A$ is the average freeboard height, both in meters. For more detail see Larsson et al (2014) and Molland et al (2017). The aggregated coefficients are given in Figure 3.4 as a function of the apparent wind angle and are then transformed into the driving force coefficient

$$C_R = 1.3\left(C_L \sin(\beta_a) - C_D \cos(\beta_a)\right), \tag{11}$$

and the heeling force coefficient

$$C_H = C_L \cos(\beta_a) + C_D \sin(\beta_a). \tag{12}$$

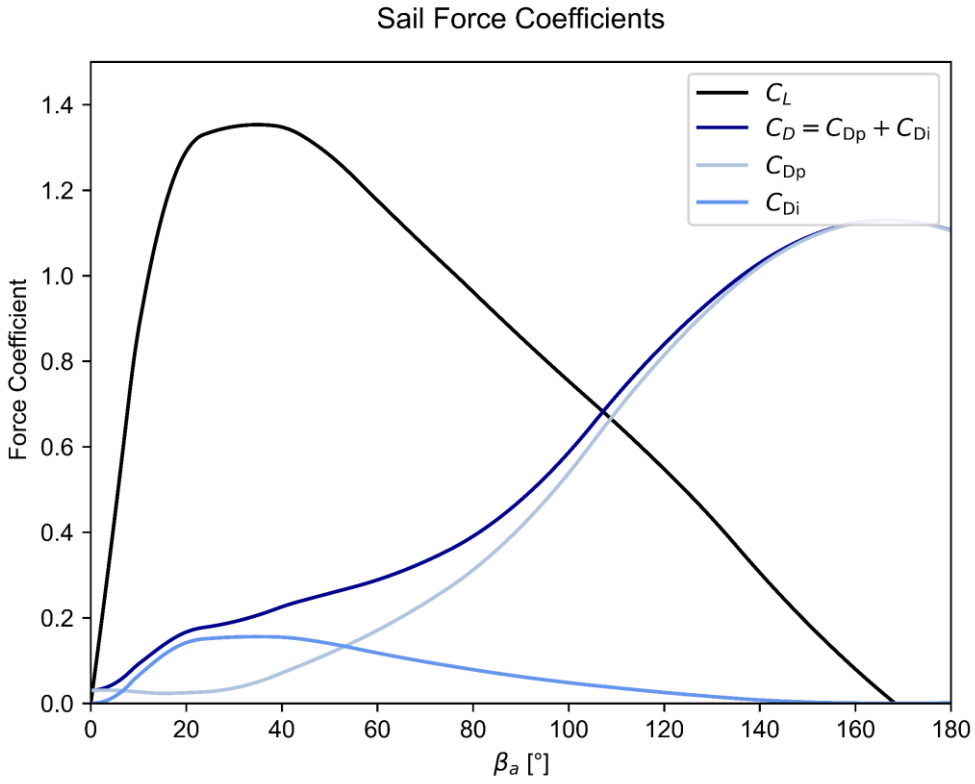



**Figure 3.4: Aggregated lift and drag coefficients of the configuration "Medium Jib + Main"**

As the forces predicted by the IMS model do not account for the increased efficiency due to the end-plated main, the rotating mast and the double skin sail, a factor on the driving force is included. According to Bergstrom and Ranzén, reported in Larsson et al (2014), the lift coefficient of a sail with a gap between the boom and the deck of 7% of the mast height leads to a 22% increase in drag coefficient and a decrease in lift coefficient of 13% compared to an end-plated sail. For an AC75-typical apparent wind angle of 13°, this results in a 20% gain in driving force due to end-plating. A significant additional performance increase is expected due to the rotating mast and double skins, not to mention the more advanced trimming possibilities. Therefore, a factor of 1.3 is applied to the driving force as shown in Equation 11. This achieves accurate results on most courses as presented in Section 5.

## 3. Hydrodynamic foil model

The forces and moments generated by the hydrofoil of the AC75 are modelled using an advanced lifting line method based on the formulation of Phillips and Snyder (2000) as summarized by Reid (2020). In contrast to the original lifting line method by Prandtl (1918), the approach employs horseshoe vortices at every station along the span and a vectorised form of the Kutta-Jukowski-Law. Every horseshoe vortex consists of a bound vortex and two semi-infinite vortices. The semi-infinite vortex sheet is oriented in the negative x-direction, so opposite to the direction of travel. The control points are located in the middle of every segment at quarter chord. Employing the vectorised lifting law includes the influence of the bound vortices on each other which is required to correctly model the effects of leeway, sweep or anhedral. Furthermore, it enables modelling the influence of multiple wings on each other. The method solves for the strength of each horseshoe vortex $\Gamma_j$ by relating two definitions of the force generated per segment. The force generated by every bound vortex is calculated from

$$\mathrm{d}\boldsymbol{F_i} = \rho\Gamma_i \boldsymbol{V_i} \times \mathrm{d}\boldsymbol{l_i}\,, \tag{13}$$

where the local velocity at each control point is computed with

$$\boldsymbol{V_i} = \boldsymbol{V}_\infty + \sum_{j=1}^{N} \Gamma_j \, \boldsymbol{v_{ji}}\,, \tag{14}$$

where $\boldsymbol{v_{ji}}$ is the induced velocity of horseshoe vortex $j$ at control point $i$ normalised by the vortex strength. Relating the vectorised vortex lifting law to a definition of the lift based on the sectional lift coefficient results in a non-linear system of equations which is solved iteratively. Due to the non-linearity, the method can account for the effects of stall but is potentially unstable. Therefore, the approach was reduced to a linear problem by applying a linear lift-curve slope to increase robustness and reduce computational time. This yields the linear system of equations

$$\rho\Gamma_i \left| \left( \boldsymbol{V}_\infty + \sum_{j=1}^{N} \Gamma_j \, \boldsymbol{v_{ji}} \right) \times \mathrm{d}\boldsymbol{l_i} \right| - \frac{1}{2}\,\rho\,{\boldsymbol{V}_\infty}^2\, C_L(\boldsymbol{V_i})\mathrm{d}A_i = 0. \tag{15}$$

For a more general introduction to lifting line methods and other numerical models for hydrofoils see Molland and Turnock (2021). Each part of the hydrofoil is discretised using 40 segments with a cosine-distribution as recommended by Phillips and Snyder (2000) for the best trade-off between accuracy and computational time. The parts of the hydrofoil that are above the waterline are `cut off´ and not included when computing the forces (Figure 3.5).

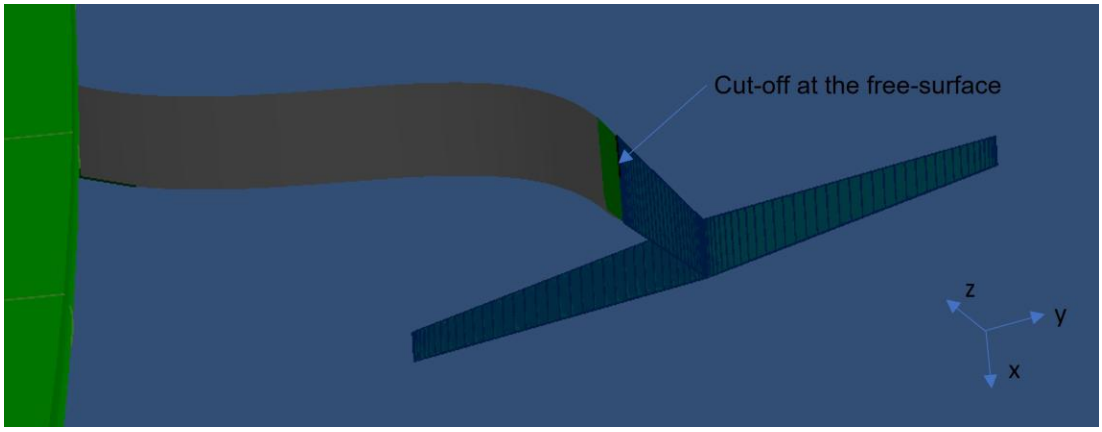



**Figure 3.5: AC75 foil discretised with 40 stations per foil part and cut-off at the free surface**

The loss in lift due to the free surface is modelled using a second transformed foil according to the biplane analogy (Faltinsen, 2005) shown in Figure 3.6a. The analogy is applicable if the submerged Froude number

$$F_{Nh} = \frac{V_S}{\sqrt{gh}} \tag{16}$$

is higher than $10/\sqrt{h/c}$, where $V_S$ is the velocity, $g$ the gravitational field strength, $h$ the submersion and $c$ the chord. For the average tip chord in the study $c = 0.25$ m and the submersion of the outboard wing at the tip $h \sim 0.25$ m ($h/c = 1$), the required minimum submerged Froude number is 10. The actual submerged Froude number for the average boat speed in the study $V_S = 19$ m s$^{-1}$ is 12.13 at the outboard tip and therefore satisfies the criterion. At the root of the half wing, where the average submersion is 0.82 m and the average chord is 0.6 m ($h/c = 1.37$), the criterion of 8.54 is not satisfied ($F_{Nh} = 6.69$), but the effect of the free-surface is smaller due to the higher $h/c$ ratio (see Figure 3.6b). More importantly, the loss in lift prior to the required submerged Froude number is even higher, so there is no harm in applying the biplane-image as long as $F_{Nh}$ does not tend to zero. Close to $F_{Nh} = 0$, the lift is increased due to the free surface, but such low Froude numbers were not experienced in the study.

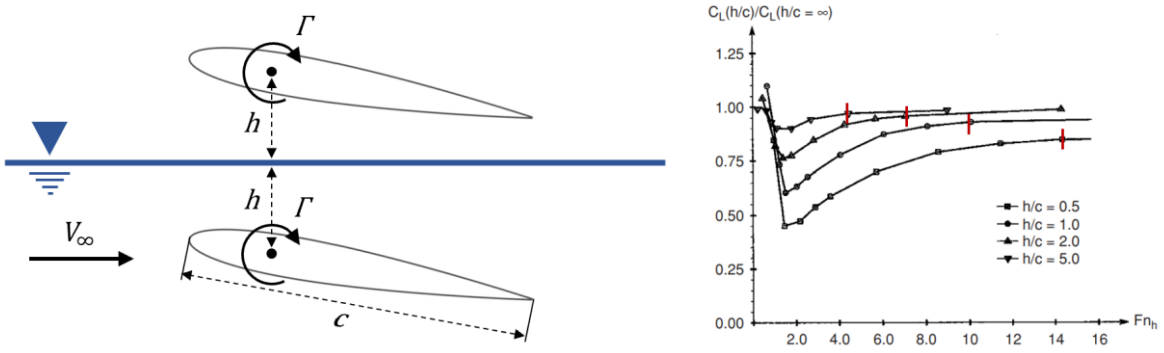



**Figure 3.6: (a) Biplane transformation around the free-surface plane; (b) Change in lift coefficient as function of the submerged Froude number for different $h/c$-ratios, red bars represent $F_{Nh} = 10/\sqrt{h/c}$; Derived from Faltinsen (2005)**

The sectional lift coefficient at every station is computed from the lift-curve-slope and the zero-lift-angle. The zero-lift-angle is provided in the form of a response surface as a function of the

flap angle and the span $s$ of the foil using a sequential interpolation based on C-splines. Similarly, the sectional profile drag coefficient per station is computed from a response surface depending on the flap angle, the Reynolds number, the lift coefficient and the span. The response surface is based on an inverted distance weighting function. The underlying data of both surfaces was computed using viscous XFoil simulations. For the extension of the foil arm, which has to be symmetric according to the AC75 rule, a NACA 0008 section was used while the foil wings have an asymmetric Eppler 874 hydrofoil section. Figure 3.7 shows the two response surfaces of the Eppler 874 section with the corresponding data points. The green colour indicates a fitting error below 1%. The range of points is chosen such that the AC75 foil will operate in this range at all times. Wave-making drag, spray drag, and junction drag are not/not yet considered.

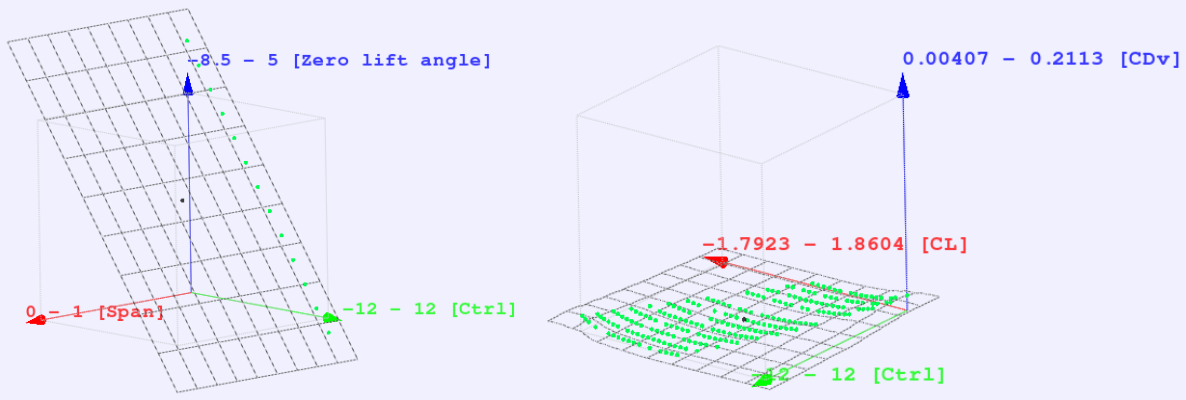



**Figure 3.7: Response surfaces for the Eppler 874 section, zero-lift-angle (a), sectional profile drag $C_{Dv}$ (b) at $Re = 3 \times 10^7$ and $s = 0$; Ctrl corresponds to flap angle**

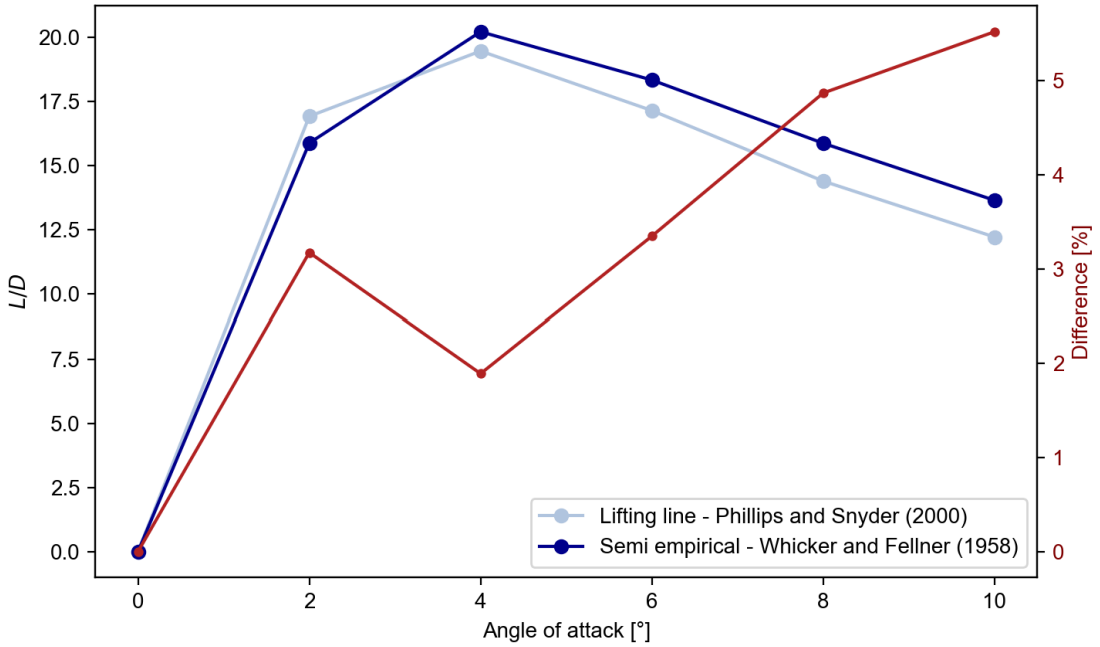



**Figure 3.8: Comparison of lift to drag ratios ($L/D$) as a function of angle of attack computed by the advanced lifting line method and Whicker and Fellner (1958) at $Re = 1.8 \times 10^6$**

The advanced lifting line method is validated with the semi empirical equations for low-aspect-ratio foils presented by Whicker and Fehlner (1958). The chosen geometry has an aspect ratio of 3, a taper ratio of 0.45 and 0° sweep at the quarter chord line. A NACA0015 section is used

with no twist along the span. Figure 3.8 shows the resulting lift to drag ratios as function of the angle of attack plus the difference of the two models. It shows that the difference is usually below 5% and only exceeds it at 10° angle of attack.

The lifting line method returns the hydrodynamic forces and moments generated from the lifting surfaces. Additionally, the force module computes the weight of the current foil design based on a weighted average density of 8305 kg $m^{-3}$ of the involved materials (75% high density steel, 20% lead and 5% hydraulic oil to represent the control systems). If the foil design does not meet the required 921 kg, an ellipsoidal bulb is automatically sized to meet the requirement. If the foil is exceeding the required weight, it is marked as too heavy. The module also computes the gravitational forces and moments of the foil which are added to the hydrodynamic forces. The centre of gravity of the foil is hereby computed based on the current design. If a bulb is included, the additional hydrodynamic drag is computed using a form factor approach (Equation 17). The frictional resistance $Rf$ is calculated from the area of the bulb and the friction drag coefficient derived from the ITTC line (Specialist Committee of 23rd ITTC, 2002). The form factor $(1+k)$ is computed based on the chord $c$ and the thickness $t$ of the bulb according to Equation 18. The bulb drag and the corresponding moments are added to the forces and moments computed previously and returned to the Newton-Raphson method as a whole. Very small bulbs might lead to a slightly lower overall drag opposed to an increased drag as they reduce junction drag, this however cannot be modelled with the lifting line method.

$$R_T = R_F(1+k) \tag{17}$$

$$(1+k) = 1 + 2\frac{t}{c} + 60\left(\frac{t}{c}\right)^4 \tag{18}$$

## 4. Parametric model

The parametric model of the hydrofoil serves to generate the shape of a hydrofoil from a set of design variables. The foil geometry is provided to the lifting line method in the form of 'ruled surfaces' which are based on NURBS surfaces. NURBS surfaces are defined as

$$S(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)w_{\mathrm{i,j}}P_{\mathrm{i,j}}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)w_{\mathrm{i,j}}} \quad \text{for} \quad 0 \le u,v \le 1, \tag{19}$$

where $P_{\mathrm{i,j}}$ are the control points, $u$ and $v$ the directions alongside the surface ($u$ spanwise, $v$ chordwise) and $p$ and $q$ are the degrees in the $u$- and $v$-direction, respectively. $N_{\mathrm{i,p}}(u)$, $N_{\mathrm{j,q}}(v)$ are the nonrational B-spline basis functions and $w_{\mathrm{i,j}}$ are the weights of the control points. For more detail see Piegl and Tiller (1997).

From the NURBS surfaces the lifting line method derives the relevant geometrical information such as the chord and the twist along the span of the foil. Figure 3.9 on the left shows an AC75 foil described by 86 input variables which are used to generate the underlying NURBS surfaces developed as part of this research. This sophisticated model is detailed enough to re-create the shapes of the hydrofoils seen in the 36th America's Cup. Two of which are shown in Figure 3.9 on the right representing geosims of the ETNZ (upper) and Luna Rossa (lower) foils, respectively. Although very complex foil shapes can be generated, the sheer number of input variables makes it impossible to examine parametrically. This sophisticated parametric model will be used in an adjoint optimisation in the future that allows the optimisation of hundreds of design variables in a very effective manner.

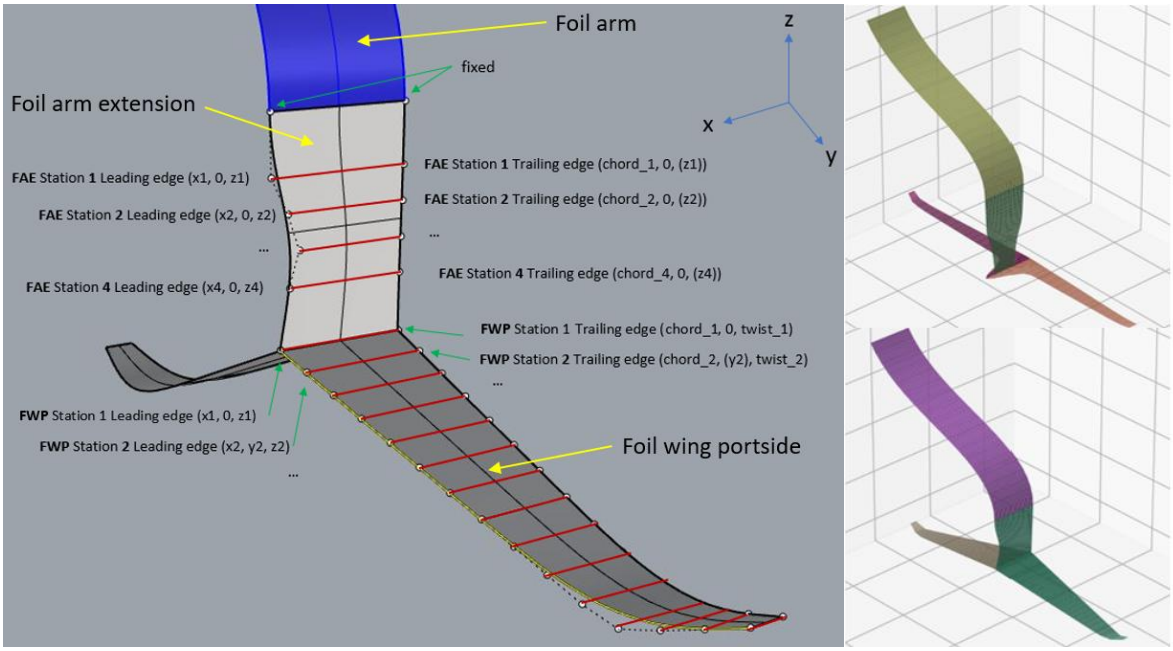



**Figure 3.9: NURBS-based parametric model of the AC75 foil using 86 design**

For the purposes of this paper however, a simplified NURBS-model was developed as shown in Figure 3.10. The foil is defined through the extension length, semi span, root chord, taper ratio and wing rake which are examined in the parametric study. The model exploits the symmetry requirements, which means that the portside half wing is mirrored to define the points of the starboard half wing. The z-position of the wing tips is fixed on the bottom of the foil box to enable the definition of the spine form with just the extension length and the semi span. Winglets are not considered. Both parametric models are based on the C++ library TinyNURBS (Jayaraman, 2022) and use the python library NURBS-python (Bingol, 2022) for visualisations. The parametric model is coupled to the lifting line method internally for increased computational efficiency and user-friendliness.

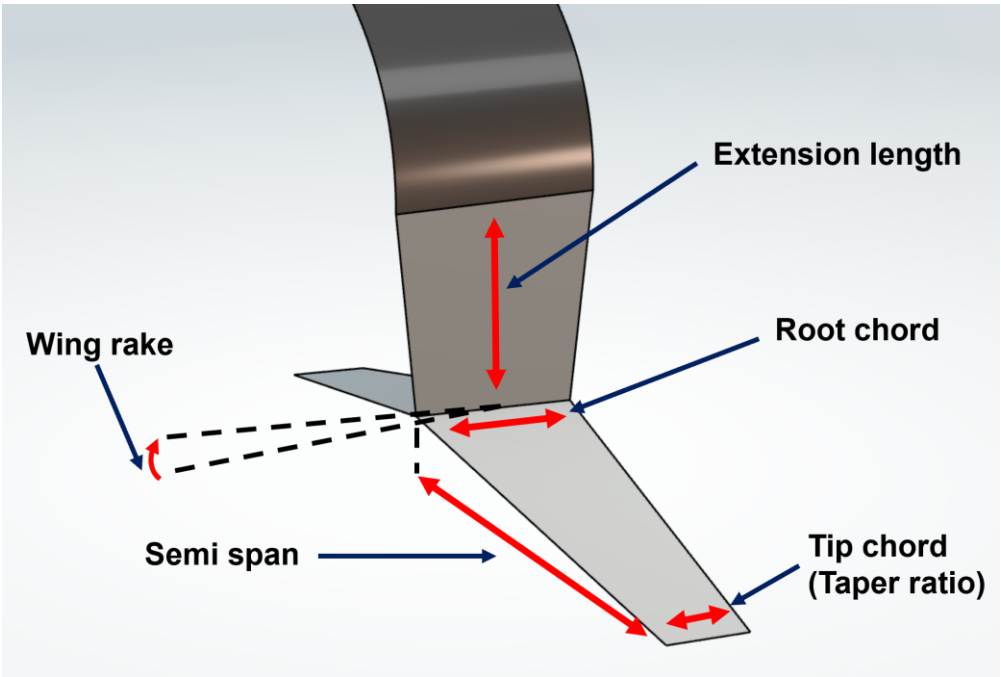



**Figure 3.10: Simple parametric model of the AC75 hydrofoil showing the control points employed to define the hydrofoil geometry**

## 5. VPP validation

The AC75 VPP model is validated with real live data recorded during the last America's Cup (America´s Cup, 2020). Ten races were sailed in light to moderate conditions, of which three races are selected to cover this range. In race one the average wind speed was around 13 kts, while races three and seven were sailed in 9 and 11 kts, respectively. From these three races, situations were chosen where both teams (ETNZ and Luna Rossa) sailed in comparable conditions. This means that there were no visible differences in wind speed and that the yachts were far enough apart to not interfere with each other. Per race, four such situations are selected, two sailing upwind and two sailing downwind. The results are presented in Table 3.4, where true wind speed $V_{TW}$, boat speed $V_S$ and velocity made good $V_{\mathrm{MG}}$ are given in knots and the true wind angle $\beta_t$ is presented in degrees. In addition to the $V_{\mathrm{MG}}$ measured at the specified time, the variation in $V_{\mathrm{MG}}$ in a ten second window is provided in brackets. For comparison, two foils similar to those of ETNZ and Luna Rossa were modelled and tested. The designs are based on the simplified parametric model, so they do not perfectly represent the details of the real foils, but they do have the similar root chords, taper ratios and anhedral angles. The hull, the sail plan and the remainder of the yacht are however not altered. The velocity polars of the two foils are computed for the true wind speeds 9, 11, and 13 kts. A macro is defined in FS-Equilibrium that initiates the force balance and plots the polars for every wind speed. The larger jib is used in 9 and 11 kts $V_{\mathrm{TW}}$, while the medium sized jib is used in 13 kts.

**Table 3.4: America's Cup race data from ENTZ and Luna Rossa (LRPP)**

| Race | Time | ETNZ $V_{\mathrm{TW}}$ [kts] | ETNZ $V_S$ [kts] | ETNZ $V_{\mathrm{MG}}$ [kts] | ETNZ $\beta_t$ [°] | LRPP $V_{\mathrm{TW}}$ [kts] | LRPP $V_S$ [kts] | LRPP $V_{\mathrm{MG}}$ [kts] | LRPP $\beta_t$ [°] |
|---|---|---|---|---|---|---|---|---|---|
| **1** | 01:27 | 13 | 36 | 24 (22 - 25) | 48.2 | 13 | 34 | 24 (24 - 27) | 45.1 |
| | 01:50 | 14 | 38 | 27 (26 - 27) | 44.7 | 14 | 36 | 24 (24 - 26) | 48.2 |
| | 13:31 | 13 | 42 | 35 (34 - 36) | 146.4 | 13 | 42 | 36 (34 - 36) | 149 |
| | 21:14 | 13 | 43 | 37 (36 - 37) | 149.4 | 13 | 43 | 36 (34 - 36) | 146.8 |
| **3** | 02:29 | 8 | 33 | 21 (20 - 22) | 50.5 | 9 | 30 | 17 (16 - 20) | 55.5 |
| | 11:57 | 9 | 33 | 19 (19 - 22) | 54.8 | 10 | 29 | 15 (12 - 21) | 58.9 |
| | 06:08 | 10 | 38 | 30 (29 - 30) | 142.1 | 10 | 35 | 27 (27 - 28) | 140.5 |
| | 16:28 | 9 | 36 | 31 (29 - 32) | 149.4 | 10 | 36 | 27 (26- 28) | 138.6 |
| **7** | 02:30 | 11 | 36 | 24 (23 - 25) | 48.2 | 11 | 32 | 21 (17 - 21) | 54.5 |
| | 09:30 | 11 | 32 | 24 (21 - 24) | 41.4 | 11 | 31 | 22 (18 - 23) | 44.8 |
| | 05:42 | 11 | 40 | 34 (33 - 35) | 148.2 | 11 | 40 | 28 (27 - 29) | 134.4 |
| | 15:10 | 11 | 42 | 33 (33 - 34) | 141.8 | 11 | 42 | 30 (28 - 30) | 135.6 |

Figure 3.11 shows the velocity polars for both yachts which also include the measured points from Table 3.4 (single points with red edging). Firstly, it can be seen that all velocity polars are smooth curves which suggests a very robust solver set-up. Secondly, the presented polars for 11 kts $V_{\mathrm{TW}}$ show very good agreement of the predicted and the measured performances. This

is also true for the polars in 13 kts. The polar of Luna Rossa in 9 kts shows excellent agreement on upwind courses but is underperforming in the downwind cases. This can be explained by the use of the IMS sails aero module which does not model the high achievable camber of the double skin main sail. This feature allows the generation of high lift coefficients and hence high drive forces, beneficial in these conditions. This effect is less visible at higher wind speeds or when sailing upwind, where the boats are operated with less camber. The polar of ETNZ in 9 kts $V_{\mathrm{TW}}$ is underperforming on all courses which can be explained by the sail plan which is based on those of the challengers and does not have the extended sail area the ETNZ boat had in reality. This extended sail area is assumed advantageous at low wind speeds. The discrepancy on downwind courses is higher which is again due to not modelling the deep camber. Similar trends were observed by Patterson and Binns (2022).

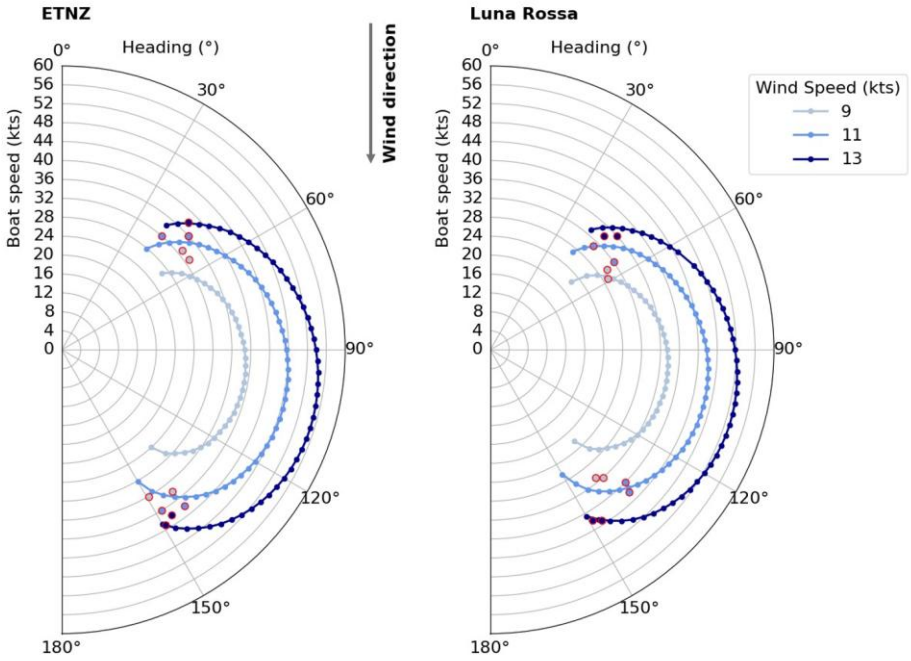



**Figure 3.11: Velocity polars of the two foil designs of ETNZ and Luna Rossa compared to the real-life data points from Table 3.4 (points with red edging)**

While the generated polars show very good agreement with the measured data, they are still heavily influenced by the tuning of the aerodynamic model of the sails (factor of 1.3 on driving force) which introduces uncertainty. They also lack in perfectly modelling downwind performance. As a consequence, the development of an aerodynamic sails model that can reflect the AC75 sails accurately is part of current work. The module will be based on a machine learning model trained with a panel code.

## 6. Parametric study

The parametric study examines the foil design parameters extension length, semi span, root chord, taper ratio and wing rake. The spine form of the foil is defined with the extension length and the semi span while the plan form of the foil is specified with root chord and taper ratio. The fixed rake of the wing is defined by the wing rake variable. The study is carried out for any combination of the variable values specified in Table 3.5. The parameter set-up and the values are chosen to maximise the understanding of the parameters, while keeping the number of required simulations as low as possible. The range of variable values results in a total of 72 design candidates. The simple parametric model originally also incorporated a variable to define the sweep of the foil. However, initial results have shown that any amount of sweep is reducing the performance of a design compared to its non-swept version. Therefore, this variable was excluded for the sake of simplicity. The performance of all candidates is evaluated

on up- and downwind courses in three different wind speeds, 8, 11 and 14 kts $V_{\mathrm{TW}}$, best representing the range of conditions recorded during the last America´s Cup. The courses sailed in the up- and downwind conditions are selected according to the courses that resulted in the highest $V_{\mathrm{MG}}$ in Section 5. The leeward foil is canted out by 25° to maximise righting moment while being just fully submerged. Running the study was automated using a python script and the batch mode of FS-Equilibrium.

**Table 3.5: Parameter values**

| Parameter | Values |
|---|---|
| Extension length | 600 mm, 950 mm, 1300 mm |
| Semi span | 1750 mm, 1880 mm |
| Root chord | 400 mm, 800 mm |
| Taper ratio | 0.25, 0.5 |
| Wing rake | 0.0°, 2.0°, 4.0 ° |

## 6.1 Results

Figure 3.12 and Figure 3.13 show the up- and downwind performances of the 72 foil candidates in 11 kts $V_{\mathrm{TW}}$. Figure 3.14 presents the upwind performance in 8 kts $V_{\mathrm{TW}}$, the slowest of the assessed conditions, and Figure 3.15 the downwind performance in 14 kts $V_{\mathrm{TW}}$, the fastest condition. The legend of the figures can be used to determine what spine and planforms the candidates have. These are defined by the extension length, the semi span, the root chord and the taper ratio. The figures show three candidates within every taper ratio band. These three designs share the same spine and planforms but have different wing rakes. The first candidate within every band has a wing rake of 0.0°, followed by the rakes 2.0° and 4.0°. If a point is not shown, no foiling equilibrium could be found. All foils weigh exactly 921 kg.

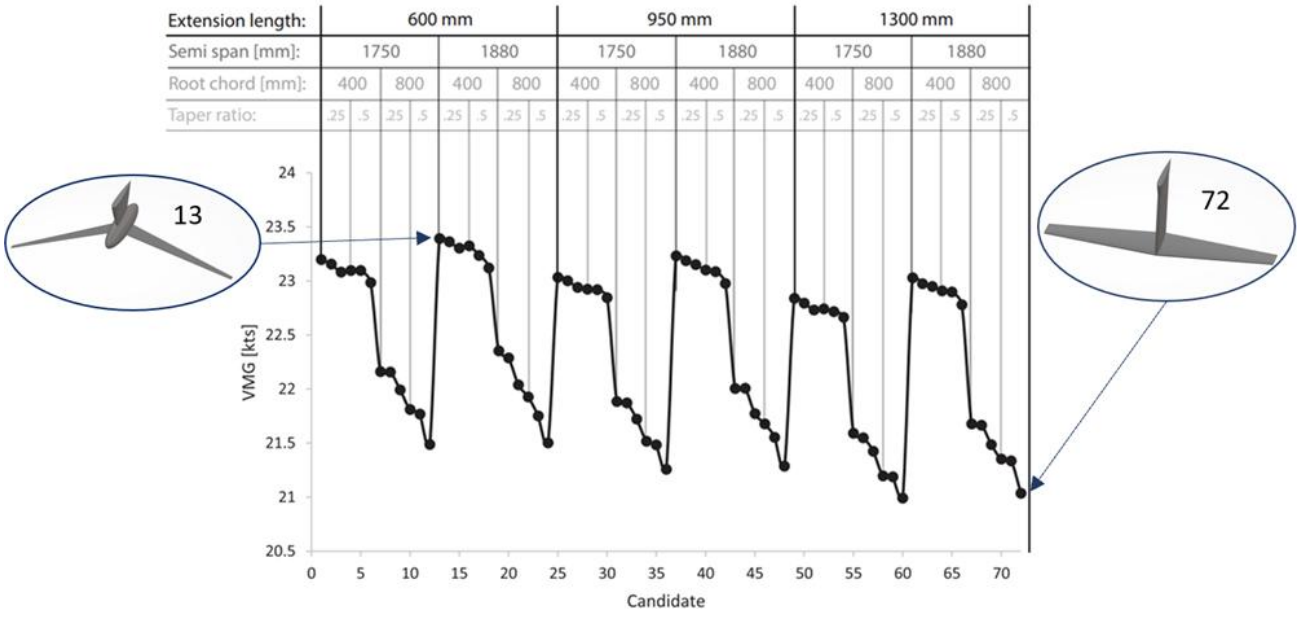



**Figure 3.12: Foil performance in 11 kts upwind at $\beta_t = 47.5°$**

Figure 3.12 and Figure 3.13 show that foils with lower extension lengths and hence higher anhedral angles perform better than foils with higher extension lengths. This is true for up- and downwind conditions in all three evaluated wind speeds (see also Figure 3.14 and Figure 3.15) and can be explained by two effects. Firstly, the extension of the foil is mainly only used to connect the arm and the wing and is not actively employed to generate side force or vertical

lift. Therefore, it is better to store the required weight in a larger wing that is then operated at a lower wing rake or flap angle or in the bulb which has a lower wetted surface area for the same volume. Storing the required weight in a longer extension only causes additional drag. Secondly, the foils with higher extension length generally have a longer spine than the same foils with lower extension. Having a fixed volume, the components are therefore automatically longer and slenderer, which means that their ratio of wetted surface area to stored weight is higher.

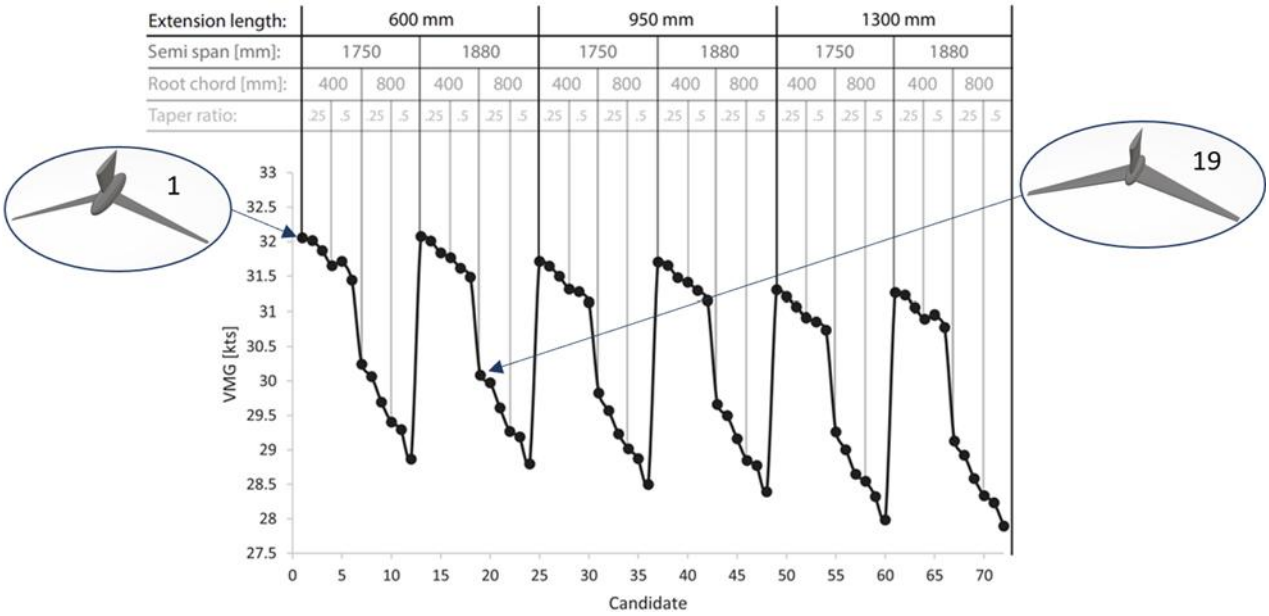



**Figure 3.13: Foil performance in 11 kts downwind at $\beta_t = 140°$**

A further benefit of the foils with anhedral angle is that the two flap segments (one on each half wing) can be used independently. When the foil is canted out, the ride height can be adjusted with the flap of the almost horizontal outboard half wing and the leeway angle can be controlled with the flap of the almost vertical inboard half wing. Furthermore, it also allows to shift the centre of effort further outboard, and hence to increase the righting moment generated by the foil. However, these effects cannot yet be reflected in the AC75 VPP model. Nevertheless, a shorter extension also means that the outboard side of the wing operates closer to the free surface when canted out compared to a higher extension foil. This does reduce the lift generated and increases the drag due to wave-making. It potentially also promotes cavitation and ventilation. The reduced lift is modelled using the biplane image, which is best suited for the design space and conditions explored and seems to not outweigh the benefits of the high anhedral angle. Wave-making drag, and potential cavitation and ventilation are not included. This means that in reality a foil with medium extension could outperform the lowest extension foil and could be an explanation why the three challengers in the America´s Cup with Y-foils did not chose to have the highest possible anhedral angle. Another explanation could be that the constraint on centre of gravity in the AC75 rule is active above a certain anhedral angle.

The semi span of the foil influences both the planform area of the foil and the aspect ratio. At lower boat speeds such as sailing upwind in 11 kts and up- and downwind in 8 kts $V_{\mathrm{TW}}$ the higher semi span performs better as it reduces the dominant induced drag (Figure 3.12 and Figure 3.14). At higher boat speeds (sailing downwind in 11 kts and up – and downwind in 14 kts $V_{\mathrm{TW}}$, Figure 3.13 and Figure 3.15) the slightly lower span performs better. In these conditions the profile drag is dominant which can be reduced by decreasing the area of the foil. This outweighs the higher induced drag caused by the lowered semi span. A foil where the root chord would be further reduced instead of the semi span would however still perform better, as it would reduce viscous drag while keeping induced drag constant.

The root chord has the highest influence on the yacht´s performance in this study. This can be partially explained by the high difference between the root chord values, especially compared to the values of the semi span. It is however also caused by the large effect of the root chord

on the area of the foil, as it also influences the tip chord via the taper ratio. The smaller root chord performs significantly better in 11 and 14 kts and going downwind in 8 kts $V_{\mathrm{TW}}$ due to the significantly lower area (Figure 3.12, Figure 3.13 and Figure 3.15). This also applies to most candidates in the slowest condition (sailing upwind in 8 kts $V_{\mathrm{TW}}$, Figure 3.14). However, some candidates (Candidates 2, 26, 42, 49, 50, 51 and 62) cannot support the weight of the yacht in this condition due to their low planform area, especially if they also have the lower semi span and the lower taper ratio. As the foil must be able to support the yacht in even slower conditions such as during take-off or during a tack and the span is limited by the rule, the root chord has to be chosen carefully and should not be too low. The root chord is also relevant to structural aspects which are not yet considered.

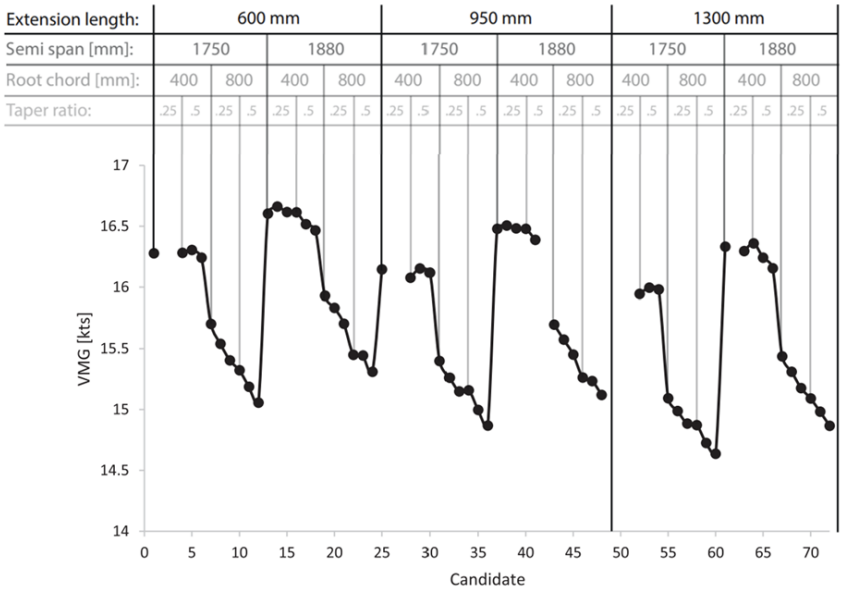



**Figure 3.14: Foil performance in 8 kts upwind at $\beta_t = 57.5°$**

The taper ratio also influences the area of the foil but less significantly than the root chord as it only affects the tip chord. Generally, the smaller taper ratio leads to higher $V_{\mathrm{MG}}$ as the foils do not require the larger area in most conditions. The weight is better stored in the bulb which has a lower wetted surface area to weight ratio. An exception are the candidates with the smaller taper ratio of those mentioned above, that could not support the weight of the yacht anymore.

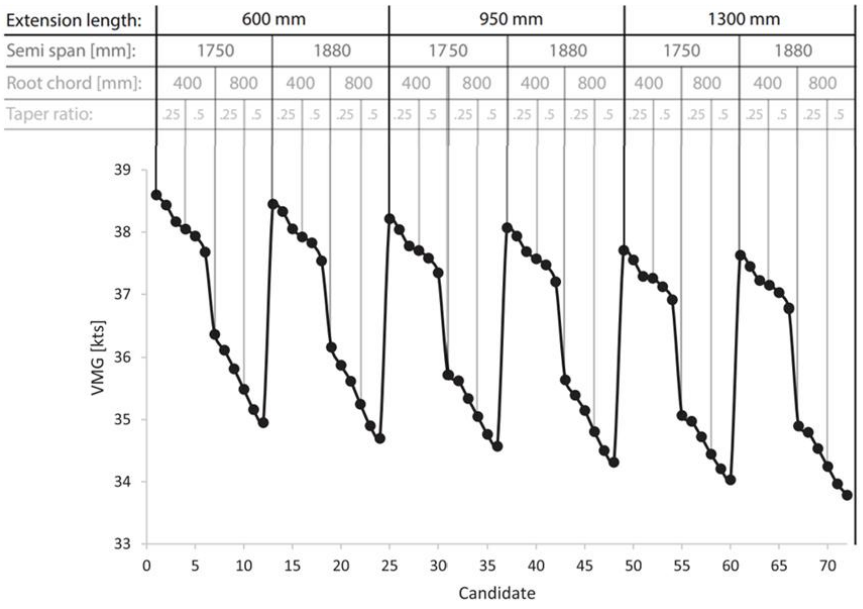



**Figure 3.15: Foil performance in 14 kts downwind at $\beta_t = 140°$**

The wing rake of the hydrofoil determines the angle of attack at every section, where a higher rake results in higher lift forces. To control the amount of lift generated, the crew uses the flap angle which can be trimmed actively. The results show that the foils with the lowest rake (0.0°) perform best in most conditions. The Eppler 874 section has a zero-lift angle of -0.5°, so the foil is generating positive lift at 0.0° rake even if the flap is not actuated. The amount of lift produced without the flap engaged is, however, not sufficient to support the weight of the yacht. Candidate 13 with 0.0° rake for example is operated at a flap angle of 8° in 11 kts $V_{\mathrm{TW}}$ upwind. Candidates 14 and 15 with the higher rakes are operated at 5.6° and 3.1° flap angle, respectively and are slower than Candidate 13. This shows that the foils with lower rake angles and higher flap angles are more efficient. This is true for all conditions assessed except the upwind condition in 8 kts $V_{\mathrm{TW}}$. Here candidate 13 with a rake of 0.0° and a flap angle of 10° is performing less than candidate 14 with 2.0° rake and 7.4° flap angle. This suggests that too high flap angles reduce efficiency again. To increase the efficiency of candidate 13 at low boat speeds, the yacht could be sailed at a lower bow-down pitch, which would reduce the required flap angle.

The study has shown that candidate 13 with the shortest extension, the larger span, the smaller root chord and taper ratio and the smallest rake performs best across the range of conditions. The worst performing design is candidate 72, the T-Foil with high semi span, high root chord, high taper ratio and highest rake. Candidate 13 is on average 13.3% faster than candidate 72 and shows the immense influence of hydrofoil design on performance. However, while candidate 13 is performing best in the assessed conditions, it might have a too low planform area to support the yacht during take-offs and tacks. This would render the foil unsuitable and shows that at least one such condition should be included in the design process. Including such a condition would potentially favor Candidate 19, the best performing foil among the designs with higher root chord. Ultimately, also the weighting of the assessed conditions has a huge influence on the design. A higher weighting of the faster conditions for example could have favored candidate 1 with the lower semi span.

### 6.2 Parametric study results vs real life observations

In contrast to the trends observed in the parametric study, the 36$^{\mathrm{th}}$ America´s Cup has shown that ETNZ with their T-Foil was faster than the challengers with their Y-Foils. While this can partially be caused by other components of the yacht, such as the larger sail plan of ETNZ or the different hull which is not reflected in this study, it is supposedly also driven by the hydrofoil design. The study has assumed that the foil is fully submerged at all times with the outboard tip of the wing just operating below the free surface. This is a valid assumption for the foils of the challengers that have been operated fully submerged. ETNZ, however, employed their foil as a surface piercing foil at higher boat speeds which allowed the reduction of wetted surface area when not needed. The parametric study has shown similar trends, as the foils with lower span performed better in high-speed conditions. In surface-piercing operation, the T-foil is assumed superior as the angle at which it pierces the surface is larger which reduces the risk of ventilation. While the operation as a surface piercing foil can be modelled with the developed VPP model, it would require the introduction of the cant angle as another parameter of the study with relatively many values. The parametric examination of the cant angle is therefore not feasible, but its optimisation will be part of the adjoint hydrofoil optimisation routine currently under development. Moreover, the model cannot account for ventilation which would probably result in the Y-Foils still being favored in the surface piercing mode. In addition, ETNZ has used a single control system to actuate both flaps simultaneously. This meant that they could not change the direction of the force vector anymore but allowed them to remove one control system which reduced the required volume and associated wetted surface area. This is not reflected in this study, which assumed a constant density. The performance results have shown a high sensitivity to wetted surface area and hence both effects are expected to have a significant influence. Finally, the foil benefitted from its low chord which is in line with the findings of this study.

## 7. Conclusions

Hydrofoils are the biggest performance differentiators on sailing yachts like the AC75 and have a significant influence on the performance and the sailing state of the yacht. However, they are mostly designed and optimised in terms of their lift and drag characteristics which do not directly reflect the performance of the yacht on the racecourse. Therefore, a physics model of the AC75 has been developed in the VPP FS-Equilibrium and used to assess the effects of the main hydrofoil design variables on the performance of the yacht in different conditions. The results showed that the candidate with the shortest extension, the longer span, the smaller root chord and taper ratio and the lowest rake performed best across the range of conditions. Furthermore, the results have shown how valuable the VPP driven approach is as it allows the direct and accurate comparison of the candidates in terms of their performance on the racecourse.

While the parametric study is helpful to identify the trends in foil design it is not suitable for detailed optimisation as it would require a prohibitively expensive amount of VPP simulations. Therefore, a VPP driven gradient based optimisation strategy is proposed as shown in Figure 3.16. Gradient based optimisation is deemed significantly more efficient than parametric studies and gradient free optimisation. The straightforward way to compute a gradient is the finite difference method where each input variable is slightly perturbated, one at a time, to approximate its influence on the $V_{\mathrm{MG}}$. This is a valid for optimising smaller numbers of variables. However, to approximate a single gradient of the advanced parametric model shown in Figure 3.9 with 86 design variables, 87 VPP simulations would be required. This is again too expensive. For cases with such a high number of input variables the only feasible method to compute the gradient is the adjoint method. This method allows to compute the gradient at the run time of just a handful of VPP simulations independent of the number of input variables. Therefore, it enables the optimisation of thousands of design variables and hence very detailed design in a short time. This adjoint VPP is currently under development and has achieved promising results in first optimisations.

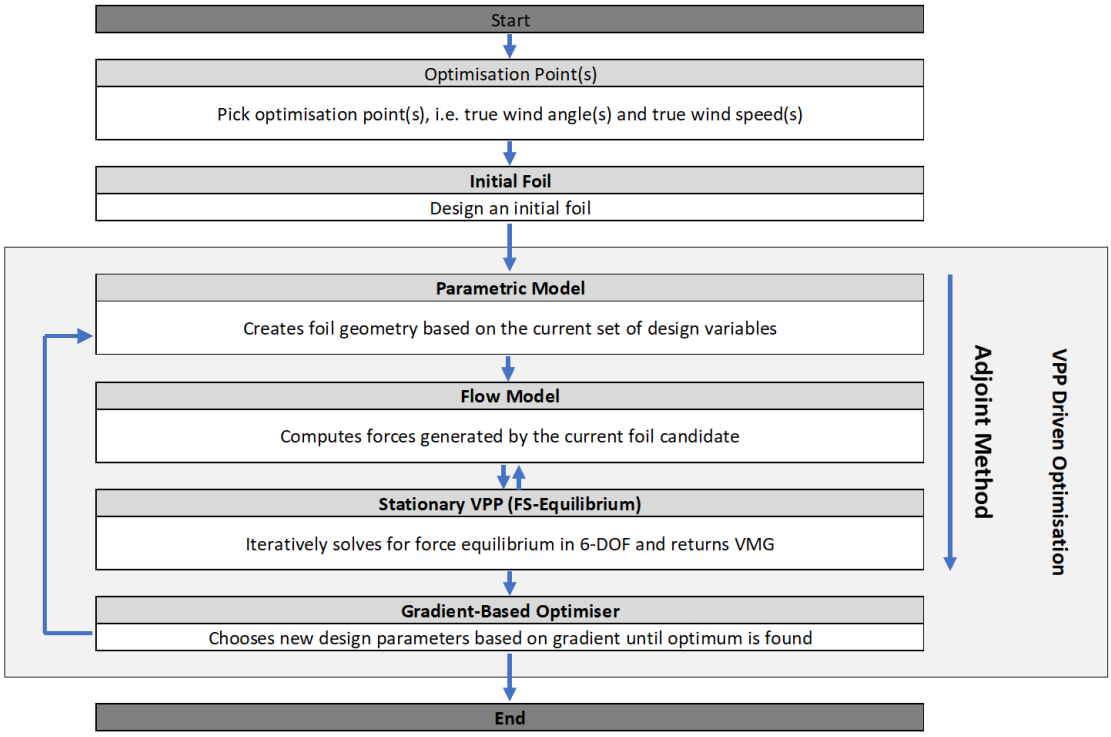



**Figure 3.16: Adjoint VPP driven optimisation strategy**

## 8. References

America´s Cup (2020). Virtual Eye, (viewed 8-Dec-2022), https://ac36.americascup.com/en/advanced-dashboard.

Bingol, O. 2022. NURBS-Python Visualization, (viewed 5-Dec-2022), python.readthedocs.io/en/5.x/visualization.html.

Eggert, F (2018), Flight Dynamics and Stability of a Hydrofoiling International Moth with a Dynamic Velocity Prediction Program (DVPP), Masters of Science in Naval Architecture and Ocean Engineering thesis, Technische Universität Berlin.

Faltinsen, O. (2005). *Hydrodynamics of High-Speed Marine Vehicles*. Cambridge University Press, Cambridge, UK.

Gattini, S. (2020). PRADA ACWS - Day 1 Photo Gallery, (viewed 22-Jun-2021), https://www.lunarossachallenge.com/en/gallery/614_PRADA-ACWS-Day-1-Photogallery.

Hazen, G.S. (1980). A Model of Sail Aerodynamics for Diverse Rig Types, The New England Yacht Symposium.

Hansen H., Hochkirch K., Burns I., and Ferguson S. (2019). Maneuver Simulation and Optimization for AC50 Class, *Journal of Sailing Technology*, 4, 142-160.

Hochkirch, K. (2018). FS-Equilibrium User Manual.

Hoerner, S. F. (1965). *Fluid-Dynamic Drag*. Published by the Author.

Jayaraman, P. 2022. GitHub – TinyNURBS, (viewed 31-May-2022), https://github.com/pradeeppyro/tinynurbs.

Larsson, L., Eliasson, R. E., and Orych, M. (2014). *Principles of Yacht Design*. International Marine/McGraw Hill Education, 4th Edition, Camden, Maine.

Melis, F. M., Hansen H., Fischer, M. and Abdel-Maksoud, M. (2022). Velocity Prediction Program for a Hydrofoiling Lake Racer. *Journal of Sailing Technology*, 7, 255-275.

Molland, A. F. and Turnock, S. R. (2021). *Marine Rudders, Hydrofoils and Control Surfaces.* Butterworth Heinneman, 2nd Edition, Oxford, UK.

Molland, A. F., Turnock, S. R. and Hudson D. A. (2017). *Ship Resistance and Propulsion.* Cambridge University Press, 2nd Edition, Cambridge, United Kingdom.

Patterson, N. and Binns, J. (2022). Development of a Six Degree of Freedom Velocity Prediction Program for the foiling America's Cup Vessels. *Journal of Sailing Technology*, 7, 120-151.

Paulin A., Hansen H., Hochkirch K. and Fischer M. (2015). Performance Assessment and Optimization of a C-Class Catamaran Hydrofoil Configuration, 5th High Performance Yacht Design Conference Auckland, New Zealand.
Phillips, W. F. and Snyder, O. D. (2000). Modern Adaptation of Prandtl's Classic Lifting-Line Theory. *Journal of Aircraft,* 3, 662-670.

Piegl, L. and Tiller, W. (1997). *The NURBS Book*. Springer, 2nd Edition.

Prandtl, L. (1918). *Tragflügeltheorie.* Königliche Gesellschaft der Wissenschaften zu Göttingen, Göttingen, Germany.

Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T. (1988). *Numerical recipies in C*. Cambridge University Press, New York.

Reid, J. T. (2020). A General Approach to Lifting Line Theory, Applied to Wings with Sweep. All Graduate Theses and Dissertations. 7842.

Royal New Zealand Yacht Squadron, Circolo Della Vela Sicilia (2020). AC75 Class Rule v1.16.

Royal New Zealand Yacht Squadron, Royal Yacht Squadron (2023). AC75 Class Rule v2.9.

Specialist Committee of 23rd ITTC (2002). Recommended Procedures and Guidelines - Testing and Extrapolation Methods, Resistance Uncertainty Analysis, Example for Resistance Test. International Towing Tank Conference.

Whicker, L. F. and Fehlner, L. F. (1958). Free-Stream Characteristics of a Family of Low-Aspect Ratio, All Moveable Control Surfaces for Application to Ship Design.

# Chapter 4 Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation

*Tannenberg et al. (2025) "Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation", under review at Optimisation and Engineering.*

# Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation

**Rafael Tannenberg**
University of Southampton, UK

**Karsten Hochkirch**
DNV Ship Performance Center, Germany.

**Andrea Walther**
Humboldt-Universität zu Berlin, Germany.

**Stephen R. Turnock**
University of Southampton, UK.

**Stephen W. Boyd**
University of Southampton, UK.

**Abstract.** Various optimisation problems concern a component of a system, whose influence is so large that it significantly affects the state of the system. In these cases, an isolated optimisation of the component does not account for the changes in system state during the optimisation. This introduces inaccuracies. At the same time, the large influence of the component results in large potential performance gains. This requires detailed optimisation. The logical consequence is to model the whole system at every step of the optimisation and to use a large number of design variables. Both of these aspects can however increase computational time so significantly that the approach becomes infeasible. One such problem are hydrofoils in yacht racing. Hydrofoils are the equivalent to airfoils but operated underwater to lift the hull of a yacht out of the water. The design of the hydrofoils has an immense influence on the performance, the state and the trim (i.e. control) of the “yacht” system. To model this whole system, a stationary physics model of the entire yacht is developed. The model is integrated into a detailed optimisation routine that requires 70 design variables, which makes it prohibitively expensive to solve with derivative free methods. Therefore, a gradient-based optimisation strategy is developed, where the gradient is computed using the adjoint method. The adjoint method allows to compute the gradient independent of the number of input variables at a small cost. The adjoint method is only applied to the bottleneck of the yacht model using the algorithmic differentiation tool ADOL-C. The remainder of the model is differentiated using finite differences. The overall gradients are provided to the optimisation algorithm IPOPT. The optimisation strategy is applied to the AC75 America´s Cup class and used to optimise its hydrofoil for velocity made good ($V_{\mathrm{MG}}$) in an upwind condition. The optimised foil shows significant improvement over the baseline foil and demonstrates the immense capabilities of adjoint system-based optimisation. Due to the vast efficiency of the adjoint method, the framework can be extended to optimise thousands of design variables.

**Keywords:** Hydrofoil; Optimisation; Adjoint Method; Algorithmic Differentiation; System-Based.

**Nomenclature**

$A_i$ Area of segment $i$ [m$^2$]
$C_L$ Lift coefficient
$c_{ro}$ Root chord [m]
$D$ Drag force [N]
$\boldsymbol{dv}$ Hydrofoil design variables [m, °]
$F$ VPP Function [kts]
$\boldsymbol{F_i}$ Force at control point $i$ [N]
$F$ Function
$F_i$ Function $i$
$f_x$ Force in $x$-direction [N]
$f_y$ Force in $y$-direction [N]
$f_z$ Force in $z$-direction [N]
$\boldsymbol{fm}$ force- and moment vector [N, Nm]
$L$ Lift force [N]
$l_i$ Lagrangian multiplier $i$
$\boldsymbol{l_i}$ Length of segment $i$ [m]
$l_{ext}$ Extension lengths [m]
$l_{spa}$ Semi span [m]
$N_{\mathrm{i,p}}(u)$ Nonrational B-spline basis function in *u*-direction
$N_{\mathrm{j,q}}(v)$ Nonrational B-spline basis function *v*-direction
$m_x$ Moment around $x$-axis [Nm]
$m_y$ Moment around $y$-axis [Nm]
$m_z$ Moment around $z$-axis [Nm]
$p, q$ Degrees of NURBS surfaces in *u*- and *v*-direction
$\boldsymbol{p}$ Point a function is evaluated for
$\boldsymbol{P}_{\mathrm{i,j}}$ Control points of NURBS-Surfaces [m]
$r_{tap}$ Taper ratio
$\boldsymbol{r_{i_0 j}}$ Vector from node $i_0$ to control $j$
$\boldsymbol{r_{i_1 j}}$ Vector from node $i_1$ to control $j$
$r_{i_0 j}, r_{i_1 j}$ Magnitudes of $\boldsymbol{r_{i_0 j}}$ and $\boldsymbol{r_{i_1 j}}$
$\boldsymbol{S}$ NURBS surface [m]
$\boldsymbol{u}_\infty$ Unit vector in the direction of the freestream
$u, v$ Directions of NURBS surfaces
$\boldsymbol{V_i}$ Local velocity at control point $i$ [m s$^{-1}$]
$V_{\mathrm{MG}}$ Velocity made good [kts]
$V_S$ Boat speed [kts]
$V_{\mathrm{TW}}$ True wind speed [kts]
$\boldsymbol{V}_\infty$ Free-stream velocity [m s$^{-1}$]
$\boldsymbol{v_{ji}}$ Influence of horseshoe vortex $j$ on control point $i$ [m$^{-1}$]
$w_{\mathrm{i,j}}$ Weights of control points of NURBS surfaces
$x_i$ Input variable $i$
$\beta_t$ True wind angle [°]
$\Gamma_j$ Strength of horseshoe vortex $j$ [m$^2$ s$^{-1}$]
$\delta_e$ Elevator rake [°]
$\delta_{\mathrm{ra}}$ Rake angle [°]
$\delta_r$ Rudder angle [°]
$\delta_{win}$ Wing rake angle [°]
$\lambda$ Leeway angle [°]
$\boldsymbol{\xi}$ Free variable vector
$\rho$ Density [kg m$^{-3}$]
$\tau$ Flat parameter

| | |
|---|---|
| as | Automatically spaced |
| AVL | Athena Vortex Lattice |
| BFGS | Broyden–Fletcher–Goldfarb–Shanno |
| CFD | Computational Fluid Dynamics |
| CoG | Centre of Gravity |
| DOF | Degree of freedom |
| ETNZ | Emirates Team New Zealand |
| FEA | Finite Element Analysis |
| IMS | International Measurement System |
| IPOPT | Interior Point Optimizer |
| LLT | Lifting line theory/method |
| LRPP | Luna Rossa Prada Pirelli |
| NURBS | Non-uniform rational B-Splines |
| RANS | Reynolds-averaged Navier Stokes |
| VPP | Velocity prediction program |

## 1. Introduction

Various optimisation problems concern a component of a system, whose influence is so large that it significantly affects the state of the system. In these cases, an isolated optimisation of the component does not account for the changes in system state during the optimisation. This introduces inaccuracies and uncertainty but only requires a single evaluation of the component model per iteration of the optimisation. Modelling the whole system mitigates the inaccuracies but requires significantly more computational resources as the whole state of the system must be computed at every iteration during the optimisation process. Solving for the state requires numerous component model evaluations itself plus the modelling of the other components in the system. Simultaneously, the large influence of the component results in large potential performance gains, which calls for detailed optimisation. To fully exploit the design space a high number of design variables is required. However, the run time of an optimisation often scales with the number of variables, which makes detailed design prohibitively expensive. This is especially the case when the whole system is modelled.

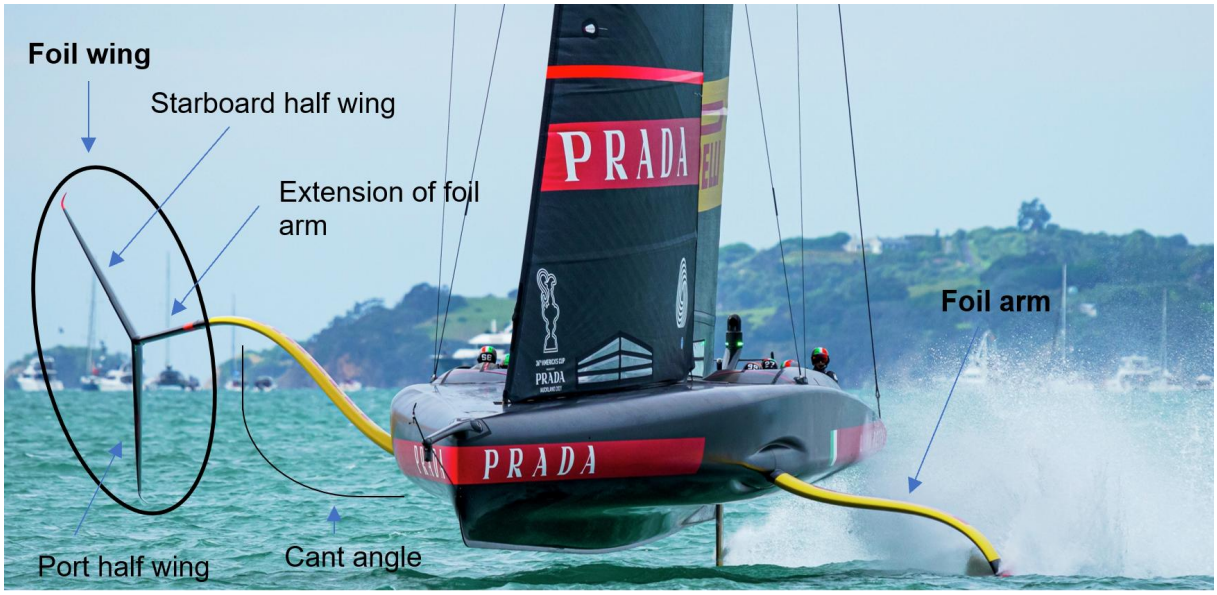



**Figure 4.1: AC75 Luna Rossa based on Gattini (2020)**

One such component are the hydrofoils on racing yachts. Hydrofoils are the equivalent to airfoils but operated underwater to lift the hull of a yacht out of the water. Their effective lift/drag-ratio allows the yacht to reach speeds much higher than with the hull in the water. The design of the hydrofoils thereby has an immense influence on the performance, the state and the trim (i.e. control) of the system, i.e. “the yacht”. This was shown by Paulin et al. (2015) at the example of a C-Class catamaran. Two promising hydrofoil designs were compared using a stationary physics model of the yacht. The model predicted differences in boat speed of up to 20%. Other state variables, such as the leeway angle, changed up to 200%. The control-variable rake changed up to 220%. Conventional hydrofoil optimisations are performed in isolation of the yacht system neglecting these immense changes in yacht state. They are usually based on the state of an initial hydrofoil design. Hence, new candidates are evaluated for a state that they will never experience in reality.

In addition, the high cost of the hydrodynamic model of the foil and/or the optimisation strategy limits the number of design variables. Examples are the optimisations performed by Meneghello et al. (2016) and Ploe (2018). Meneghello used a lower fidelity vortex lattice method, while Ploe modelled the hydrofoil forces with high fidelity computational fluid dynamics (CFD). Despite both contributions employing surrogate models, their approaches were limited to seven and two design variables, respectively. A detailed optimisation of the spine- and planform of the foil, as well as its twist distribution, requires a much higher number of design parameters (70 – several hundreds if section design is to be included). The only feasible option for more than 70 design variables is a gradient-based algorithm combined with the adjoint method for gradient computation. The adjoint method computes the gradient independent of the number of input variables at a small cost but requires extensive implementation efforts. Adjoint optimisations have been performed in many fields including Biology (Hovland et al., 1997), Medicine (Jee et al., 2005), Physics (Kim et al., 2006) and Engineering. Examples in engineering range from electrical engineering (Hart et al., 2006), over controller design (Röbenack, 2007), to structural optimisation (Tadjouddine et al., 2006). In the context of hydrofoil optimisation, Garg et al. (2017) conducted an adjoint hydrostructural optimisation using high fidelity CFD and finite element analysis (FEA). The use of the adjoint method resulted in the ability to optimise 200 design variables. This optimisation was, however, also carried out in isolation of the yacht system.

This work presents the development of an adjoint hydrofoil optimisation strategy where the whole yacht system is considered. The system is modelled with a six-degree-of-freedom stationary physics model of the yacht developed in the velocity prediction program (VPP) FS-Equilibrium. The adjoint version of the VPP model is derived using the algorithmic differentiation tool ADOL-C (Walther and Griewank 2020). The adjoint method is however only applied to the bottleneck of the VPP, which corresponds to the parametric model of the hydrofoil and the lifting line method computing the hydrofoil forces. This represents the first part of the gradient. The remainder of the program (solver and other force modules) is differentiated using the simpler finite difference method (second part of the gradient). This saves implementation time and preserves modularity. The two gradient parts are combined using the chain rule and are provided to the optimisation algorithm IPOPT (Interior Point Optimizer) (Wächter and Biegler 2006) at every step of the optimisation.

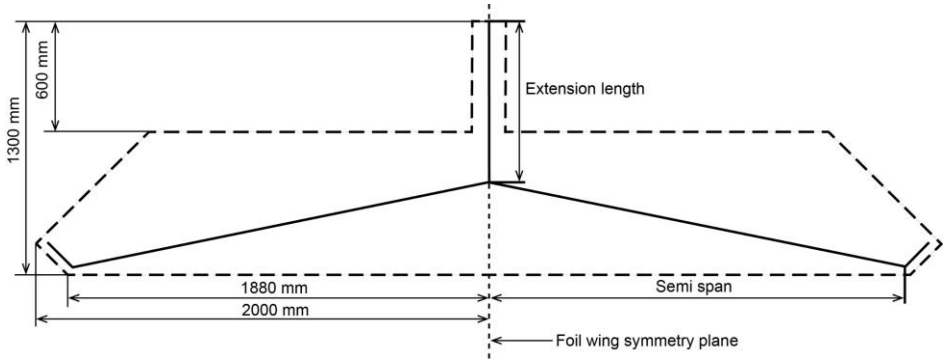



**Figure 4.2: Foil Box**

The methodology is applied to an AC75 type yacht. The AC75 is the foiling monohull class currently sailed in the America´s Cup, the Formula 1 of sailing. Figure 4.1 shows an AC75 operating with the windward foil out of the water to provide righting moment and the leeward foil in the water to produce hydrodynamic lift and side force. The main foils consist of two parts which are the foil arm and the foil wing. The design of the foil wing is open to development inside the `Foil box´ (Figure 4.2) and must be symmetric around the `Foil wing symmetry plane´. Furthermore, the foil wing has to weigh exactly 921 kg. The wings are built from a combination of steel and lead to accommodate this weight while having a low volume and wetted surface area. Some of the weight can be stored in a bulb and one or two flaps are used for force control. In the last America's Cup, Emirates Team New Zealand (ETNZ) and American Magic used bulbs and wings with small chord while Luna Rossa Prada Pirelli (LRPP) and INEOS Team UK opted wings with no bulb and larger chord. ETNZ´s foil had no anhedral angle (T-Foil) and a single flap, the other teams chose Y-Foils with two flaps.

The developed framework is first used to optimise five design variables describing the general design of the AC75 hydrofoil. The results are compared to the findings of a parametric study recently published by Tannenberg et al. (2023) which acts as a first test of the routine. The parametric study used the same yacht model and variables. As a second test, the twist-distribution of a straight T-Foil with no taper ratio is optimised. This case is analysed to determine if the optimiser can tailor the twist-distribution such that the desired elliptical lift-distribution is achieved. With the two tests completed, a detailed optimisation of the entire foil is performed. The detailed optimisation has 68 design variables describing the spine- and the planform of the foil as well as the twist distribution in detail.

## 2. Velocity prediction model

The physics model of the AC75 is developed in the VPP FS-Equilibrium and is briefly introduced in this section, for a more thorough description see Tannenberg et al. (2023). For more information on FS-Equilibrium see Hochkirch (2018). FS-Equilibrium is a workbench for sailing yacht performance prediction and can compute stationary and dynamic sailing states. "Force modules" are used to model the different components of the yacht and return the forces and moments they produce. Once all forces and moments equate to zero in the six degrees of freedom (DOF), a steady sailing state is reached. This steady sailing state is determined using a Newton-Raphson method. The method modifies state and trim variables until force equilibrium is found. FS-Equilibrium has been applied to hydrofoiling yachts before such as the C-Class Catamaran Groupama (Paulin et al., 2015), the International Moth (Eggert, 2018), the AC50 (Hansen et al., 2019) and the QFX Lake Racer (Melis et al., 2022).

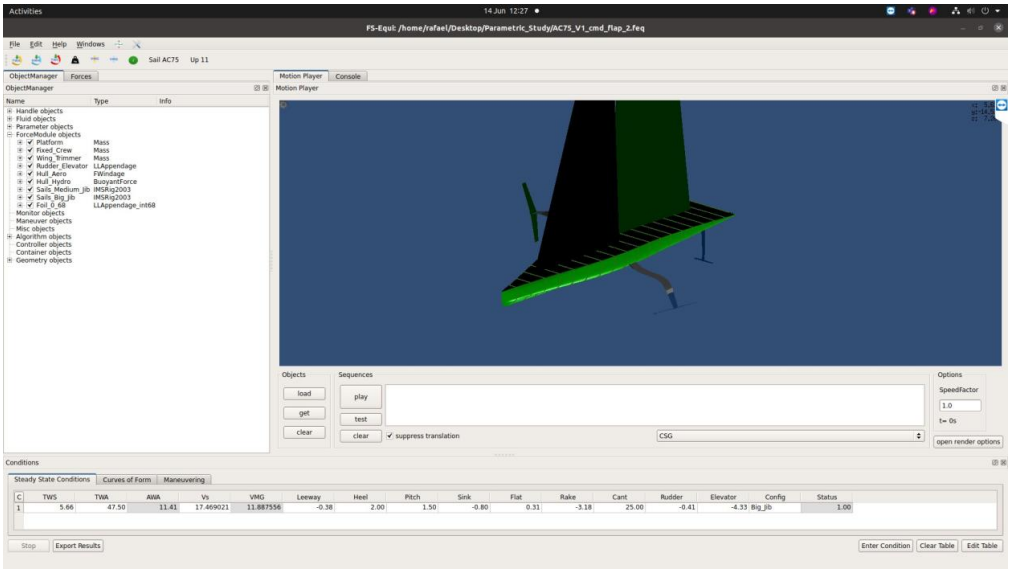



**Figure 4.3: FS-Equilibrium´s graphical user interface showing a generic AC75**

FS-Equilibrium is used to compute the steady sailing states of the AC75, by balancing the six forces and moments

$$\boldsymbol{fm} = \left(f_x, f_y, f_z, m_x, m_y, m_z\right)^T. \tag{1}$$

Table 4.1 shows the state and trim variables used to balance the forces of the AC75 model. These variables do not exclusively influence one degree of freedom but are the variables with the highest influence on the respective degree of freedom. For simplification reasons the foil wings are raked rather than controlled with flaps as on the real yachts. The remaining state variables, heel angle, pitch angle and ride height (sink) are fixed for increased robustness.

**Table 4.1: Solver set-up of the AC75 physics model**

| DOF | Balancing variable with the highest influence on the DOF |
|---|---|
| $f_x$ | boat speed $V_S$ (state) |
| $f_y$ | leeway angle λ (state) |
| $f_z$ | rake angle $\delta_{\mathrm{ra}}$ (trim) |
| $m_x$ | sail-power (flat) τ (trim) |
| $m_y$ | elevator rake $\delta_e$ (trim) |
| $m_z$ | rudder angle $\delta_r$ (trim) |

The problem is expressed with Lagrange multipliers ($l_i \neq 0$) in the function

$$F = -V_S + l_0 \sum f_x + l_1 \sum f_y + l_2 \sum f_z + l_3 \sum m_x + l_4 \sum m_y + l_5 \sum m_z\,, \tag{2}$$

where the first order condition for the optimum is

$$F \overset{!}{=} \min \rightarrow \nabla F \overset{!}{=} 0. \tag{3}$$

This leads to a non-linear system of equations:

$$\begin{aligned}
&F_0 = \frac{\partial F}{\partial V_S} \overset{!}{=} 0, \quad F_1 = \frac{\partial F}{\partial \lambda} \overset{!}{=} 0, \quad F_2 = \frac{\partial F}{\partial \delta_{\mathrm{ra}}} \overset{!}{=} 0,\\
&F_3 = \frac{\partial F}{\partial \tau} \overset{!}{=} 0, \quad F_4 = \frac{\partial F}{\partial \delta_e} \overset{!}{=} 0, \quad F_5 = \frac{\partial F}{\partial \delta_r} \overset{!}{=} 0,\\
&F_6 = \frac{\partial F}{\partial l_0} \overset{!}{=} 0, \quad F_7 = \frac{\partial F}{\partial l_1} \overset{!}{=} 0, \quad F_8 = \frac{\partial F}{\partial l_2} \overset{!}{=} 0,\\
&F_9 = \frac{\partial F}{\partial l_3} \overset{!}{=} 0, \quad F_{10} = \frac{\partial F}{\partial l_4} \overset{!}{=} 0, \quad F_{11} = \frac{\partial F}{\partial l_5} \overset{!}{=} 0.
\end{aligned} \tag{4}$$

This non-linear system of equations is solved using the Newton-Raphson method. This solver set-up allows the direct inclusion of a further optimisation variable. This feature is not yet used but can be relevant for including the optimisation of an additional state variables in the future. For the current problem, the Newton-Raphson could also be directly employed for force balancing. This is more computationally efficient. A similar approach was used by Patterson and Binns (2022).

The force modules of the AC75 are presented in Table 4.2. For details see Tannenberg et al. (2023). The forces and moments generated by the hydrofoils are modelled using an advanced lifting method (LLT) as presented in the next section.

**Table 4.2: AC75 force modules**

| Component | Model | Comments |
|---|---|---|
| Platform gravity | Gravity model | Weight and CoG from AC75 rule |
| Crew gravity | Gravity model | Weight from rule, CoG estimated |
| Hull aero | Coefficient-based | Coefficients from Hoerner (1965) |
| Sails aero | Tuned IMS2003 | Factor of 1.3 on driving force |
| Rudder hydro | Lifting Line | Based on Phillips and Snyder (2000) |
| Main foil hydro | Lifting Line (adjoint) | Same as for Rudder, but adjoint |

### 3. Hydrodynamic foil model

The forces and moments produced by the main hydrofoils of the AC75 are computed with a lifting line method. The method was originally derived by Prandtl (1918) and is a non-viscous method for 3D foil force prediction. Viscosity is of importance in stalled cases or when the flow separates. For the desired operation of a hydrofoil, this is not the case, and the lifting line method has an accuracy similar to viscous methods such as Reynolds-averaged Navier Stokes (RANS). RANS methods require hours of computational time on multiple cores for a single foil force evaluation and are not suited for the use within yacht models that involve numerous such evaluations to compute the sailing state. Lifting line methods on the other hand solve in milliseconds on a single core and are the most common method for foil force prediction in stationary and dynamic yacht simulators (e.g. SumToZero, 2021). The lifting line method uses vortices in the free stream to represent how a foil section affects its surrounding flow. Multiple vortices are oriented along a line that follows the span of the foil. For a comprehensive introduction to lifting line methods see Katz and Plotkin (1991). A variation of Prandtl´s method is used for this study. This advanced method was developed by Phillips and Snyder (2000) and employs horseshoe vortices at every station along the span and a vectorised form of the Kutta-Jukowski-Law. Each horseshoe vortex is formed of a bound vortex and two semi-infinite vortices. The vectorised lifting law includes the influence of the bound vortices on each other. This is required to model the effects of leeway, sweep and anhedral. It also includes the influence of multiple surfaces on each other. The method solves for the strength of each horseshoe vortex $\Gamma_j$ by relating two definitions of the force generated at every segment. The force produced by each bound vortex is computed with

$$\mathrm{d}\boldsymbol{F_i} = \rho \Gamma_i \boldsymbol{V_i} \times \mathrm{d}\boldsymbol{l_i}\,, \tag{5}$$

where the local velocity at every control point is calculated from

$$\boldsymbol{V_i} = \boldsymbol{V}_\infty + \sum_{j=1}^{N} \Gamma_j \, \boldsymbol{v_{ji}}\,. \tag{6}$$

The velocity $\boldsymbol{v_{ji}}$ is the induced velocity of horseshoe vortex $j$ at control point $i$ normalised by the vortex strength and is computed with

$$\boldsymbol{v_{ji}} = \frac{1}{4\pi}\left[\frac{\boldsymbol{u}_\infty \times \boldsymbol{r_{i_1 j}}}{r_{i_1 j}(r_{i_1 j} - \boldsymbol{u}_\infty * \boldsymbol{r_{i_1 j}})} + \frac{(r_{i_0 j} + r_{i_1 j})(\boldsymbol{r_{i_0 j}} \times \boldsymbol{r_{i_1 j}})}{r_{i_0 j} r_{i_1 j}(r_{i_0 j} r_{i_1 j} + \boldsymbol{r_{i_0 j}} * \boldsymbol{r_{i_1 j}})} - \frac{\boldsymbol{u}_\infty \times \boldsymbol{r_{i_0 j}}}{r_{i_0 j}(r_{i_0 j} - \boldsymbol{u}_\infty * \boldsymbol{r_{i_0 j}})}\right] \tag{7}$$

where the first term in the brackets is the outbound semi-infinite vortex influence, the second term the bound vortex influence and the third term the inbound semi-infinite vortex influence of segment $i$ on colocation point $j$. The vectors $\boldsymbol{r_{i_0 j}}$ and $\boldsymbol{r_{i_1 j}}$ point from the semi-infinite vortices to the colocation point and $r_{i_0 j}$ and $r_{i_1 j}$ are the lengths of these vectors. The bound vortex term is excluded when $i = j$, as the bound vortex is not causing downwash on its own segment. Relating the lift based on the sectional lift coefficient to the vectorised vortex lifting law results in a non-linear equation system. This is solved iteratively but is potentially unstable. Hence, the method was reduced to a linear problem through a linear lift-curve slope and yields the following system of equations

$$\rho \Gamma_i \left| \left( \boldsymbol{V}_\infty + \sum_{j=1}^{N} \Gamma_j \, \boldsymbol{v_{ji}} \right) \times \mathrm{d}\boldsymbol{l_i} \right| - \frac{1}{2} \rho \, {\boldsymbol{V}_\infty}^2 \, C_L(\boldsymbol{V_i}) \mathrm{d}A_i = 0. \tag{8}$$

For a more general introduction to hydrofoils and associated models see Molland and Turnock (2021). The two half wings of the foil are discretised with 30 stations each, while the extension uses 10, all with equal spacing. This is in contrast to the 40 stations per part with cosine-spacing proposed by Phillips and Snyder (2000) but delivers much more robust results in the optimisation. The cosine-distribution has led to unrealistic optimal geometries especially in the region of the junction. This was amplified by the small spacing at the junction caused by the high number of stations. Parts of the foil that are above the waterline are not considered (Figure 4.4). The loss in lift caused by the presence of the free surface is modelled with a second transformed foil. The biplane analogy (Faltinsen, 2005) is used, which is best suited for the design space and conditions explored.

The force module of the main hydrofoil additionally computes the forces and moments from the weight of the foil based on an integration of the sectional areas along the span using Simpson´s rule. An average density of 8305 kg m$^{-3}$ of the involved materials (75% high density steel, 20% lead and 5% hydraulic oil to represent the control systems) is used. A bulb is automatically sized to meet the weight requirement of 921 kg. The centre of gravity of the foil is determined by taking moments of the sectional areas around the junction of the foil arm and the foil arm extension. The resistance of the potential bulb is computed using a form factor approach. The lifting line method combined with the form factor captures the general trends well but loses accuracy in the region of the bulb. Implementing a slender body model would enhance the accuracy in the area but should also be treated with caution and results should be validated with viscous CFD simulations. Furthermore, wave-making drag, and spray drag are not/not yet considered. The model is available in normal mode for finding the steady sailing state and in adjoint mode for gradient computation as explained later.

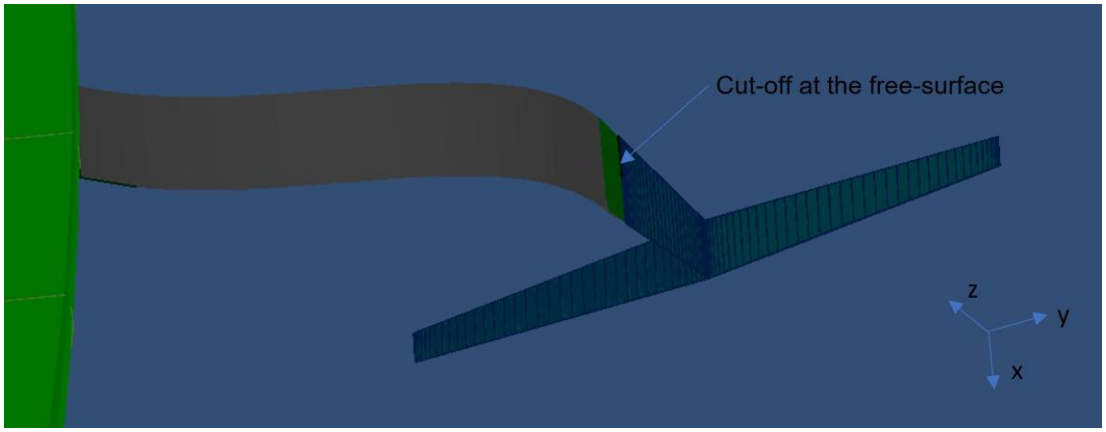



**Figure 4.4: AC75 foil discretised with 30 stations per half wing and 10 stations at the extension using an equal distribution**

## 4. Parametric model

The parametric model serves to generate the shape of the hydrofoil from the set of design variables. The lifting line method reads the geometrical information of the hydrofoil from ruled surfaces which entail for example the position of each segment, its chord length and twist. The ruled surfaces are provided from the parametric model in the form of NURBS (Non-rational uniform B-spline) surfaces. NURBS surfaces are defined as

$$\boldsymbol{S}(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)w_{\mathrm{i,j}}\boldsymbol{P}_{\mathrm{i,j}}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{\mathrm{i,p}}(u)N_{\mathrm{j,q}}(v)w_{\mathrm{i,j}}} \quad \text{for} \quad 0 \leq u,v \leq 1. \tag{9}$$

where $N_{\mathrm{i,p}}(u)$, $N_{\mathrm{j,q}}(v)$ are the nonrational B-spline basis functions, $\boldsymbol{P}_{\mathrm{i,j}}$ are the control points and $w_{\mathrm{i,j}}$ are the weights of the control points. The directions $u$ and $v$ are alongside the surface ($u$ spanwise, $v$ chordwise) and $p$ and $q$ are the degrees in the $u$- and $v$-direction, respectively. See Piegl and Tiller (1997) for more detail. The cross-sections of the foil at every station (i.e. the "thickness" of the foil) are supplied through their sectional properties (i.e. lift curve slopes, the zero lift angles and the profile drag coefficients). This is discussed in detail in Tannenberg et al. (2023).

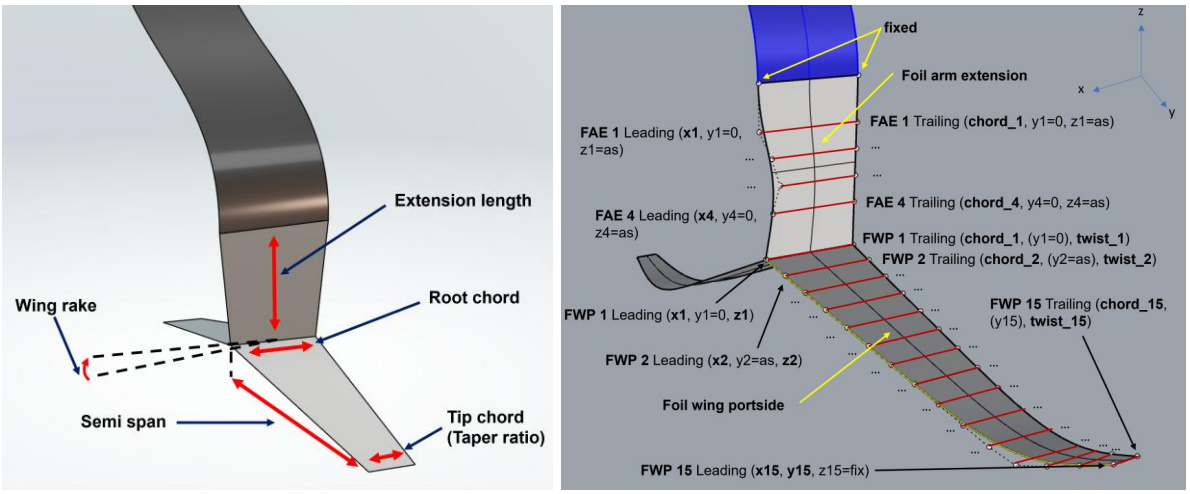



**Figure 4.5: Simple parametric model with 5 variables (a), detailed model with 68 variables (b); design variables bold; as = automatically spaced**

Figure 4.5a shows the simple parametric model which was used in the parametric study. It describes the hydrofoil with the parameters extension length $l_{ext}$, semi span $l_{spa}$, root chord $c_{ro}$, taper ratio $r_{tap}$ and wing rake $\delta_{win}$. This simple model is used in a first optimisation to compare the optimisation results with the findings of the parametric study.

The twist-distribution optimisation and the detailed optimisation of the hydrofoil are performed using the advanced parametric model shown in Figure 4.5b. This model is based on 68 design variables and allows thorough optimisation of the spine and plan form of the foil as well as the twist distribution. The extension of the foil arm has four stations which are defined by the x-coordinates of every leading-edge point and the chords. The y-coordinates and the twists are set to zero due to the symmetry requirement. The stations are equally spaced in z-direction between the end of the foil arm and the root of the foil wing. This results in 8 variables for the extension. The portside half wing has 15 stations. Every station is defined by the x-and z-values of the leading edge, a chord and a twist variable. Station 15 additionally has a y-variable that defines the span of the wing. Its z-value is however fixed at the bottom of the foil box. The y-value of station 1 is zero due to the symmetry constraint. The intermediate stations are spaced equally in between. This results in 60 variables for the portside half wing, which is mirrored around the foil wing symmetry plane to generate the starboard half wing. This model is sophisticated enough to re-create the shapes of the hydrofoils seen in the 36$^{th}$ America's

Cup as shown in Figure 4.6 representing the ETNZ (a) and LRPP (b) hydrofoil, respectively. The models are based on the C++ library TinyNURBS (Jayaraman, 2022) and use the python library NURBS-python (Bingol, 2022) for visualisations. The parametric models are internally coupled to the lifting line method to allow seamless gradient computation as discussed in the next section.

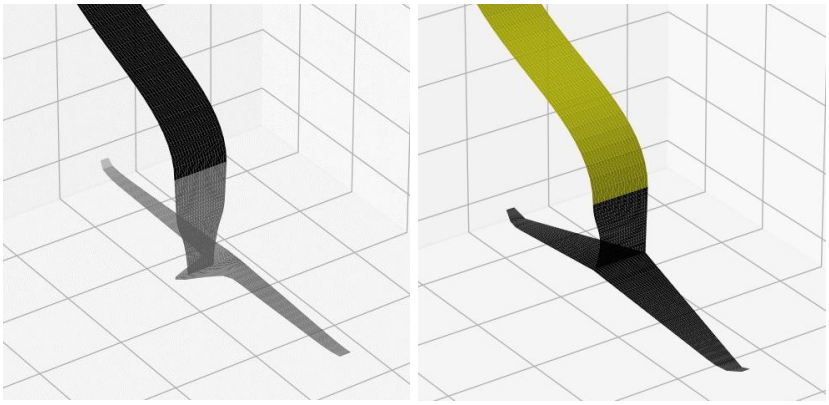


**Figure 4.6: Remodelled hydrofoils of ETNZ (a) and LRPP (b)**

## 5. Model validation

The correct functioning and accuracy of the hydrodynamic model of the foil and the entire physics model of the yacht are crucial for reliable optimisation results. The hydrodynamic model has been validated in terms of lift/drag-ratio. The whole yacht model was validated with real life race data from the last America´s Cup (America´s Cup, 2020). Both showed good agreement as discussed in Tannenberg et al (2023). Herein, a more thorough validation is performed for the lift-distribution computed by the hydrofoil model. The predicted lift-distributions are benchmarked against the distributions computed by Athena Vortex Lattice (AVL) (Drela and Youngren, 2022) and those found in the literature for two standard foil shapes. It is also analysed if the computation of the weight and the centre of gravity of the foils is executed correctly.

The lift-distribution along the span of the foil is an important characteristic in hydrofoil design with respect to induced drag. Figure 4.7 shows the lift-distributions for a half-wing with no taper and one with a taper ratio of 0.25, both with constant twist. The lift-distributions are predicted by a lifting line method and a computational fluid dynamics code (Hospodář et al., 2022). The straight wing shows a lift-distribution with a more rectangular shape than the desired elliptical shape due to the constant chord. The taper of the second foil reduces the lift generated towards the tip which results in a more triangular shape.

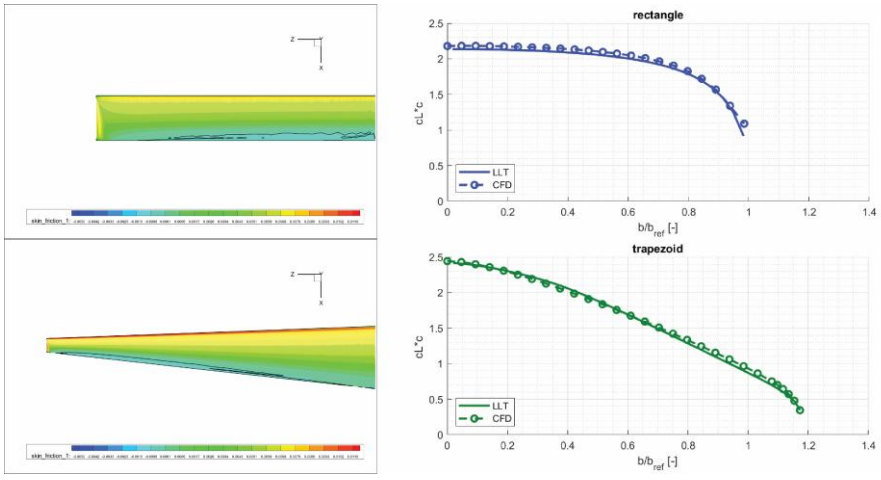



**Figure 4.7: Lift-distributions of a non-tapered and a tapered wing (Hospodář et al. 2022)**

Two similar foils are analysed with the lifting line method used in the optimisation framework and AVL for further comparison. The results are shown in Figure 4.8. Both, the lifting line method and AVL predict the rectangular and the triangular lift-distributions observed in Figure 4.7. Slight differences between the two can be seen towards the tip in the rectangular distribution and in the middle between the root and the tip of the tapered foil. The lift-distribution of the tapered foil predicted by the lifting line method also shows a slight drop in lift at the root. This is not predicted by the other models. In general, however, the methods show good agreement, and the lifting line method is hence deemed suitable for lift-distribution prediction.

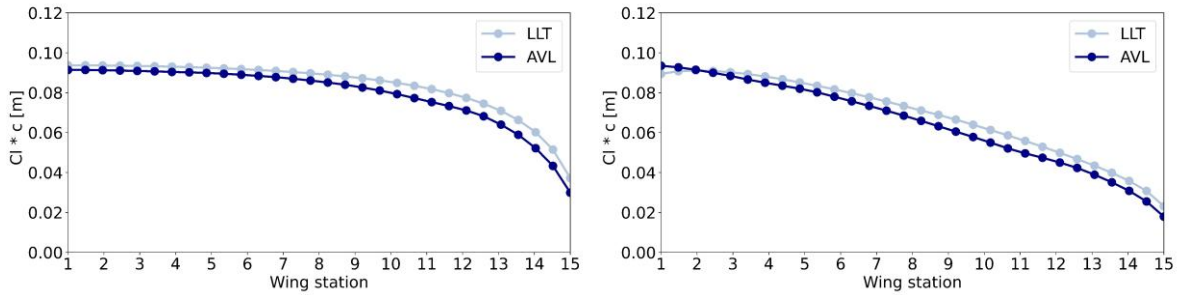



**Figure 4.8: Lift-distribution predicted by the lifting line method and AVL for the constant chord wing (a) and the tapered wing (b)**

The hydrofoil force module also computes the weight and the centre of gravity of the foil. The sectional area of the foil is integrated along the span which results in a 2.5D approach rather than a full 3D approach. Moments of the sectional areas are used to compute the centre of gravity. This is perfectly acceptable if the spine of the foil is straight but can lose accuracy if the spine is curved. Therefore, foils with different amounts of curvature have been tested and compared to a full 3D approach implemented in a commercial computer aided design software. The weight error was 3.2% for the foil with the highest curvature and 0% for a straight foil. The centre of gravity error was 4% for the highest curvature foil and 0% for the straight foil. This is deemed sufficient.

## 6. Optimisation strategy

This section presents the logic of the VPP driven hydrofoil optimisation routine while the details of the gradient computation are discussed in the next chapter. Figure 4.9 shows the flow of the proposed routine. In a first step, the optimisation point is defined. This is the true wind angle ($\beta_t$) and true wind speed ($V_{\mathrm{TW}}$) the yacht sails in and hence the condition for which the foil is optimised. At a later stage, multiple of those points will be considered to optimise for a broader range of conditions. At this stage however, only one condition can be assessed in one optimisation run. In the second step, the design parameters of an initial foil are defined. The parameters are passed to the parametric model which generates the initial hydrofoil geometry. The VPP is now run solving for the equilibrium sailing state of the AC75 in the set condition with the initial hydrofoil. This is an iterative process where the hydrofoil forces are calculated numerous times by the advanced lifting line method for changing states and trims (VPP loop). The lifting line method is directly coupled to the parametric model from which it receives the geometric information. Once a valid equilibrium sailing state is found, the VPP returns the achieved $V_{\mathrm{MG}}$ and passes it to the optimisation algorithm IPOPT. The optimisation is currently performed for $V_{\mathrm{MG}}$ but with a fixed $\beta_t$. This means optimising for $V_{\mathrm{MG}}$ is the same as optimising for $V_{\mathrm{S}}$, but $\beta_t$ will be included as an optimisation variable in the future. Additionally, the cant angle will be included as an optimisation variable. It is currently fixed to the position that generates the highest righting moment, while preventing the tip from piercing the surface.

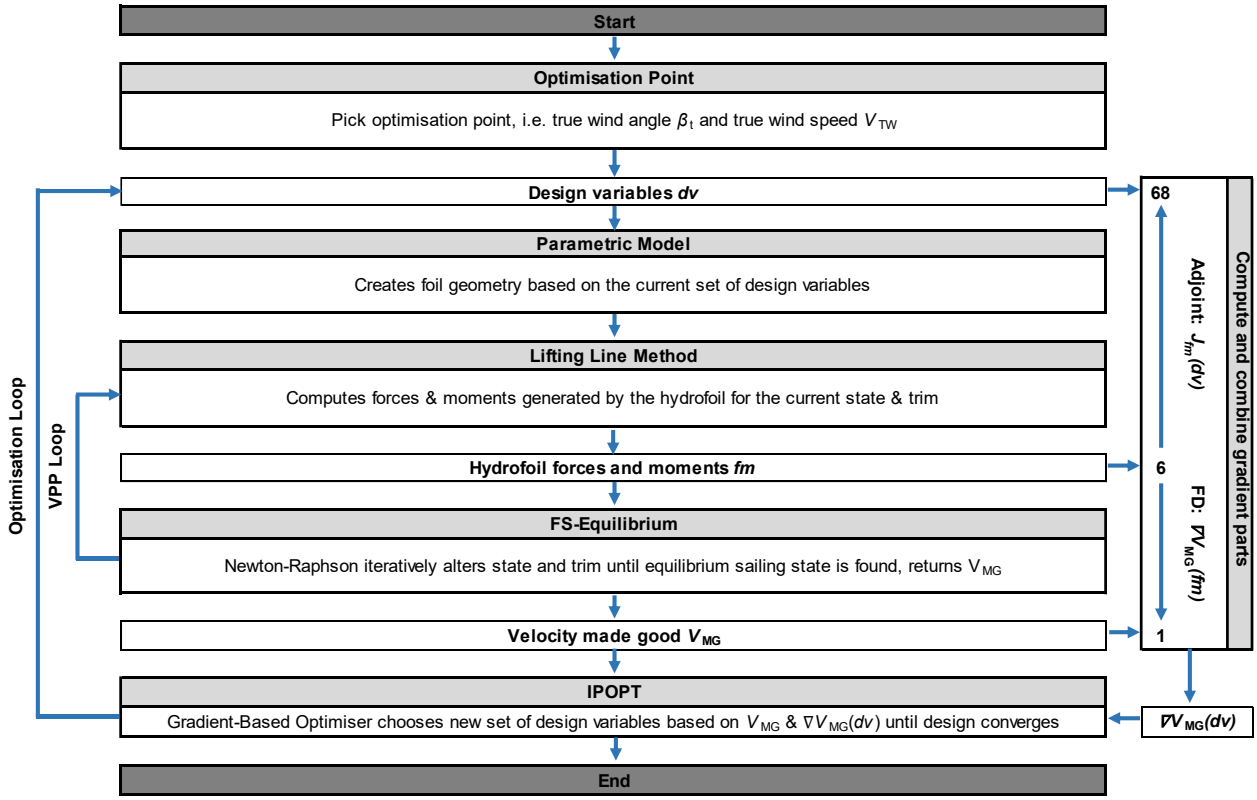



**Figure 4.9: Adjoint VPP-driven optimisation routine**

IPOPT also requires the gradient of the function (VPP including lifting line and parametric model) which is computed next and passed to the optimiser. This three-step process is described in detail in the next section. Based on the $V_{\mathrm{MG}}$ and the gradient, IPOPT computes a set of design variables which results in a higher $V_{\mathrm{MG}}$. This new set is fed back to the parametric model which updates the geometry and the VPP is run again. The new $V_{\mathrm{MG}}$ and gradient are fed to IPOPT again which comes up with an even better performing set of design variables (Optimisation loop). This process is repeated until the design has converged to the optimum.

The most expensive part of this routine is the computation of the gradient. The runtime required for it increases with every design variable. This makes the VPP driven optimisation of a high number of variables infeasible when the gradient is approximated with finite differences. The next section presents an innovative approach that allows to compute the gradient at a small run time independent of the number of input variables. As a result, thousands of variables can be optimised efficiently. Interior point methods also require second order derivatives. These are approximated by IPOPT using a BFGS (Broyden–Fletcher–Goldfarb–Shanno) method.

## 7. Gradient computation

The gradient is the direction and rate of fastest increase of a scalar-valued function $f$ with respect to its input variables $x_i$ at the point $\boldsymbol{p} = (x_1, \dots, x_n)$. If a function has more than one output the individual gradients are presented in the Jacobian matrix, where the rows are the transposed gradients (see Equation 13). Gradients can be computed in several ways, the easiest of which is finite difference, where an approximation of the gradient is obtained by evaluating a function once for an initial set of input variables, then perturbating a single variable and calculating the function again. This is repeated for every design variable. From the difference in the outputs and the step sizes the gradient is approximated. Hence, for a function with 68 input variables (hydrofoil design variables $\boldsymbol{dv}$) and 1 output ($V_{\mathrm{MG}}$), 69 function evaluations (VPP runs) are required. This is prohibitively expensive. Another possibility is to differentiate the function using the chain rule starting from the input variables differentiating to

the output variable (also referred to as forward differentiation). This can either be done by hand or by automatic/algorithmic differentiation tools that can automatically differentiate computer code. Similar to finite difference, one partial derivative has to be evaluated per input variable. Hence, the time required to compute a gradient is also similar to finite difference and depends on the number of input variables. Both approaches are, however, independent of the number of output variables, so these methods are better used for functions with a small number of inputs and a large number of outputs. For typical optimisation problems the opposite is the case most often. In such cases the hand differentiation/algorithmic differentiation can also be applied starting from the outputs differentiating to the inputs. This is called reverse differentiation or the adjoint method. It means the gradient of a function can now be computed independent of the number of input variables and its complexity only depends on the number of output variables. This is extremely powerful for optimisation problems with large numbers of input variables. More details on algorithmic differentiation including the stated complexity results can be found in Griewank and Walther (2008).

Differentiating a computer code however is a complex and time-consuming process even if algorithmic differentiation tools are used. Finite differences in contrast are simple to implement. To reduce development time, while making the run time independent of the number of input variables a hybrid approach was developed. The adjoint method is only applied to the bottleneck of the routine, while finite differences are used where it does not significantly affect run time. Consequently, the gradient parts are joined together. The resulting gradient computation runs at a small multiple of a single normal evaluation of the VPP model.

The bottleneck of the VPP driven routine is the parametric model and the lifting line method. This includes the routine which computes the bulb size and forces. Here the Jacobian has to be computed for the six forces and moments $\boldsymbol{fm}$ the hydrofoil produces with respect to the large number of design variables $\boldsymbol{dv}$. In case of the advanced parametric model, this results in 68 input variables and 6 output variables. Applying the adjoint method here means the Jacobian can be computed by evaluating six partial derivatives only. This holds true for any number of input variables. For the simple parametric model with five input variables $\boldsymbol{dv} = (l_{ext}, l_{spa}, c_{ro}, r_{tap}, \delta_{win})^T$ the Jacobian looks as follows

$$J_{\boldsymbol{fm}}(\boldsymbol{dv}) = \begin{bmatrix} \nabla {f_x}^T(\boldsymbol{dv}) \\ \nabla {f_y}^T(\boldsymbol{dv}) \\ \nabla {f_z}^T(\boldsymbol{dv}) \\ \nabla {m_x}^T(\boldsymbol{dv}) \\ \nabla {m_y}^T(\boldsymbol{dv}) \\ \nabla {m_z}^T(\boldsymbol{dv}) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial \delta_{win}}(\boldsymbol{dv}) \\ \frac{\partial f_y}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial \delta_{win}}(\boldsymbol{dv}) \\ \frac{\partial f_z}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial \delta_{win}}(\boldsymbol{dv}) \\ \frac{\partial m_x}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial \delta_{win}}(\boldsymbol{dv}) \\ \frac{\partial m_y}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial \delta_{win}}(\boldsymbol{dv}) \\ \frac{\partial m_z}{\partial l_{ext}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial l_{spa}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial c_{ro}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial r_{tap}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial \delta_{win}}(\boldsymbol{dv}) \end{bmatrix} \tag{10}$$

where the rows of $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ are the transposed gradients of every force/moment with respect to $\boldsymbol{dv}$. The Jacobian $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ of the big parametric model is computed in the same fashion but has 68 columns. The Jacobian is derived using the algorithmic differentiation tool ADOL-C, which uses the concept of operator overloading to differentiate C/C++ code. Thereby, the full source code must be available and has to be prepared for differentiation. This means all pre-compiled libraries have to be replaced in the first step before the whole routine is templated to facilitate a special datatype required for differentiation. This datatype allows the storage of primal values and to compute the derivatives. The templating includes any part of the code that influences the gradient, so from the overlaying lifting line class down to the functions of

vector- and matrix libraries, in total 8000 lines of code. Functions that interact with the remainder of the VPP, that runs in the normal datatype, have to be wrapped to ensure seamless communication. Special entry points must be implemented that can call the lifting line method with standard datatype for normal VPP solving and the special datatype for gradient computation. The included “adolc”-header provides the overloaded operators and the functions to compute the Jacobian. For more information on ADOL-C see Walther and Griewank (2020).

The remainder of the VPP computes the $V_{\mathrm{MG}}$ influenced by the forces and moments acting on the yacht produced by the various components of the yacht (e.g. sails and foils). The total forces are the sum of the components forces. To link the design variables to the performance of the yacht, the effect of the forces and moments on the yacht´s performance must be included. This requires the computation of the gradient of $V_{\mathrm{MG}}$ with respect to $\boldsymbol{fm}$, so from 6 input variables to 1 output variable. Again, the number of inputs is higher than the number of outputs, which would generally favor the adjoint method, but the total number of inputs is small. Hence, the efficiency gain of the adjoint method is negligible while the implementation time can be reduced significantly by employing finite differences. The gradient is given by

$$\nabla V_{\mathrm{MG}}(\boldsymbol{fm}) = \left[\frac{\partial V_{\mathrm{MG}}}{\partial f_x}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial f_y}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial f_z}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_x}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_y}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_z}(\boldsymbol{fm})\right]^T \tag{11}$$

and is computed with a forward finite differencing scheme. Using the chain rule $[f(g(x))]' = f'\big(g(x)\big) * g'(x)$, the two parts can be linked together, where $g'(x)$ corresponds to $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ and $f'\big(g(x)\big)$ to ${\nabla V_{\mathrm{MG}}}^T(\boldsymbol{fm})$. The resulting gradient is

$${\nabla V_{\mathrm{MG}}}^T(\boldsymbol{dv}) = {\nabla V_{\mathrm{MG}}}^T(\boldsymbol{fm}) * J_{fm}(\boldsymbol{dv})$$

$$= \left[\frac{\partial V_{\mathrm{MG}}}{\partial l_{ext}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial l_{spa}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial c_{ro}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial r_{tap}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial \delta_{win}}(\boldsymbol{dv})\right] \tag{12}$$

and represents the influence of the hydrofoil design variables on the $V_{\mathrm{MG}}$ of the yacht. For example, the partial derivative of $V_{\mathrm{MG}}$ with respect to $l_{ext}$ is hence given by

$$\frac{\partial V_{\mathrm{MG}}}{\partial l_{ext}}(\boldsymbol{dv}) = \left(\frac{\partial f_x}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial f_x} + \frac{\partial f_y}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial f_y} + \frac{\partial f_z}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial f_z} + \frac{\partial m_x}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial m_x} + \frac{\partial m_y}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial m_y} + \frac{\partial m_z}{\partial l_{ext}}\frac{\partial V_{\mathrm{MG}}}{\partial m_z}\right). \tag{13}$$

The gradient $\nabla V_{\mathrm{MG}}(\boldsymbol{dv})$ of the big parametric model is computed in the same manner but has 68 rows. $\nabla V_{\mathrm{MG}}(\boldsymbol{fm})$ remains similar. Splitting the gradient computation in two parts has a further benefit. The Newton-Raphson method solves for the steady sailing state in an iterative process. The valid steady sailing state is the last step of this process where all forces and moments equate to zero. $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ is only influenced by this final state of the boat, so the last step of the Newton-Raphson. A full differentiation using finite differences for example would, however, include one full solving process per input variable plus the initial. Hence, numerous states are computed that do not affect $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ and are not required for the approximation of $\nabla V_{\mathrm{MG}}(\boldsymbol{fm})$. With the split gradient computation this can be exploited. The steady sailing state is computed and then $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ is evaluated for the last step only. This requires little effort compared to a full solve, especially as the adjoint method is used. The gradient $\nabla V_{\mathrm{MG}}(\boldsymbol{fm})$ is then computed with 6 further VPP runs, which represents the influence of the forces and moments on the $V_{\mathrm{MG}}$. This means 7 VPP runs plus the computation of $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ and results in a significant further reduction in computational time. The Jacobians and the full gradients have

been validated with finite difference approximations of gradients and showed excellent agreement. The computation of the gradient with the combined adjoint/finite difference method is thereby 36 times faster than a purely finite difference-based approach for 68 design variables.

## 8. General design optimisation

The first optimisation using the VPP-driven framework is conducted with the simple parametric model. The results of this optimisation are compared to the findings of the parametric study to ensure a correct implementation and set-up of the optimisation routine. In the parametric study, 72 different designs were tested comprising of any combination of the values listed in the "Values"-column in Table 4.3. The $V_{\mathrm{MG}}$ of these candidates were computed for up- and downwind conditions in 8, 11 and 14 kts $V_{\mathrm{TW}}$.

**Table 4.3: Comparison Set-up**

| Parameter | Values | Initial Design |
|---|---|---|
| Extension length $l_{ext}$ | 0.6 m, 0.95 m, 1.3 m | 0.95 m |
| Semi span $l_{spa}$ | 1.75 m, 1.88 m | 1.815 m |
| Root chord $c_{ro}$ | 0.4 m, 0.8 m | 0.6 m |
| Taper ratio $r_{tap}$ | 0.25, 0.5 | 0.375 |
| Wing rake $\delta_{win}$ | 0.0°, 2.0°, 4.0 ° | 2.0° |

For the comparison the upwind condition in 11 kts $V_{\mathrm{TW}}$ was selected, the results of which are shown in Figure 4.10. In the upwind conditions in 11 kts $V_{\mathrm{TW}}$, candidate 15 with the lowest root chord (0.4 m) and taper ratio (0.25) performed best. The small half wings result in low-wetted area foils as a larger portion of the weight is stored in the bulb. The round shape of the bulb has less wetted surface area per weight. This is accompanied by the lowest extension length (0.6 m), which again reduces wetted surface area and hence profile drag. The induced drag is reduced by maximising the span (1.88 m). The wing rake of 4 degrees is faster than the other settings. The influence is however negligible as the design variable wing rake is nearly identical to the control variable rake which is used for balancing the forces in z-direction. Candidate 15 was 12% faster than candidate 70, the slowest in the study, and demonstrates the immense influence of hydrofoil design on yacht performance.

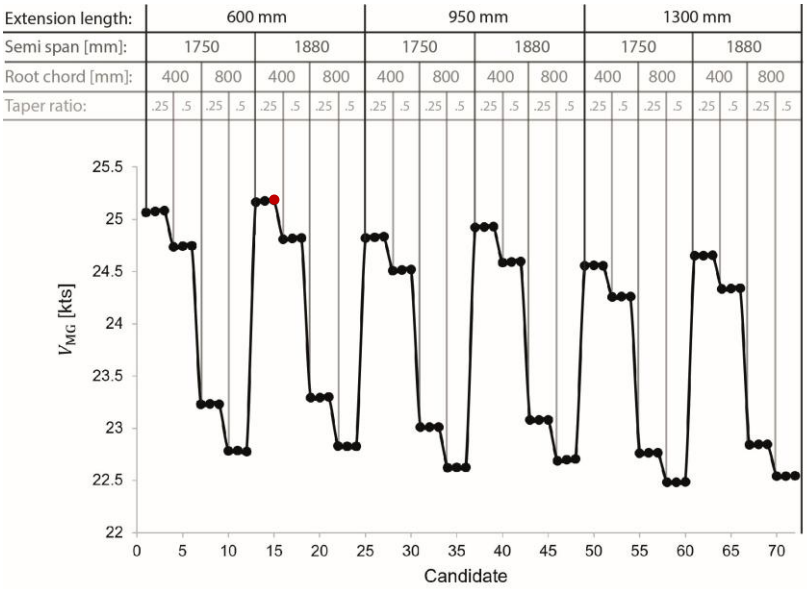



**Figure 4.10: Foil performance in 11 kts at $\beta_t$ = 47.5°, candidate 15 with highest $V_{\mathrm{MG}}$ in red**

The corresponding adjoint optimisation run with the simple mode is started from the middle of the value ranges in Table 4.3 which is detailed in the “Initial Design”-column and shown in Figure 4.12a. The lowest and highest values of every variable are used as bounds. The convergence criterium is set to $1x10^{-4}$ in conjunction with an acceptable level criterium stopping the run if the changes are below $1x10^{-3}$ for five consecutive iterations. The maximum number of iterations is limited to 25.

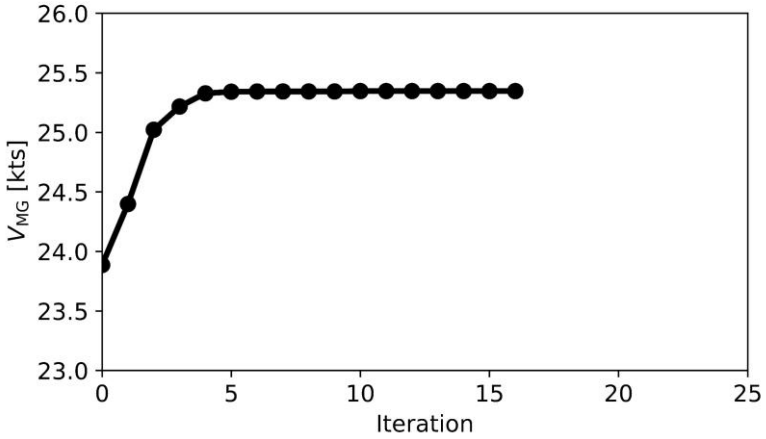



**Figure 4.11: Convergence history of the general design optimisation in 11 kts upwind**

The optimisation run converges to the optimal solution within 16 iterations as shown in Figure 4.11. From a practical point of view however the optimal foil is already found at the fifth iteration after which the further design changes are negligible. The final foil is presented in Figure 4.12b and exactly corresponds to candidate 15, the fastest candidate from the parametric study. This proves the correct functioning of the approach. The optimum foil is 1.44 kts and hence 6.02% faster than the initial foil. This corresponds to a time saving of 12.8 s on a 1.5 nautical mile upwind leg. Two more optimisation runs were started, one from the lower bounds and one from the upper bounds. Both runs again identify the design corresponding to candidate 15 as optimal. The runs are solved to the acceptable level in 12 and 16 iterations, respectively. This suggests a uni-modal design space and reinforces the results from the initial run. A pure finite differences-based run found the same result. Due to the low number of input variables both methods were equally fast in this case and solved in a matter of minutes on a standard desktop PC.

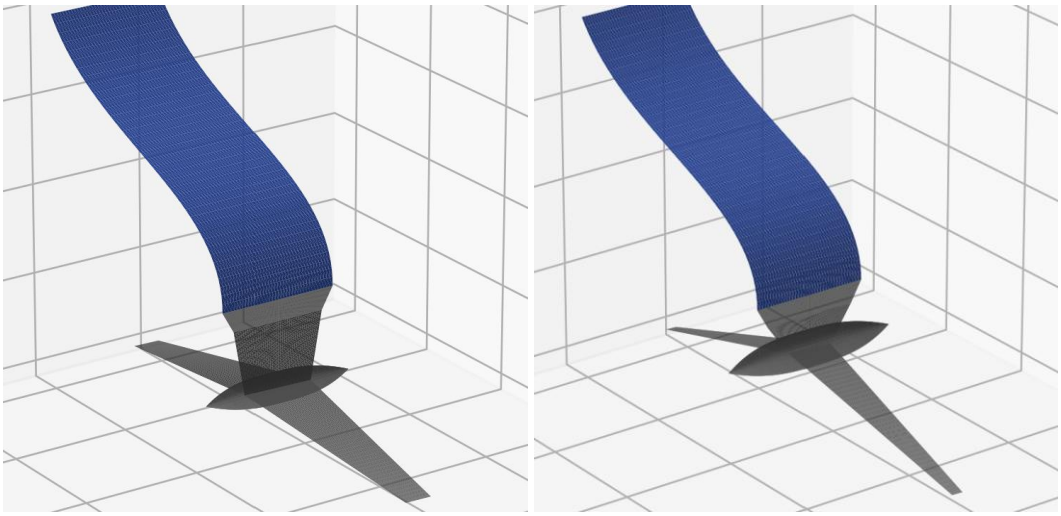


**Figure 4.12: Initial foil (a), optimal foil corresponding to Candidate 15 (b)**

The 36th America´s Cup was won by ETNZ with a yacht that was clearly the fastest in the fleet. The boat featured a T-foil with no anhedral and this foil is believed to have played a major role in the yacht´s superior performance. This is in contrast to the findings of the parametric study that suggests that the higher the anhedral angle is, the better the foil performs. The reason that the ETNZ´s T-foil performed better than the Y-foils is that it allowed the use of a single flap and a single control system. This meant that they could not change the direction of the force vector anymore, but it significantly reduced the required volume and wetted surface area. This possibility of reducing the required volume due to a single flap on the T-foil was not considered in the parametric study. The study is based on a constant density representative of a foil with two flaps/control systems. For a constant density foil with two flaps, the maximum anhedral foil is still superior as it has the lowest wetted surface area for the given density and is hence identified correctly as the optimal foil. A further benefit of the T-foil is that it can be used as a surface piercing foil at higher speeds which allows the reduction of wetted surface area when not needed. This is not possible for Y-foils that pierce the surface at a smaller angle which promotes ventilation. As ventilation cannot be modelled and the foil wing is assumed fully immersed at all times, this factor is also not influencing the optimisation towards a lower anhedral.

## 9. Twist-distribution optimisation

According to Prandtl (1921), the induced drag of a hydrofoil is minimal when the lift-distribution is elliptic across the span of the foil. This can be achieved through a tailored twist-distribution. To validate whether the optimisation framework is capable of identifying the correct twist-distribution, an optimisation of the twist-distribution across 15 stations is performed with the aim to maximise $V_{\mathrm{MG}}$. The optimisation is commenced for a straight T-foil without taper shown in Figure 4.13a. The green lines represent the 15 sections where the twist can be modified by the optimisation algorithm on the outboard half wing. The inboard half wing is affected in the same way to fulfill the symmetry requirement of the foil wing. The convergence criterium is set to $1\text{x}10^{-4}$ in conjunction with an acceptable level criterium stopping the run if the changes are below $1\text{x}10^{-3}$ for five consecutive iterations. The maximum number of iterations is set to 50 accounting for the higher complexity of the problem compared to the general design optimisation.

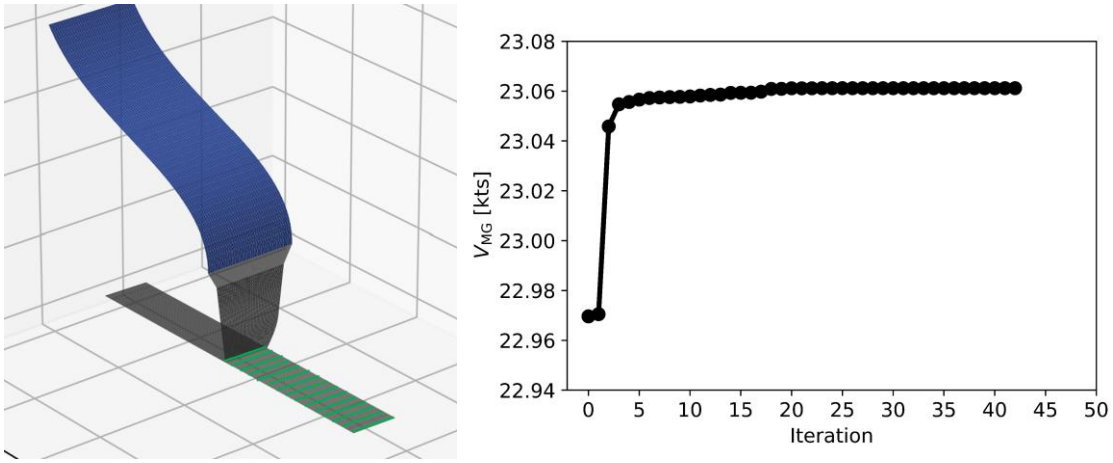



**Figure 4.13: Base foil for the twist-distribution optimisation, green lines represent the twist variables (a), convergence history of 11 kts upwind twist-distribution optimisation (b)**

The optimisation run converges after 42 iterations to the optimal solution as shown in Figure 4.13b. The twist-distribution of the initial and the optimal foil are shown in Figure 4.14a and the resulting lift-distributions in Figure 4.14b. The initial foil has a constant twist of 2° which, in

conjunction with the constant chord and the absence of sweep, leads to the rectangular lift-distribution also observed in Figure 4.7 and Figure 4.8a. This requires a reduction of the lift towards the tip (Station 15) and an increase of the lift towards the junction to achieve the desired elliptical lift-distribution. The optimal foil shows this characteristic and the resulting elliptical lift-distribution and proves that the optimisation framework is able to correctly optimise the twist-distribution for minimum induced drag. The optimal foil is 0.1 kts or 0.44% faster in terms of $V_{\mathrm{MG}}$. Two more optimisation runs have been started, one from a twist-distribution with a decreasing twist towards the tip and one from a distribution with an increasing twist towards the tip. Both runs identified the same optimal twist-distribution as the initial run and solved to the acceptable level within 39 and 47 iterations, respectively. The runs solved in roughly 30 minutes on a standard desktop PC.

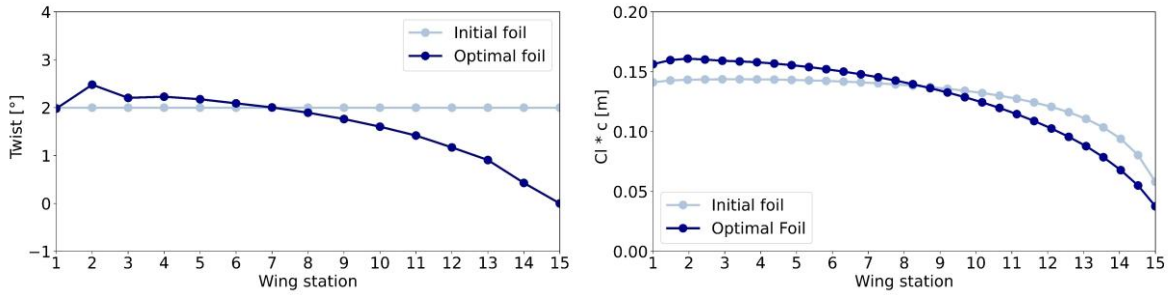



**Figure 4.14: Twist-distribution (a) and lift-distribution (b) of the initial and optimal foil**

## 10. Detailed optimisation

Following the successful optimisations of the general design of the foil and the twist-distribution, a detailed optimisation is performed. The detailed optimisation uses the large parametric model with 68 design variables which allows the optimisation of the general design, the twist-distribution and a detailed spine- and plan form in a single run. The large, hybrid adjoint/finite-difference gradient was validated with the full finite difference gradient showing excellent agreement between the two methods. The computation of the gradient using the hybrid method is thereby 36 times faster than the full finite difference method. This demonstrates the immense efficiency of the developed approach.

The detailed optimisation is also started from the initial geometry shown in Figure 4.12a. The same limits on extension length, semi span and chord along the wing apply (smallest allowed chord at every station corresponds to the chord at this location with minimum root chord and minimum taper ratio in the small parametric model). This allows validation with the previous results and a fair comparison between the general design optimisation and the detailed one. The model has additional freedom in the x- and z-position of every station (defined through the respective leading-edge point). The x-values can be manipulated by +- 0.2 m and the z-values can move between the bottom of the box and the waterline in the canted case. Bounding the foil at the waterline rather than the top of the box ensures that all designs are fully submerged. This still allows foils such as candidate 15 with the minimum extension length. Furthermore, the twist at every station is open to optimisation with +- 3°. The chord of the extension can be changed by +- 0.2 m from the initial design. The optimisation run is subject to the same convergence criteria as the previous runs with a maximum number of 150 iterations to account for the much higher dimensionality. The convergence history is shown in Figure 4.15.

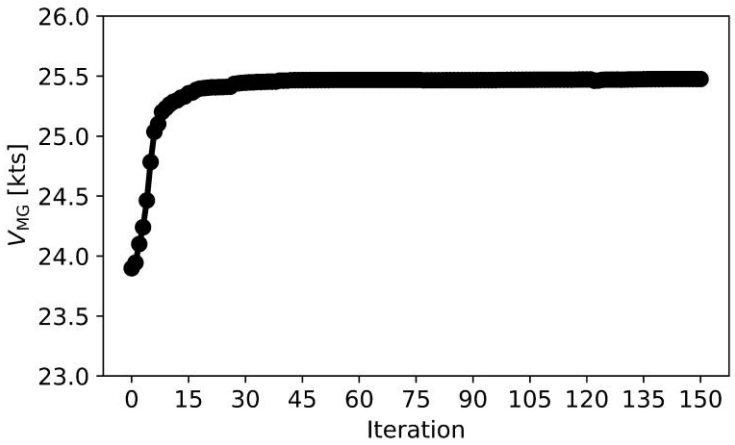



**Figure 4.15: Convergence history of 11 kts upwind optimisation, detailed model**

The optimisation run terminates with the maximum number of iterations of 150 reached. The routine fails to fully converge to the given criteria within this limit. A higher number of maximum iterations has also not led to convergence, which suggests that the gradients are not perfectly accurate or the function has a very flat valley. The convergence history also shows small dents in the performance at for example iteration 120, which further suggests that the gradients are not perfectly accurate in some cases. These slight inaccuracies are introduced in the second part of the gradient computation. Here, the accuracy of the gradient is depending on the convergence of the Newton-Raphson determining the sailing state. Different step sizes and convergence criteria for the Newton-Raphson have been tested and the best setting has been used for the above run. Other settings have led to worse convergence in the optimisation. The step size of the gradient computation of $\nabla {V_{\mathrm{MG}}}^{T}(\boldsymbol{fm})$ also plays a role. A delta of 0.001 N and 0.001 Nm has delivered the best results. However, despite the lack of perfect convergence, the changes in the design towards the end of the optimisation are negligible from an engineering perspective. Hence, the convergence is deemed sufficient.

The final foil is 1.58 kts and hence 6.6 % faster than the initial foil in terms of $V_{\mathrm{MG}}$. For an upwind leg of 1.5 nautical miles, this results in 14 s time saving over the initial foil. Compared to the optimal foil of the general design optimisation, the detailed optimisation has increased the performance by a further 0.58 % or 0.139 kts $V_{\mathrm{MG}}$. Over a single race consisting of six legs, this results in a 7.2 s timesaving or a 94 m lead (4.1 boat lengths). This is a very significant advantage in this highly competitive event, where every fraction of a second and every meter counts. Having a faster yacht additionally has strategic and tactical advantages that are likely to amplify these performance gains.

The final foil has the same extension length and semi span as Candidate 15 which demonstrates the correct functioning of the routine with respect to the general design. The chord at all 15 half wing stations is minimised to the lower bounds. This results in the same root chord and taper ratio as Candidate 15 which was also at the lower bounds and proves that the method can find the correct results for high numbers of design variables. The chord at the stations of the extension is reduced as much as possible to further reduce wetted surface area. The bulb is therefore slightly larger. This was expected and is reasonable. The final foil is shown in Figure 4.16.

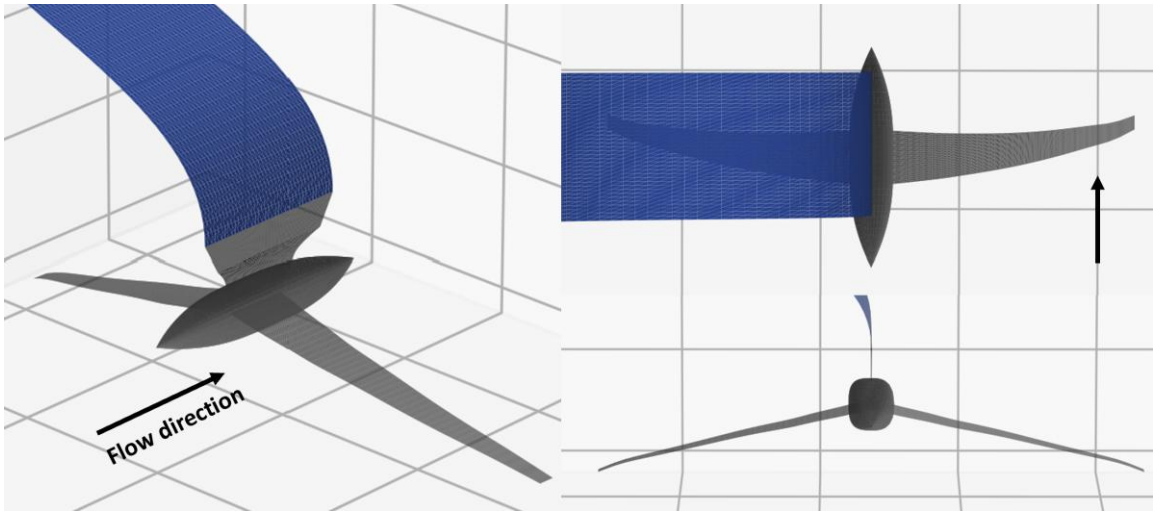



**Figure 4.16**: **Final foil of the detailed optimisation**

The view from the top in Figure 4.16 shows that the final foil of the detailed optimisation is slightly swept aft and has a curved leading edge. This was not directly expected but appears to help in achieving the elliptical lift-distribution shown in Figure 4.17b. A separate optimisation run where the sweep was prohibited, however, also achieved an elliptical lift-distribution through a slightly higher twist towards the tip and a slightly lower one close to the junction. Therefore, it is not clear why the sweep should be beneficial. However, it is striking how smooth the swept and curved leading edge is. Whether this is only the optimum of the numerical model or also physically cannot yet be determined.

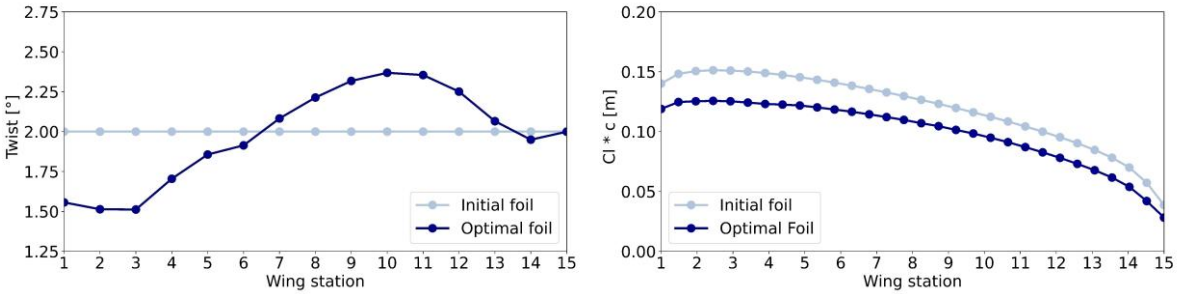



**Figure 4.17: Twist-distribution (a) and lift-distribution (b) of the initial and the detailed optimal foil**

The twist-distributions of the initial and the final foils are shown in Figure 4.17a. The initial foil has a constant twist of 2° which leads to a slightly triangular load distribution. The optimal foil has a lower twist towards the root (Station 1) and a higher twist towards the tip to counter this trend. The result is the desired elliptical lift-distribution that results in minimum induced drag. This is shown in Figure 4.17b. The initial foil produces a higher lift as the optimised candidate when expressed in terms of $C_L * c$. This is due to the larger surface area of the small foil. However, the optimised foil is resulting in higher speeds and the forces produced by the hydrofoil scale with the velocity squared. If the velocity is considered, both foils produce similar amounts of vertical lift and can support the yacht.

Several more optimisation runs have been started from different initial designs. This included different plan forms, spine forms and twist-distributions. In some cases, the same optimum as in the first run was found. In others, the foil designs were not able to achieve force equilibrium anymore after a certain number of iterations. In these cases, it is clear that the final foils are not the optimum. It shows that the results of such a detailed optimisation must be treated with caution. Nevertheless, the method has delivered consistent results from many other starting

points reinforcing the findings of the initial run. The optimisation runs solved in roughly 1.5h on a standard desktop PC, demonstrating the efficiency of the method.

The results prove the capability of the method to conduct detailed design, while it is also capable of optimising the general dimensions such as the span and the extension length. This capability has resulted in a significant $V_{\mathrm{MG}}$ gain of 0.139 kts compared to the general design optimisation. The gains are expected to be even higher in optimisations for slower wind and hence boat speeds where the induced drag and the twist distribution has a higher influence. In addition, the detailed model allows further freedom to explore the design space, such as winglets for example, which was herein limited to enable validation and fair comparison with the general design optimisation. This shows the immense potential of the developed approach. Due to the high efficiency of the adjoint method, the framework can be extended to thousands of design variables which would allow to include the optimisation of hydrofoil sections and structural aspects.

## 11. Conclusions

Hydrofoils are the biggest performance differentiators in the “yacht” system and significantly affect the state and trim of the yacht. This requires the modelling of the entire system in an optimisation, which was achieved through the use of a stationary physics model of the entire yacht. It also requires detailed optimisation to fully extract the immense potential, which is prohibitively expensive with conventional optimisation strategies. This can only be achieved with a gradient-based optimisation algorithm paired with the adjoint method to compute the gradients. The adjoint version of the physics model was derived using the algorithmic differentiation tool ADOL-C for the bottleneck of the routine. The remainder of the physics model was differentiated with finite differences to reduce development time. The resulting routine computes the gradient of 68 input variables 36 times faster than the conventional full finite difference method and is the key to detailed system-based optimisation. To the authors knowledge, this makes the developed approach the first adjoint velocity prediction program for yachts and ships in general. The approach is not limited to hydrofoils on racing yachts but is beneficial whenever a component of a system has a huge influence on the system itself, provided that the model of the system is differentiable. The routine was first used to optimise the five variables of the parametric study, the results of which show excellent agreement to the findings of the optimisation. In a second step, a twist-distribution optimisation using 15 twist variables was conducted. The resulting twist-distribution led to the desired elliptical lift-distribution. Finally, a detailed optimisation of the entire foil with 68 design variables was performed. The results show that the method is capable of optimising the foil to great detail. The optimisation included the full spine- and planform as well as the twist distribution over 19 stations. The resulting foil has a minimum extension and minimum chord everywhere, paired with a maximum span and a twist-distribution that exhibits the elliptical lift-distribution. The detailed optimisation has increased the $V_{\mathrm{MG}}$ by 1.58 kts compared to the baseline foil. This is a very significant performance gain in the America´s Cup world. The optimisation run solved in 1.5h on a standard desktop PC. Due to this high computational efficiency, the framework can be extended to optimise thousands of design variables and would allow the inclusion of structural and section design. Further work is commenced to include additional true wind angles and true wind speeds to optimise the foil for a broader set of conditions.

## 12. References

America´s Cup (2020). Virtual Eye, (viewed 8-Dec-2022), https://ac36.americascup.com/en/advanced-dashboard.

Bingol, O. (2022). NURBS-Python Visualization, (viewed 5-Dec-2022), python.readthedocs.io/en/5.x/visualization.html.

Eggert, F. (2018). *Flight Dynamics and Stability of a Hydrofoiling International Moth with a Dynamic Velocity Prediction Program (DVPP)*. Technische Universität Berlin, Germany.

Drela, M. and Youngren, H. (2022). Athena Vortex Lattice, (viewed 10-Apr-2024), https://web.mit.edu/drela/Public/web/avl/.

Faltinsen, O. (2005). *Hydrodynamics of High-Speed Marine Vehicles*. Cambridge University Press, Cambridge, UK.

Garg, N., Kenway, G., Martins, J. and Young Y. (2017) High-fidelity Multipoint Hydrostructural Optimization of a 3-D Hydrofoil. *Journal of Fluids and Structures*, 71, 15-39.

Gattini, S. (2020). PRADA ACWS - Day 1 Photo Gallery, (viewed 22-Jun-2021), https://www.lunarossachallenge.com/en/gallery/614_PRADA-ACWS-Day-1-Photogallery.

Griewank, A. and Walther, A. (2008). *Evaluating Derivatives - Principles and Techniques of Algorithmic Differentiation*. Society for Industrial and Applied Mathematics, 2nd Edition, Philadelphia, USA.

Hansen, H., Hochkirch, K., Burns, I., and Ferguson, S. (2019). Maneuver Simulation and Optimization or AC50 Class, *Journal of Sailing Technology*, 4, 142-160.

Hart, F. P., Kriplani, N., Luniya, S. R., Christoffersen, C. E. and Steer, M. B. (2006). Streamlined Circuit Device Model Development with fREEDAR® and ADOL-C, in Bücker, M., Corliss, G., Naumann, U., Hovland, P. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Berlin, Heidelberg, Springer Berlin Heidelberg, 295–307

Hochkirch, K. (2018). *FS-Equilibrium User Manual*.

Hoerner, S. F. (1965). *Fluid-Dynamic Drag*. Published by the Author.
Hospodář, P., Drábek, A. and Prachař, A. (2022). *Aerodynamic Design and Strength Analysis of the Wing for the Purpose of Assessing the Influence of the Bell-Shaped Lift Distribution*. Aerospace 9, no. 1: 13

Hovland, P., Bischof, C., Spiegelman, D. and Casella, M. (1997). Efficient Derivative Codes through Automatic Differentiation and Interface Contraction: An Application in Biostatistics, *SIAM Journal on Scientific Computing*, vol. 18, no. 4, 1056–1066.

Jayaraman, P. (2022). GitHub – TinyNURBS, (viewed 31-May-2022), https://github.com/pradeeppyro/tinynurbs.

Jee, K. W., McShan, D. L. and Fraass, B. A. (2005). Implementation of Automatic Differentiation Tools for Multicriteria IMRT Optimization, in Bücker, M., Corliss, G., Hovland, P., Naumann, U. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Springer.

Katz, J. and Plotkin, A. (1991) *Low-speed aerodynamics: From wing theory to panel methods*, New York, McGraw-Hill

Kim, J. G., Hunke, E. C. and Lipscomb, W. H. (2006). Sensitivity analysis and parameter tuning scheme for global sea-ice modeling, *Ocean Modelling*, vol. 14, 1-2, pp. 61–80.

Melis, F. M., Hansen, H., Fischer, M. and Abdel-Maksoud, M. (2022). Velocity Prediction Program for a Hydrofoiling Lake Racer. *Journal of Sailing Technology*, 7, 255-275.

Melis, F. M., Tannenberg, R., Boyd, S. W., Scharf, M. and Abdel-Maksoud, M. (2024). AC75 Aerodynamic Force Prediction Using a 3D Panel Code, 8th High Performance Yacht Design Conference, Auckland, New Zealand.

Meneghello, G., Beyhaghi, P. and Bewley, T. (2016). *Simulation-based optimization of the hydrofoil of a flying catamaran*. Massachusetts Institute of Technology, Cambridge, US.

Molland, A. F. and Turnock, S. R. (2022) *Marine rudders, hydrofoils and control surfaces: Principles, data, design and applications*, Oxford, Cambridge, Butterworth-Heinemann an imprint of Elsevier.

Offshore Racing Council (2001). International Measurement System.

Paulin A., Hansen H., Hochkirch K. and Fischer M. (2015). Performance Assessment and Optimization of a C-Class Catamaran Hydrofoil Configuration, 5th High Performance Yacht Design Conference Auckland, New Zealand.

Phillips, W. F. and Snyder, O. D. (2000). Modern Adaptation of Prandtl's Classic Lifting-Line Theory. *Journal of Aircraft,* 3, 662-670.

Ploe, P. (2018). *Surrogate-based optimization of hydrofoil shapes using RANS simulations*. École centrale de Nantes, France.

Prandtl, L. (1918). *Tragflügeltheorie.* Königliche Gesellschaft der Wissenschaften zu Göttingen, Göttingen, Germany.

Prandtl, L. (1921) *Applications of modern hydrodynamics to aerodynamics: NACA Report No. 116*.

SumToZero (2021) *GOMBOC Designer*, (viewed 31 January 2021), http://sumtozero.com/products/gomboc-designer/.

Röbenack, K. (2007). Controller design for nonlinear multi-input – multi-output systems based on an algorithmic plant description, *Mathematical and Computer Modelling of Dynamical Systems*, vol. 13, no. 2, 193–209.

Tadjouddine, M., Forth, S. A. and Keane, A. J. (2006). Adjoint Differentiation of a Structural Dynamics Solver, in Bücker, M., Corliss, G., Naumann, U., Hovland, P. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 309–319.

Tannenberg, R., Turnock, S.R., Hochkirch, K. and Boyd, S.W. (2023). VPP Driven Parametric Design of AC75 Hydrofoils. *Journal of Sailing Technology*, 8, 161-181.

Walther, A. and Griewank, A. (2020). *ADOL-C: A Package for the Automatic Differentiation of Algorithms Written in C/C++.*

Wächter, A. and Biegler, L. (2006). On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming 106(1),* 25-57.

# Chapter 5 Multi-Condition Hydrofoil Optimisation Using an Adjoint Velocity Prediction Program

*Tannenberg et al. (2025) "Multi-Condition Hydrofoil Optimisation Using an Adjoint Velocity Prediction Program", under review at Ocean Engineering.*

# Multi-Condition Hydrofoil Optimisation Using an Adjoint Velocity Prediction Program

**Rafael Tannenberg[a], Karsten Hochkirch[b], Stephen R. Turnock[a] and Stephen W. Boyd[a]**

[a] Faculty of Engineering and Physical Sciences, University of Southampton, Burgess Road, Southampton, SO16 7QF
[b] DNV Ship Performance Center, Brooktorkai 18, 20457 Hamburg

**Abstract.** Hydrofoils in yacht racing are operated across a wide range of conditions. Between these conditions the state and the trim of the yacht changes significantly. This must be reflected in the optimisation of the hydrofoils. Previously, we have performed hydrofoil optimisations for a single condition. These were conducted using an adjoint velocity prediction program (VPP) in a gradient-based optimisation routine. The method allows accurate optimisation with hundreds of design variables at a small computational cost. This framework is now extended to optimise foils for multiple conditions. Therefore, the VPP is solved for different conditions and returns the respective velocities made good ($V_{\mathrm{MG}}$) and gradients. The different $V_{\mathrm{MG}}$ and the according gradients are averaged to enable the multi-condition optimisation within a single objective. The framework is used to optimise the hydrofoil of an America´s Cup yacht for six conditions derived from historical weather data. The optimal foil is perfectly smooth and exhibits the expected features. It is 7.2% faster than the initial foil in terms of average $V_{\mathrm{MG}}$. This amounts to an 81.9 second timesaving over a whole race. The multi-condition optimisation can be performed at no additional cost and demonstrates the immense capabilities of adjoint VPP-based hydrofoil optimisation.

**Keywords:** Multi-Condition; Hydrofoil; Optimisation; Adjoint Method; Algorithmic Differentiation.

Rafael Tannenberg, rafael.tannenberg@soton.ac.uk

## 1. Introduction

Hydrofoils have a high lift/drag-ratio and are an efficient means to support a boat´s weight at semi-displacement or planing speeds. Therefore, they have been introduced to the America´s Cup, the Formula 1 of sailing, in 2013. They are also increasingly used to reduce the energy consumption of motor yachts and commercial small craft. The hydrofoils have a huge influence on the performance, the state and the trim of the vehicle they are applied to. Hence, it is important to optimise them accurately and thoroughly. Previous research by the authors performed a hydrofoil optimisation for an AC75 America´s Cup class yacht (Tannenberg et al., 2025). The optimisation was based on a stationary, six-degree-of-freedom physics model of the entire yacht developed in the velocity prediction program (VPP) FS-Equilibrium (Tannenberg et al. 2023). Modeling the full physics accurately captures the influence of a hydrofoil design on the yacht´s performance. The physics model was integrated into a gradient-based optimisation routine, where the gradient is computed using the efficient adjoint method. The adjoint method allows one to obtain the gradient of a function at a small computational cost, independent of the number of design variables and is the key to detailed design. This has resulted in the capability to optimise 68 design variables describing the spine- and planform, as well as the twist-distribution of the AC75 foil, in detail. Due to the enormous efficiency of the adjoint method, the number of design variables can be further increased to include the optimisation of the section shapes and the intricate structure. However, the previous optimisations were carried out for a single condition where the yacht is sailing upwind in 11 kts

true wind speed ($V_{\mathrm{TW}}$). This is not reflecting that the yacht will be operated across a wide range of conditions during an America´s Cup including up- and downwind courses in different true wind speeds.

In this work, the existing framework is extended to optimise the hydrofoil for multiple conditions. The most-likely conditions the yacht will encounter are determined from historical weather data of the venue and the time of the America´s Cup match, resulting in the three wind speeds 8, 11 and 14 kts $V_{\mathrm{TW}}$. As the America´s Cup is a windward-leeward race, an upwind and a downwind course are considered for every wind speed. At every step of the optimisation, the VPP computes the $V_{\mathrm{MG}}$ achieved in each condition and the corresponding gradients. The $V_{\mathrm{MGs}}$ and the gradients are then averaged to obtain a single objective. Weights can be applied to account for the probability of the different conditions. The average $V_{\mathrm{MG}}$ and gradient is provided to the gradient-based Interior-Point Optimiser (IPOPT) (Wächter and Biegler, 2006). The optimiser computes a new set of design variables, that results in a higher average $V_{\mathrm{MG}}$. This process is repeated until it converges to the optimal design.

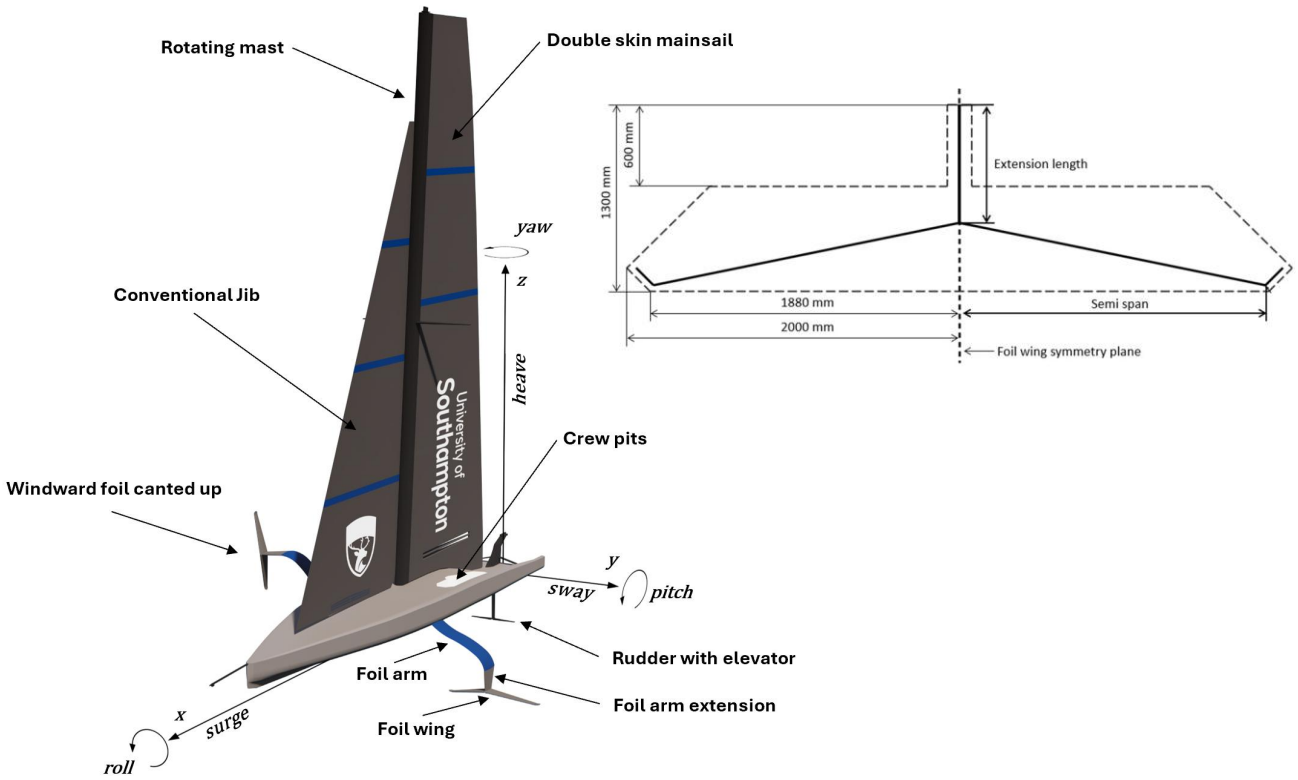



**Figure 5.1: AC75 with foil box from the AC75 rule**

The approach is used to optimise the hydrofoil of an AC75 class racing yacht as currently sailed in the America´s Cup. The yacht is shown in Figure 5.1 with its leeward foil operating in the water to generate hydrodynamic lift and side force. The windward foil is canted out of the water to provide righting moment. The design of the foil arm is mostly fixed by the rule and not included in the optimisation, while the foil wing can be developed within the `Foil box´ shown in Figure 5.1. The wing must be symmetric around the `Foil wing symmetry plane´ and have a weight of 921 kg. One or two flaps can be used to control the forces from the hydrofoil. In the last America´s Cup, Luna Rossa Prada Pirelli (LRPP) and INEOS Team UK opted for wings with a large chord that accommodated all the required weight. Emirates Team New Zealand (ETNZ) and American Magic designed wings with a smaller chord where the remaining weight was stored in a bulb. ETNZ used a single flap on a T-Foil (no anhedral angle), while the other teams chose Y-Foils with two flaps. The AC75 rule was initially written for the 36th America´s Cup in Auckland in 2021 but was slightly adapted for the next edition to be held in Barcelona

in October 2024. This paper is based on the 2021 rule to enable comparison with existing designs and performances.

The framework is first used to optimise the hydrofoil for the six conditions individually. In a second step, the multi-condition optimisation is conducted optimising the foil for the six conditions combined. The individual optimal foils are compared against the combined optimal foil.

## 2. Conditions

Since the optimal design of a hydrofoil is sensitive to the conditions in which it is operated, it is vital to identify the most likely conditions and then optimise for those. These are derived from historical weather data and for the purposes of this study for Auckland, New Zealand, where the last America´s Cup was held in 2021. The racing took place between the 10th and the 17th of March 2021. A wind limit was set allowing sailing if the true wind speed was between 6.5 and 21 kts (America´s Cup, 2021).

**Table 5.1: Average daily wind speeds in Auckland from March 10th to March 17th in the years 2010 to 2020 in kts (Weather Underground, 2024)**

| Date | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 | 2017 | 2018 | 2019 | 2020 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **10th** | 7.65 | 5.04 | 5.91 | 7.13 | 3.48 | 7.39 | 7.65 | 12.95 | 5.39 | 9.65 | 12.77 |
| **11th** | 14.60 | 6.00 | 9.04 | 5.13 | 5.21 | 4.69 | 10.95 | 12.43 | 7.73 | 4.69 | 8.69 |
| **12th** | 19.47 | 3.82 | 16.16 | 5.21 | 4.87 | 6.78 | 5.13 | 11.82 | 12.08 | 3.82 | 5.82 |
| **13th** | 13.90 | 3.91 | 11.82 | 5.82 | 4.52 | 5.65 | 4.17 | 11.99 | 18.16 | 6.34 | 5.65 |
| **14th** | 12.51 | 5.13 | 5.74 | 5.74 | 7.56 | 5.47 | 4.17 | 9.82 | 12.17 | 4.43 | 5.56 |
| **15th** | 13.73 | 3.74 | 5.91 | 6.43 | 19.12 | 11.04 | 6.00 | 8.52 | 4.17 | 6.78 | 6.60 |
| **16th** | 9.99 | 5.04 | 4.78 | 9.38 | 14.86 | 13.90 | 6.17 | 4.26 | 5.91 | 3.65 | 5.91 |
| **17th** | 11.82 | 3.74 | 9.21 | 12.51 | 7.91 | 11.38 | 6.00 | 3.74 | 6.43 | 5.65 | 13.73 |

Table 5.1 shows the daily average wind speed recorded in Auckland from the 10th to the 17th of March in the years 2010 to 2020. This amounts to 88 days in total of which 42 were within the wind limit. The average wind speed of those was 11.3 kts with a standard deviation of 3.3 kts. Therefore, the wind speeds 8, 11 and 14 kts best represent the most-likely conditions and are chosen as the base for the optimisation. The America´s Cup in its current state is a pure up- and downwind race, so an upwind and a downwind condition are considered per wind speed. This results in the six conditions shown in Table 5.2. Equal weights ($w_\mathrm{i}$) are used for the conditions, these can however be changed if the influence of particular conditions should be increased or decreased.

**Table 5.2: Conditions for the optimisation**

| Condition | $V_\mathrm{TW}$ [kts] | Course | Weight $w_\mathrm{i}$ |
|---|---|---|---|
| **1** | 8 | Upwind | 1 |
| **2** | 8 | Downwind | 1 |
| **3** | 11 | Upwind | 1 |
| **4** | 11 | Downwind | 1 |
| **5** | 14 | Upwind | 1 |
| **6** | 14 | Downwind | 1 |

## 3. Parametric model

The parametric model generates the shape of the hydrofoil based on the current set of design variables. It describes the foil´s planform, spine-form and twist-distribution through ruled surfaces and is directly connected to the lifting line method computing the hydrofoil forces. The model is shown in Figure 5.2. Four stations are used to define the extension of the foil arm with an x-coordinate of the leading-edge point and a chord-value each. The stations are automatically, equally spaced in z-direction between the end of the foil arm and the root of the foil wing. Due to the symmetry requirement, no parameters for the y-position or the twist are considered. This results in 8 variables for the extension. The foil wing is defined through the port side half wing with 15 stations and is mirrored to obtain the starboard side wing, which ensures symmetry. Each station is described with an x- and a z-value of the leading edge and a chord and a twist variable. Station 1 describes the root of the wing and is fixed in the middle of the box for symmetry purposes. Station 15 defines the tip of the wing with an additional y-variable setting the span of the foil but is fixed at the bottom of the foil box in terms of z-position. The intermediate stations are automatically spaced equally in between in the y-direction resulting in 60 parameters for the foil wing. The full 68 variables are combined in the design variable vector

$$\boldsymbol{dv} = (\mathrm{FAE}_{x1}\,,\ \mathrm{FAE}_{c1}\,,\ \mathrm{FAE}_{x2}, \ldots\,, \mathrm{FWP}_{t15})^{T}. \tag{1}$$

$\mathrm{FAE}$ corresponds to foil arm extension and $\mathrm{FWP}$ is the port side foil wing. $x, y, z$ are the coordinates of the leading-edge point of a station, $c$ its chord length and $t$ its twist. The index denominates the station number. The cross-sectional properties are defined through the lift curve slopes, the zero lift angles and the profile drag coefficients at every station and are not part of the optimisation. For more details see Tannenberg et al. (2025). The model is based on the C++ library TinyNURBS (Jayaraman, 2022) and uses the python library NURBS-python (Bingol, 2022) for visualisations.

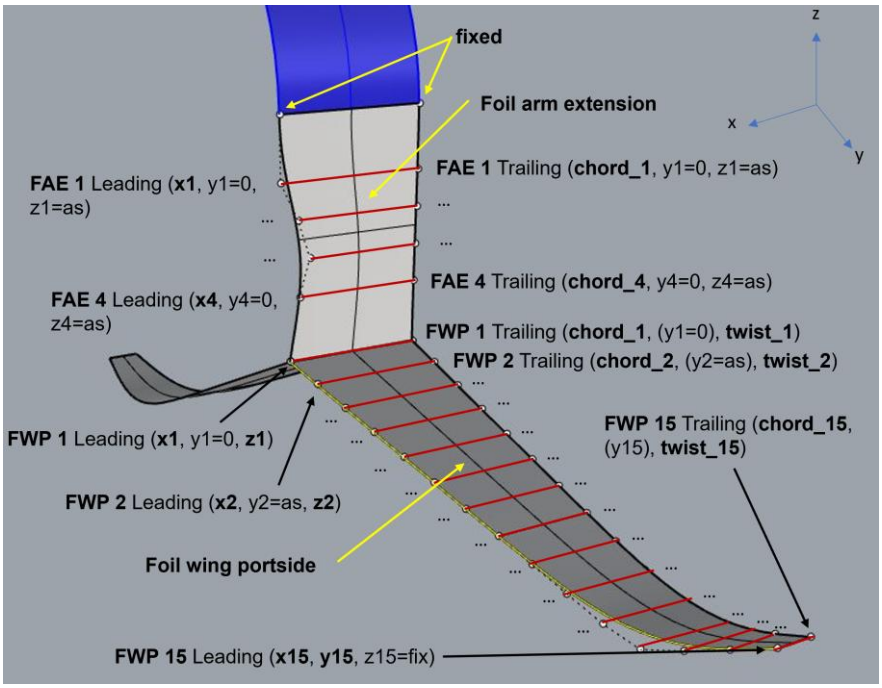



**Figure 5.2: Detailed parametric model with 68 variables; FAE = Foil arm extension; FWP = Foil wing port; as = automatically spaced**

## 4. Velocity prediction model

The physics model of the yacht is developed in FS-Equilibrium, a velocity prediction tool that can solve for stationary and dynamic sailing states in six degrees of freedom. The program has been successfully applied to foiling yachts in the past. Examples are the C-Class Catamaran Groupama (Paulin et al., 2015), the International Moth (Eggert, 2018), the AC50 (Hansen et al., 2019) and the QFX Lake Racer (Melis et al., 2022). The AC75 model is briefly introduced in this section, for a more thorough description see Tannenberg et al. (2023). For more information on the VPP in general see Hochkirch (2018). The VPP model is shown in Figure 5.3.

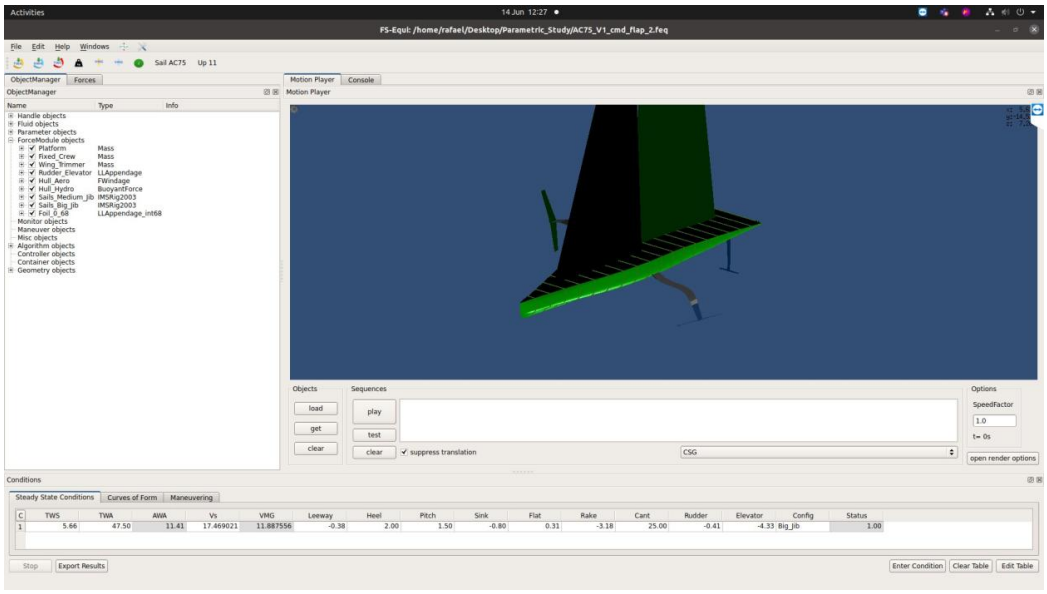



**Figure 5.3: Graphical user interface of FS-Equilibrium showing the AC75 model**

For the performance prediction of the AC75, the stationary mode of FS-Equilibrium is employed with a tailored solver set-up. The program solves for sailing states where all forces and moments

$$\boldsymbol{fm} = \left(f_x, f_y, f_z, m_x, m_y, m_z\right)^T \tag{2}$$

equate to zero. This is achieved using a Newton-Raphson method that alters specified state and trim variables until force equilibrium is found and returns the resulting velocity made good,

$$V_{\mathrm{MG}}(\boldsymbol{fm}). \tag{3}$$

The state and trim variables used are shown in Table 5.3 and best represent the sailing style of an AC75 yacht.

**Table 5.3: Solver set-up of the AC75 physics model**

| Degree of freedom | Balancing variable |
|---|---|
| $f_x$ | boat speed $V_S$ (state) |
| $f_y$ | leeway angle λ (state) |
| $f_z$ | rake angle $\delta_{\mathrm{ra}}$ (trim) |
| $m_x$ | sail-power (flat) τ (trim) |
| $m_y$ | elevator rake $\delta_e$ (trim) |
| $m_z$ | rudder angle $\delta_r$ (trim) |

“Force Modules” are employed to compute the forces from the components of the yacht such as the hydrofoils and the sails. The full list of modules is given in Table 5.4. The forces and moments generated by the hydrofoils are modelled using an advanced lifting line method (LLT)

(Phillips and Snyder, 2000). The foil arm extension is discretised with 10 stations, 30 stations are used per half wing. This discretisation has delivered the most robust optimisation results, for more information see Tannenberg et al. (2025). Additionally, the model computes the forces and moments from the weight of the foil based on an integration of the sectional areas along the span and an average density of 8305 kg $m^{-3}$ (75% high density steel, 20% lead and 5% hydraulic oil to represent the control systems). The bulb is sized automatically to fulfil the weight requirement. The method is directly coupled to the parametric model and returns the hydrofoil forces as a function of the design variables,

$$\boldsymbol{fm}(\boldsymbol{dv}). \tag{4}$$

This results in the velocity made good being a function of the design variables considering the behaviour of the entire yacht

$$V_{\mathrm{MG}}(\boldsymbol{dv}). \tag{5}$$

The force module of the hydrofoil and the entire yacht model have been validated, for details see Tannenberg et al. (2025). It currently predicts boat speeds that are slightly too high which might be a consequence of the tuned aero model. A more accurate model of the full sail plan based on a Boundary Element Method is currently under development (Melis et al., 2024) and will be implemented in the future.

**Table 5.4: AC75 force modules**

| Component | Model | Comments |
|---|---|---|
| Platform gravity | Gravity model | Weight and CoG from AC75 rule |
| Crew gravity | Gravity model | Weight from rule, CoG estimated |
| Hull aero | Coefficient-based | Coefficients from Hoerner (1965) |
| Sails aero | Tuned IMS2003 | Factor of 1.3 on driving force |
| Rudder hydro | Lifting Line | Based on Phillips and Snyder (2000) |
| Main foil hydro | Lifting Line (adjoint) | Same as for Rudder, but adjoint |

## 5. Multi-condition gradient-based optimisation strategy

The velocity prediction model of the AC75 is integrated into an automated optimisation routine. A gradient-based strategy is chosen over a derivative-free approach as these are much more computationally expensive and prohibit detailed optimisation with high numbers of design variables.

The developed methodology starts with an initial set of design variables. Based on this set, the parametric model generates the hydrofoil shape and passes it to the hydrofoil force model. The VPP is now run solving for the equilibrium sailing state of the AC75 in the first condition with the initial hydrofoil. This is an iterative process where the hydrofoil forces are calculated numerous times by the lifting line method for changing states and trims (VPP loop). Once a valid equilibrium sailing state is found, the VPP returns the achieved $V_{\mathrm{MG}}$. In a second step, the gradient for this condition is computed. This is explained in detail later. The gradient is stored, and the process is repeated for the next condition. Once, all six conditions have been evaluated, the gradients are averaged and passed to the optimisation algorithm IPOPT. IPOPT identifies a better performing set of design variables which is sent to the parametric model again. A new averaged gradient is computed and passed to IPOPT again (optimisation loop). The process is repeated until the design has converged to the optimum. This is shown below in Figure 5.4.

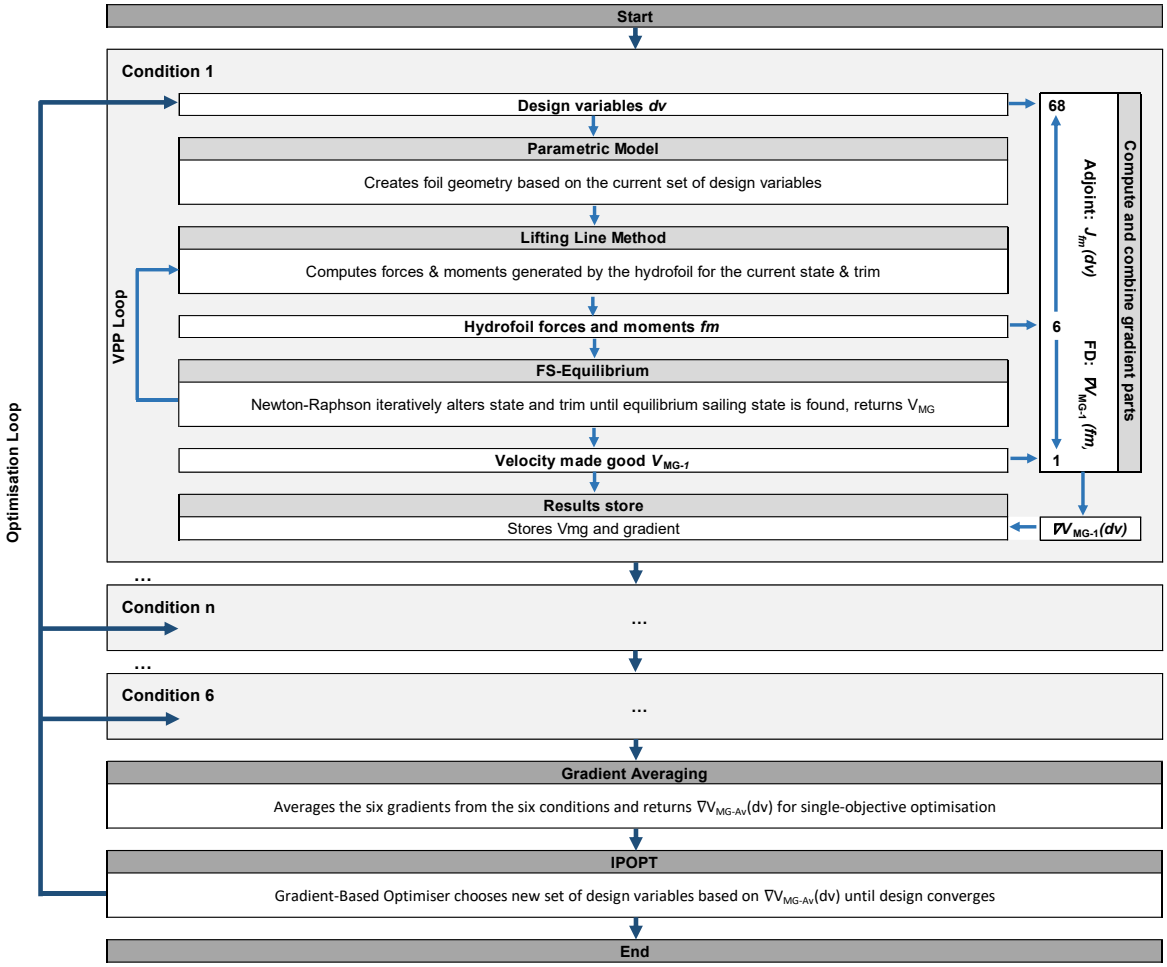



**Figure 5.4: Multi-condition optimisation approach using the adjoint VPP**

The most expensive part of gradient-based optimisation is the computation of the gradients. These are usually approximated using the finite difference method where a single design variable is perturbated, the whole function is evaluated, and the influence of the design variable is computed from the change in the output and the step size of the perturbation. This process must be repeated for every other design variable to obtain a single full gradient. Therefore, for a function with 68 input variables (hydrofoil design variables $\boldsymbol{dv}$) and 1 output ($V_{\mathrm{MG}}$), the approximation of the full gradient requires 69 function evaluations (VPP runs). This is prohibitively expensive as the gradient must be computed hundreds of times during an optimisation run.

The only feasible approach to compute the gradient of a function with a very high number of input/design variables and a low number of outputs is the adjoint method which corresponds to a reverse application of the chain rule to a function. By reversely applying the chain rule, the gradient can be computed with a single partial derivative per output variable, independent of the number of input variables. This is opposed to forward differentiation and finite difference which require the computation/approximation of one partial derivative per input variable. Therefore, the adjoint approach allows for very high numbers of design variables to be optimised at a small computational cost which is the key to detailed optimisation.

However, the reverse differentiation of a complex function is a challenging task. Algorithmic differentiation tools can speed up the process and reduce errors but still require extensive preparation of the code for differentiation. The finite difference method on the other hand is simple to implement. Therefore, a hybrid approach was developed that uses the adjoint method for the bottleneck of the function and finite difference for the remainder of the program.

The two gradient parts are then combined using the chain rule. This makes the run time independent of the number of input variables, while it significantly reduces development time. The resulting gradient is computed at a small multiple of a single normal evaluation of the VPP model, for any number of design variables.

The parametric model of the foil and the lifting line method are the bottleneck of the routine. Here, the gradients of the 6 forces and moments $\boldsymbol{fm}$ with respect to the 68 input variables $\boldsymbol{dv}$ have to be computed. The six gradients are combined in the Jacobian matrix

$$J_{\boldsymbol{fm}}(\boldsymbol{dv}) = \begin{bmatrix} \nabla {f_x}^T(\boldsymbol{dv}) \\ \nabla {f_y}^T(\boldsymbol{dv}) \\ \nabla {f_z}^T(\boldsymbol{dv}) \\ \nabla {m_x}^T(\boldsymbol{dv}) \\ \nabla {m_y}^T(\boldsymbol{dv}) \\ \nabla {m_z}^T(\boldsymbol{dv}) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial f_x}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial f_x}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \\ \frac{\partial f_y}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial f_y}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial f_y}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \\ \frac{\partial f_z}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial f_z}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial f_z}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \\ \frac{\partial m_x}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial m_x}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial m_x}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \\ \frac{\partial m_y}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial m_y}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial m_y}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \\ \frac{\partial m_z}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) & \frac{\partial m_z}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) & \dots & \frac{\partial m_z}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \end{bmatrix} \quad (6)$$

where the rows of $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ are the transposed gradients of every force/moment with respect to $\boldsymbol{dv}$. $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ has 68 columns for the 68 design variables. Using the adjoint method, the Jacobian matrix can be calculated by evaluating six partial derivatives only (as opposed to 68 with the conventional methods). This holds true for any number of design variables. The reversely differentiated version of the parametric model and the lifting line method is thereby obtained using the algorithmic differentiation tool ADOL-C for differentiation of C/C++ code. For more information on ADOL-C see Walther and Griewank (2012) and on algorithmic differentiation in general see Griewank and Walther (2008).

The remainder of the VPP computes the $V_{\mathrm{MG}}$ influenced by the hydrofoil forces and moments. This requires the computation of the gradient of $V_{\mathrm{MG}}$ with respect to $\boldsymbol{fm}$. With 6 inputs and 1 output, the number of input variables is higher than the number of output variables. This does again favor the adjoint method, but since the absolute number of inputs is small, the efficiency gain is negligible. Hence, the finite difference method is used here which significantly reduces development time. The gradient is given by

$$\nabla V_{\mathrm{MG}}(\boldsymbol{fm}) = \left[ \frac{\partial V_{\mathrm{MG}}}{\partial f_x}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial f_y}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial f_z}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_x}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_y}(\boldsymbol{fm}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial m_z}(\boldsymbol{fm}) \right]^T \quad (7)$$

and is calculated using a forward finite differencing scheme. Employing the chain rule $[f(g(x))]' = f'\big(g(x)\big) * g'(x)$, the two gradient parts can be linked together, where $g'(x)$ corresponds to $J_{\boldsymbol{fm}}(\boldsymbol{dv})$ and $f'\big(g(x)\big)$ to $\nabla {V_{\mathrm{MG}}}^T(\boldsymbol{fm})$. The resulting gradient is

$$\nabla {V_{\mathrm{MG}}}^T(\boldsymbol{dv}) \;=\; \nabla {V_{\mathrm{MG}}}^T(\boldsymbol{fm}) * J_{fm}(\boldsymbol{dv}) \;=$$

$$= \left[ \frac{\partial V_{\mathrm{MG}}}{\partial \mathrm{FAE}_{x1}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial \mathrm{FAE}_{c1}}(\boldsymbol{dv}) \quad \frac{\partial V_{\mathrm{MG}}}{\partial \mathrm{FAE}_{x2}}(\boldsymbol{dv}) \quad \dots \quad \frac{\partial V_{\mathrm{MG}}}{\partial \mathrm{FWP}_{t15}}(\boldsymbol{dv}) \right] \quad (8)$$

and represents the influence of the hydrofoil design variables on the $V_{\mathrm{MG}}$ of the yacht. The gradient has 68 rows. As shown in Figure 5.4, this process must be performed for every condition. The six resulting gradients are then combined in a weighted average,

$$\nabla {V_{\mathrm{MG-Av}}}^{T}(\boldsymbol{dv}) = \frac{\sum_{i=1}^{6} w_i \ \nabla {V_{\mathrm{MG-i}}}^{T}(\boldsymbol{dv})}{\sum_{i=1}^{6} w_i}, \tag{9}$$

resulting in a single gradient for single-objective optimisation where $i$ denominates the condition and $w_i$ are the weights. This multi-condition, but single-objective approach allows much faster optimisation than a multi-objective optimisation and is the preferred option as the desired weighting of the conditions should be based on the expected likelihood of the conditions. Averaging the gradients is the same as differentiating the averaging of the different $V_{\mathrm{MG}}$ but can be conducted in parallel and is hence faster.

The Jacobians and the full gradients have been validated with finite difference-based gradients and showed excellent agreement. For more details on the gradient computation and the application of ADOL-C to the parametric model and the lifting line method see Tannenberg et al. (2025). This also discusses the exploitation of the iterative nature of the Newton-Raphson method for even faster gradient computation. To further reduce computational time, the six conditions and their gradients can also be computed in parallel as opposed to the serial computation shown in Figure 5.4. This means the multi-condition optimisation would then have a run-time similar to that of a single-condition optimisation (around 1.5 h).

## 6. Individual optimisations

Individual optimisations for the six conditions have been performed for comparison with the multi-condition optimisation. Figure 5.5 shows the resulting hydrofoil shapes and Table 5.5 gives relevant metrics of the optimum designs and their performance. The individual optimisations are all started from the same initial geometry that was also used in Tannenberg et al. (2025). This geometry represents the middle of the design space with respect to the bounds on extension length, span, chord- and twist-distribution, as well as the x-position of the leading-edge points. The bounds are set to ensure that the foil candidates will stay within the limits of the foils box and that no part of the foil wing will be above the waterline. The semi span of the foil is limited to 1.88 m, so the pocket in the box for a potential winglet is not considered, but an optimisation including a winglet will be commenced in the future. In the x-direction every leading-edge point can be manipulated by +- 0.2 m. The twist at every station can be changed within +- 3°. Minimum and maximum chord-values are set for every wing station. The values are chosen such that the foils will not become too thin structurally or too large so that the foil will be above the 921 kg. The chord of the extension stations can be changed by +- 0.2 m from the initial design. In condition 1, the slowest of all, the boat is pitched by 0°, while a bow-down pitch of 1.5° is used for the higher speed conditions.

The convergence criteria is set to $1\text{x}10^{-4}$ in conjunction with an acceptable level criterium stopping a run if the changes are below $1\text{x}10^{-3}$ for five consecutive iterations. The maximum number of iterations is limited to 150.

The six runs showed good convergence with the planform and the spine-form optimised early in the optimisation where their gradients are strongest while the twist-distribution with smaller influence, and therefore weaker gradients, was optimised at higher iteration numbers. All runs were terminated through the maximum iteration criterion. A single run took around 1.5 h to solve on a standard desktop PC. Also in some runs, the best performing foil wasn´t the last candidate, which is a consequence of imperfect convergence. It is believed to be a consequence of the iterative nature of the solver in conjunction with the use of finite difference for computing the VPP gradient. The solution, if not converged absolutely perfectly, reduces the accuracy of the gradient which is amplified by the use of the finite difference. At later stages of the optimisation, where the last little details with small influence are optimised, these

inaccuracies can hinder the convergence process. Nevertheless, the convergence is still good and delivered expected and high performing designs.

**Table 5.5: Design details and performance of the six individually optimised hydrofoils**

| Condition/ Foil | Extension length [m] | Semi span [m] | Lift-distribution | $V_{\mathrm{MG}}$ [kts] | Gains $V_{\mathrm{MG}}$ [kts] | Gains $V_{\mathrm{MG}}$ [%] | Gains [s/1.5nm] |
|---|---|---|---|---|---|---|---|
| 1 | 0.6 | 1.88 | trapezoidal | 18.32 | 1.47 | 8.74 | 25.76 |
| 2 | 0.6 | 1.88 | elliptical | 23.96 | 1.94 | 8.79 | 19.81 |
| 3 | 0.6 | 1.88 | elliptical | 25.48 | 1.58 | 6.6 | 13.99 |
| 4 | 0.6 | 1.779 | trapezoidal | 33.7 | 2.36 | 7.52 | 12.04 |
| 5 | 0.6 | 1.88 | elliptical | 29.14 | 1.65 | 6.01 | 11.13 |
| 6 | 0.6 | 1.75 | elliptical | 39.87 | 2.67 | 7.17 | 9.71 |

In all six conditions the extension length is minimised, i.e. the anhedral angle is maximised. This minimises the wetted surface area and hence the profile drag for the given immersed volume. This was also found in a parametric study (Tannenberg et al, 2023). However, it contrasts with what was seen in the last America´s Cup, ETNZ with a T-foil (no anhedral angle) had the fastest boat. The T-foil was performing better as it allowed the use of a single flap and control system, which reduced the volume required inside the foil and hence the associated wetted surface area. The possibility of reducing the required volume through the single-flap approach was not considered in this optimisation. So, for a foil of constant density with two flaps, the maximum anhedral foil is still superior. It has the lowest wetted surface area and is therefore correctly identified as the optimal foil. The single-flap approach could be explored in a separate run where a higher foil density is assumed and the extension length of the foil is fixed to the maximum of 1.3 m, which corresponds to a T-foil. Another benefit of the T-foil is that it can be operated as a surface piercing foil at high speeds. This allows the reduction of wetted surface area when not needed which is not possible for Y-foils that pierce the surface at a smaller angle as it promotes ventilation. However, ventilation cannot be modelled, and the foil wing is assumed fully immersed at all times. The lack of ventilation is not influencing the optimisation towards a lower anhedral and the maximum anhedral foil remains the optimal design for the physics modelled. While all six foils have the same extension lengths, they do not share the same spine-form. Foils 1, 2 and 3 have straight half wings, where only the tip is curved down slightly. The higher-speed foils 4 and 6 have half wings that are entirely curved downwards. Candidate 5 has a spine that is first curved upwards and then curved downwards, which might be caused by convergence problems.

The span of the foil is at the maximum for the slow and medium boat speed conditions (1, 2, 3, and 5), but is reduced for the higher speed conditions (4 and 6). Increasing the span reduces the induced drag which is dominant at lower speeds. At higher speeds, the profile drag is dominant, which can be reduced by reducing the area of the foil. It outweighs the higher induced drag caused by the shorter span.

The chords across the 15 wing stations and the four extension stations is minimised to the lower bounds in every condition. Consequently, more of the weight is stored in the bulb which has a lower ratio of wetted surface area to volume. Therefore, profile drag is reduced, while induced drag, which depends on the span only, is not affected. Hence, the reduction of the

chord has a huge influence on the performance and is relevant in all assessed conditions. However, if the chord becomes too small, structural problems are to be expected and the take-off speed is increased. This is not modelled in the optimisation and is the reason for the use of the lower bounds.

A notable feature is the aft sweep of the foils and the even further aft raked wingtips. The slower foils 1, 2 and 3 have significant sweep with convex leading edges, while the higher speed foils 4, 5 and 6 have more moderate sweep and straight or slightly concave leading edges. All foils feature an even more aft raked wingtip. No amount of sweep or wingtip rake was expected, so further investigations are commenced to identify the reason for the sweep. In a first step it is investigated why the higher speed foils do not exhibit significant sweep while this seems to be beneficial for the lower speed foils. The higher speed foils generally require more righting moment, which is achieved by shifting the centre of effort of the foil aft. This results in the elevator having to produce lower lift or even down force to maintain pitch equilibrium. As a consequence, more vertical lift has to be produced by the main foil to support the weight of the yacht. The main foil is further outboards than the elevator and hence creates more righting moment. Therefore, it is expected that the entire foil is pulled back when optimised for higher speed conditions. This can be observed for the root of the foils which is significantly further aft in the higher speed designs. The tips however remain in roughly the same position as before reducing the sweep. However, the position of the tips is already very close to the lower bounds potentially limiting the sweep. For this reason, a separate optimisation for condition 6 is started with additional freedom. In this run, the bounds of +-0.2 m on the x-values of the leading edge are extended to +-0.4 m. While the root remains in the same position, the tip is now shifted aft much further. This leads to a similarly significant sweep as for the low-speed foils and shows that the sweep is not driven by the conditions but must result in generally higher efficiency. In a second step, the root of this unexpected efficiency gain is investigated. Therefore, an optimisation run for condition 3 is commenced that has exactly the same freedom as previous runs, but without the ability to sweep. The run identifies the same spine form and chord distribution as for the swept foil, with a slightly different twist-distribution to achieve the elliptical lift-distribution for the non-swept version. The non-swept foil is 0.13 % slower in terms of $V_{\mathrm{MG}}$ compared to the swept foil and has a slightly lower lift-to-drag- and side-force-to-drag-ratio. Analysing the different drag components, it is found that profile drag, and the bulb drag are nearly identical while the induced drag is 4.6% lower on the swept wing. A plot of the induced drag components per station shows that every station generates induced drag on the non-swept wing. The swept wing however, produces a small “induced thrust” on the two most outward stations of the tips. The next few stations going further inboards produce induced drag but much smaller than on the non-swept wing. This causes the overall reduced drag of the swept wing. This feature is believed to be similar to the raked wingtips on aircrafts such as the Boeing 787 Dreamliner (Boeing, 2024) and should be investigated further. See for example Norton (1921) or Gold and Visser (2012). It might be a feature not yet known and exploited in hydrofoil design and could lead to efficiency gains of many other hydrofoils. However, winglets are still believed to result in a higher efficiency for this limited-span problem, but they have been prohibited by the bounds of this study. A separate optimisation should be performed including the freedom for winglets.

An elliptical lift-distribution is known to exhibit minimum induced drag. The lift-distribution is mainly influenced by the planform, the chord-distribution and the twist-distribution. However, it was observed in many runs that the spine-form, the planform and the chord-distribution are first optimised to achieve minimum profile drag. The twist-distribution is then optimised at later iterations to achieve the elliptical lift-distribution and hence minimum induced drag. Four of the six runs found a twist-distribution that results in the desired elliptical lift-distribution. These were runs 2, 3, 5 and 6. The optimal foils for condition 1 and 4 did not achieve an elliptical lift-distribution. This was not expected, especially for condition 1 with the lowest boat speed, where induced drag has the highest influence. This appears to be caused by convergence problems that hinder IPOPT from optimising the finer details.

In general, the optimisations have achieved very significant gains in performance, even if not fully converged at all times. The $V_{\mathrm{MG}}$ of the six optimal foils is given in Table 5.5 along with the

gains over the initial foil in kts, % and the deltas in s/1.5nm. 1.5 nm is the length of a typical leg in an America´s Cup race. A single race usually consists of six such legs resulting in 9 nm total. The results show that the optimisations have on average increased the $V_{\mathrm{MG}}$ by 7.47%. Foil 1 achieved an 8.74% gain which amounts to a time saving of 25.76 seconds over a 1.5 nm upwind leg. Foil 2 achieved a 19.81 s time saving on a single 1.5 nm downwind leg. This shows the immense potential of hydrofoil design and the enormous capability of the developed approach to extract this potential.

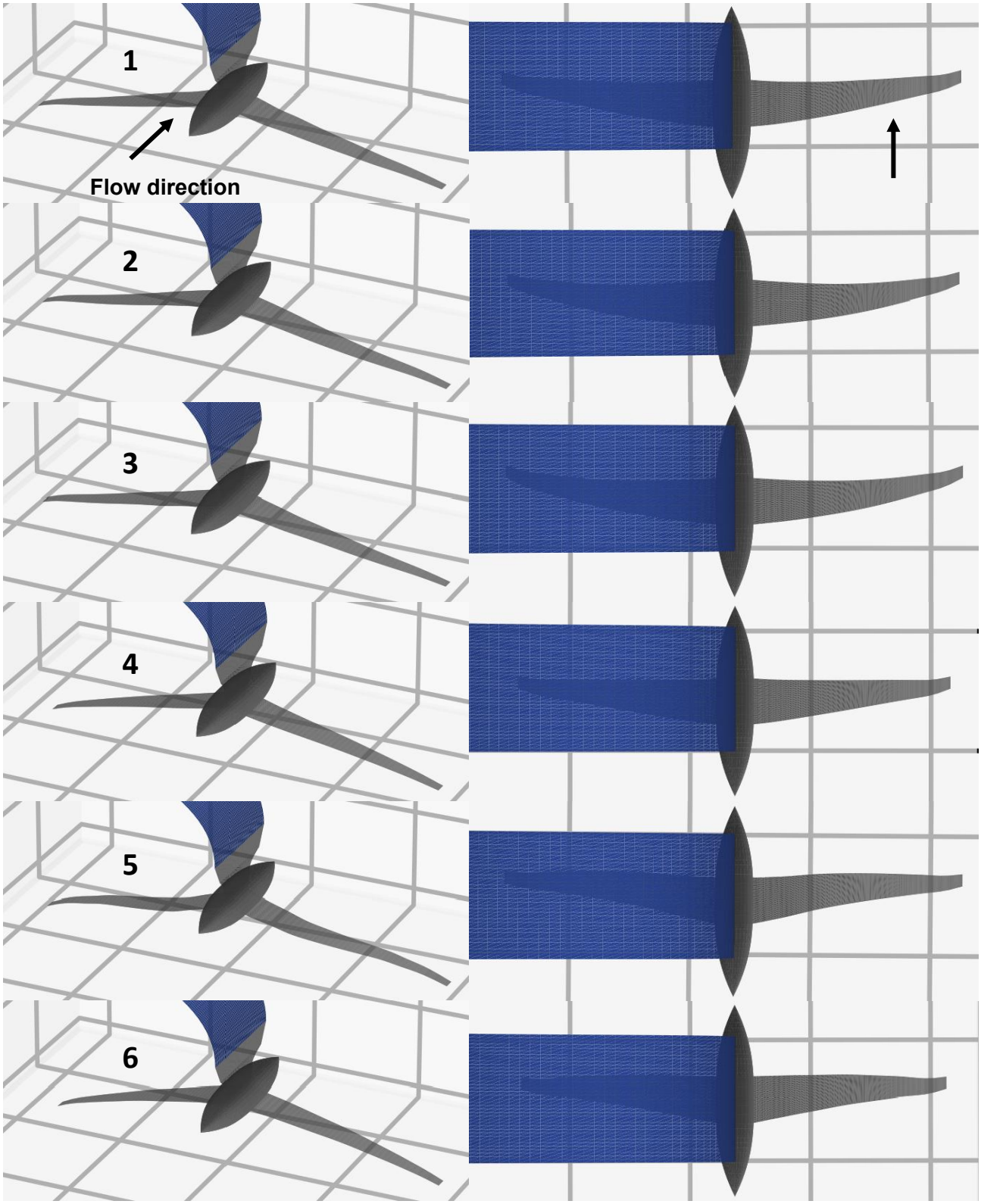



**Figure 5.5: Plan- and spine-forms of the six individually optimised hydrofoils**

## 7. Multi-condition optimisation

The multi-condition run is started from the same initial foil as runs 1 – 6 and shares the same convergence criteria. The same bounds apply on spine- and planform design. The bounds on the twist variables have been reduced from +-3° on the initial value to +-2°. This yields better convergence for this optimisation and should not affect the design as the changes in twist are usually well below +-2°. Initially, a maximum of 150 iterations was used. In previous single-condition runs, extending the number of maximum iterations has not led to better designs. In the multi-condition optimisation however, better performing candidates have been found after iteration 150, so the limit is extended to 200 iterations. Figure 5.6 shows the convergence of the multi-condition run. The optimisation converges well with ever evolving performances with the exception of a little dent around iteration 60. The best design is found at iteration 163. After that no better designs are found, hence they are not shown in the convergence history. The run terminates with reaching the maximum number of iterations. The run has solved in around 9 h on a standard desktop PC, but parallelising the gradient computation would mean that it could also be performed in around 1.5 h.

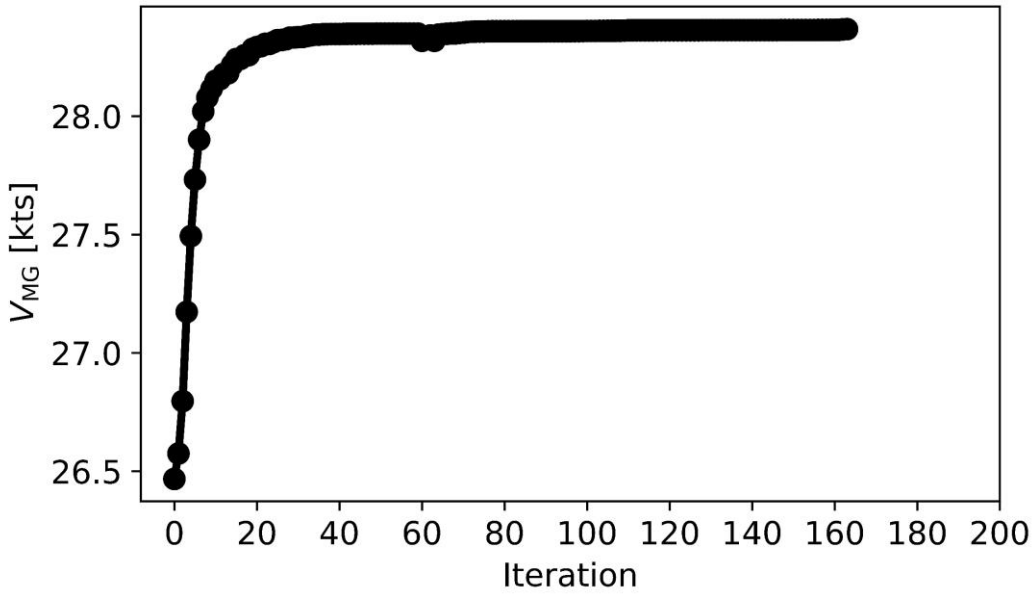



**Figure 5.6: Convergence history of the multi-condition optimisation run**

The optimum multi-condition foil is shown in Figure 5.7. It has an extension length of 0.6 m and minimum chord at each station, so like the individually optimised foils. The semi span of the foil is at the maximum with 1.88 m. The spine of the half wing is mostly straight and similar to foil 2, whose speed is in the middle of the range. The leading edge of the foil is convex, like the lower speed foils 1, 2 and 3, but has a more moderate sweep. The twist-distribution is tailored to achieve an elliptical lift-distribution in all conditions. The final twist-distribution is shown in Figure 5.8. Station 1 is the junction and Station 15 the tip of the foil.

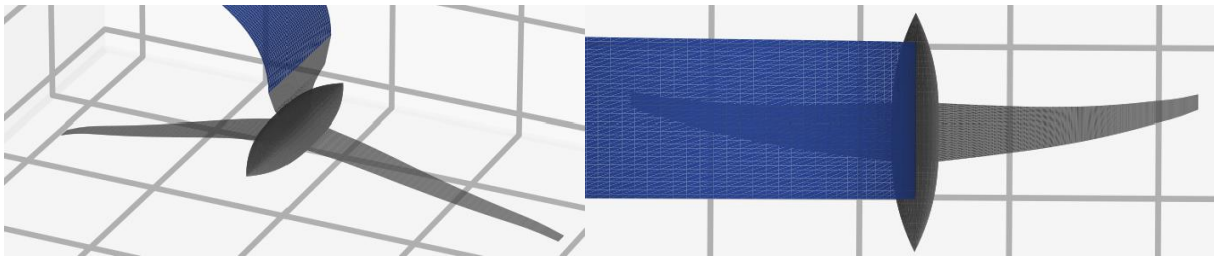


**Figure 5.7: Plan- and spine-form of the optimal multi-condition foil**

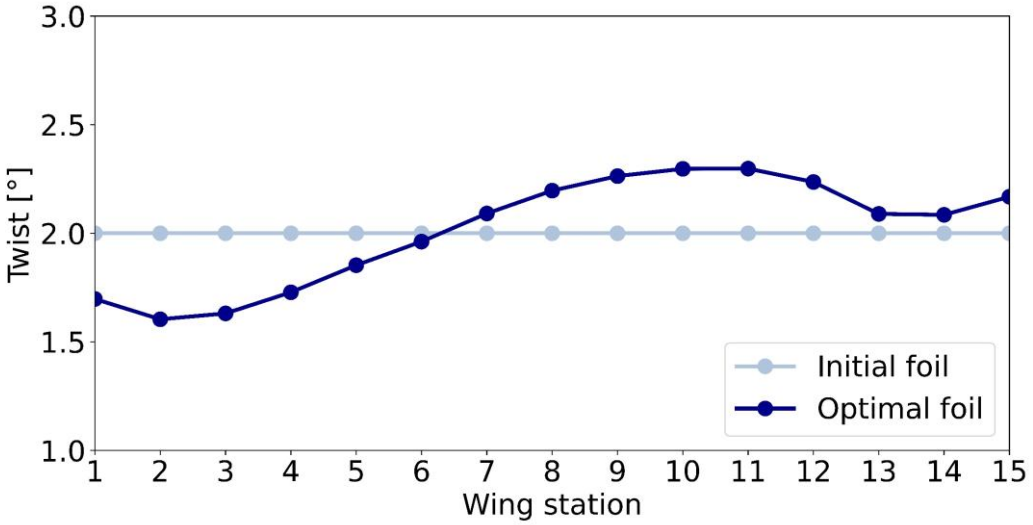



**Figure 5.8: Optimum twist-distribution of the multi-condition foil**

The optimal foil reaches an average $V_{MG}$ of 28.37 kts compared to foil 0 which achieved 26.47 kts and results in a 1.9 kts or 7.2% gain. This is very significant with a time saving of 13.65 seconds per 1.5 nm leg or 81.9 seconds per 9 nm race. It once again demonstrates the high importance of hydrofoil optimisation and the enormous power of the developed approach. Several more starting points have been tested and resulted in either the same optimum design or quickly developed very unrealistic geometries for which no equilibrium could be found anymore.

### 8. Comparison of the optimal foils

Three comparisons are performed to verify the correct functioning of the optimisation routines and to acquire an understanding of the differences in the optimal foil designs. First, the individually optimised foils are compared to each other in all six individual conditions. This is shown in Table 5.6 and demonstrates how much the optimal foils differ for the different conditions. The performance is given in seconds per leg of the race, where one leg is typically around 1.5 nm.

**Table 5.6: Foil performance of the individually optimised foils in the six individual conditions in seconds per leg (1 leg = 1.5 nm)**

| Cond. / Foils | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 294.79 | 294.90 | 294.91 | 296.35 | 296.67 | 297.79 |
| 2 | 227.49 | 225.35 | 227.31 | 227.74 | 227.74 | 227.81 |
| 3 | 212.40 | 212.01 | 211.96 | 212.34 | 212.03 | 212.48 |
| 4 | 160.75 | 160.44 | 160.47 | 160.22 | 160.54 | 160.11 |
| 5 | 185.88 | 185.51 | 185.49 | 185.28 | 185.34 | 185.18 |
| 6 | 136.41 | 136.18 | 136.21 | 135.77 | 136.10 | 135.44 |

The results reflect that a foil specifically optimised for one condition should perform best in this condition, while the foils optimised for other conditions should perform less well. Foil 1 for example is specifically optimised for condition 1 and hence performs best in this condition. Foils 2 to 6 are optimised for ever faster conditions and therefore perform less well in condition 1 with descending order. In condition 1, Foil 1 is 3 s faster on a leg of 1.5 nm than foil 6, the slowest foil in this condition. This amounts to a lead of 1.22 boat lengths at the end of the leg.

This is a significant advantage in an America´s Cup race in terms of direct performance, but even more so as it opens strategic opportunities that can lead to additional performance gains. This trend is continued in conditions 2, 3 and 6. In all of those, the foil that is specifically optimised for the respective condition performs best, whereas the foils optimised for slower/faster conditions perform less well. This proves that the routine is sensitive and capable enough to identify optimal foils for their respective conditions. Only in condition 4 and 5, this is not the case. Here, foil 6 is faster than foils 4 and 5, respectively. This is believed to be a consequence of the imperfect convergence, that lead to the non-elliptical lift distribution of foil 4 and the unexpected spine form of foil 5. However, Foil 6 is only 0.11s faster than foil 4 in condition 4 and 0.16s faster than foil 5 in condition 5, so the differences are marginal.

**Table 5.7: Optimal single-condition foils vs optimal multi-condition foil in the individual conditions**

| Condition | Single-best $V_{MG}$ [kts] | Multi-best $V_{MG}$ [kts] | Delta [s/1.5nm] | Delta [boat lengths/1.5nm] |
|---|---|---|---|---|
| 1 | 18.32 | 18.30 | 0.27 | 0.11 |
| 2 | 23.96 | 23.96 | 0.00 | 0.00 |
| 3 | 25.48 | 25.48 | 0.01 | 0.00 |
| 4 | 33.70 | 33.67 | 0.17 | 0.13 |
| 5 | 29.14 | 29.12 | 0.08 | 0.05 |
| 6 | 39.87 | 39.67 | 0.67 | 0.60 |

In the second test, the optimal multi-condition foil is now analysed for every single condition. Its performance is compared to the performances of the respective optimal single-condition foils. The individual foils should perform better as they are optimised for their respective individual condition only as opposed to the multi-condition foil which is optimised for the range of conditions. The results are shown in Table 5.7. In five of the six conditions the optimal single-condition foil performs better than the optimal multi-condition foil generally verifying the correct functioning of the routines. In condition 2 however, the multi-condition foil achieves the same speed. The reason for this is that the average boat speed encountered in the multi-condition is very close to the boat speed of condition 2. This means the foils are optimised for nearly the same condition. Still, foil 2 should be slightly faster in condition 2, but run 2 has potentially not converged as well as the multi-condition run and the resulting design is hence similar in speed. Nevertheless, the other single-condition foils do perform better than the multi-condition foil, with increasing deltas towards the slowest and fastest conditions.

**Table 5.8: Optimal single-condition foils vs optimal multi-condition foil in the multi-condition**

| Foil | Average $V_{MG}$ [kts] | [Delta s/ 9nm] | Delta [boat lengths/9nm] |
|---|---|---|---|
| Multi | 28.37 | 0.00 | 0.00 |
| 1 | 28.29 | 3.32 | 2.11 |
| 2 | 28.36 | 0.27 | 0.17 |
| 3 | 28.33 | 1.70 | 1.09 |
| 4 | 28.33 | 1.47 | 0.94 |
| 5 | 28.30 | 2.53 | 1.61 |
| 6 | 28.33 | 1.31 | 0.83 |

However, hydrofoils are used across a range of conditions rather than a single condition, so the third test compares the different optimal foils on how they perform across the six conditions combined. This includes the optimal multi-condition foil, as well as the optimal single condition foils. Here, the multi-condition foil should perform best, as it is optimised exactly for this trade-off of conditions. The foils are compared in terms of average $V_{\mathrm{MG}}$ and delta to the multi-condition foil. A typical race in the 36th America´s Cup consisted of six legs with 1.5 nm each, which amounts to 9 nm per race. As the multi-condition rather represents a whole race than a single leg, the deltas are now given in boat lengths per 9nm. The results are shown in Table 5.8. As expected, the optimal multi-condition foil performs best, as it was specifically optimised for this trade-off of conditions. Foils 1-6 which perform best in their respective niches, do perform less well across the board of conditions. This again confirms the correct functioning of the single-condition and the multi-condition optimisations and demonstrates their sensitivity and capability. The multi-condition foil is between 0.27 s and 3.32 s faster than the individual foils. This amounts to a lead at the end of the race of 0.17 and 2.11 boat lengths, respectively.

In general, this study has shown four things:

- Hydrofoil optimisation is immensely important and the developed adjoint VPP in combination with IPOPT is a powerful approach. The method allows the optimisation of 68 design variables with a full physics model of the AC75, where traditional methods could only optimise a handful of parameters. This powerful approach has achieved an average gain in $V_{\mathrm{MG}}$ of 7.47% in the single-condition optimisations. In condition 1 it resulted in a 25.76 s time saving on a single upwind leg, which is race-dominating.

- The sensitivity of optimal hydrofoil design to the conditions in which the foil is operated was shown. Foil 6 for example, which was optimised for condition 6 (downwind in 14 kts TWS), was 3 s slower on a single leg in condition 1 (upwind in 8 kts TWS), which was specifically optimised for this condition. The delta is smaller when compared to the delta of initial to optimised foils but is still very relevant as it would amount to a double-digit delta over a whole race.

- The optimal foils of the higher speed conditions exhibit a planform that shifts the centre of lift further aft as compared to the lower speed foils. This causes more righting moment, but also additional induced drag on the rudder and the main foil. Making the right trade-off for every condition is only possible with the full physics model and underlines the importance of system-based optimisation as it was shown that the rudder and sail forces influence the optimal foil design.

- It was shown how a hydrofoil can be optimised for a variety of conditions including up- and downwind legs in different wind speeds. The conditions were derived from historical weather data and form the best base for a reliably high performing foil design. The multi-condition optimisation gained 7.2% in average $V_{\mathrm{MG}}$ which amounts to 81.9 seconds over a whole race, which is again absolutely race-dominating. Comparing the optimal multi-condition foil to the optimal single foils, the deltas are smaller, but still significant, as every meter counts. The multi-condition foil was for example 3.3 s faster than foil 1 over a whole race with the range of conditions. This is results in a 2.11 boat length advantage at the finish.

## 9. Conclusions

Hydrofoils are operated across a wide range of conditions and their optimal design is sensitive to these conditions. This has to be accounted for in the optimisation. Therefore, a multi-condition hydrofoil optimisation strategy was developed. The strategy is based on a stationary model of the entire yacht integrated into a gradient-based optimisation routine. The gradients are computed using a combination of the efficient adjoint method and finite differences. This allows the optimisation of any number of design variables and is the key to detailed design. The existing method was previously only capable of optimising for a single condition but was extended to multi-condition optimisation in this paper. This was achieved by computing the $V_{\mathrm{MG}}$ and the gradient of multiple conditions and then averaging them at every iteration to obtain a single objective. The developed routine was applied to an AC75 type America´s Cup yacht in six different conditions. The six conditions were identified as the most-likely ones at the time and venue of the last America´s Cup and were derived from historical weather data. The resulting optimal multi-condition foil is 7.2 % faster than the initial foil, which amounts to 81.9 s time saving over a 9 nm race. In addition, the optimal multi-condition foil was compared to foils optimised for each condition individually. Here, it was shown that the multi-condition foil performs best across the range of conditions, while the optimal individual foils perform best in their respective niche. This proves the sensitivity of the physics model to changing conditions and the correct working of the optimisation routine. The multi-condition foil was 0.26 s to 3.3 s faster than the individually optimised foils over a 9 nm race in the range of conditions. The strategy can be extended to optimise for more conditions and with more design variables. It can be used to optimise the hydrofoils of any hydrofoiling craft and is not only relevant to sailing yachts but also to the increasing number of commercial and recreational hydrofoiling motor vessels.

## 10. References

America´s Cup, 2021. Prada Cup Final and 36th America´s Cup Match Amended Race Conditions and Sailing Instructions. https://www.americascup.com/news/1040_PRADA-CUP-FINAL-AND-36TH-AMERICAS-CUP-MATCH-AMENDED-RACE-CONDITIONS-AND-SAILING-INSTRUCTIONS (accessed 15 May 2024).

Bingol, O., 2022. NURBS-Python Visualization. python.readthedocs.io/en/5.x/visualization.html (accessed 5 December 2022).

Boeing, 2024. Boeing 787 Dreamliner. https://www.boeing.com/commercial/787 (accessed 10 October 2024).

Eggert, F., 2018. Flight Dynamics and Stability of a Hydrofoiling International Moth with a Dynamic Velocity Prediction Program (DVPP). Master thesis. Technische Universität Berlin, Germany.

Gattini, S., 2020. PRADA ACWS - Day 1 Photo Gallery https://www.lunarossachallenge.com/en/gallery/614_PRADA-ACWS-Day-1-Photogallery (accessed 22 June 2021).

Gold, N., Visser, K., 2002. Aerodynamic effects of local dihedral on a raked wingtip. 40th AIAA Aerospace Sciences Meeting & Exhibit, Reno, NV, U.S.A. https://doi.org/10.2514/6.2002-831.

Griewank, A., Walther, A., 2008. Evaluating Derivatives - Principles and Techniques of Algorithmic Differentiation, second ed. Society for Industrial and Applied Mathematics, Philadelphia, USA.

Hansen, H., Hochkirch, K., Burns, I., Ferguson, S., 2019. Maneuver Simulation and Optimization or AC50 Class. Journal of Sailing Technology 4, 142-160. https://doi.org/10.5957/jst.2019.4.1.142.

Hochkirch, K., 2018. FS-Equilibrium User Manual.

Hoerner, S. F., 1965. Fluid-Dynamic Drag. Published by the Author.

Jayaraman, P., 2022. GitHub – TinyNURBS. https://github.com/pradeeppyro/tinynurbs (accessed 31 May 2022).

Melis, F. M., Hansen, H., Fischer, M., Abdel-Maksoud, M., 2022. Velocity Prediction Program for a Hydrofoiling Lake Racer. Journal of Sailing Technology 7, 255-275. https://doi.org/10.5957/jst/2022.7.12.255.

Melis, F. M., Tannenberg, R., Boyd, S. W., Scharf, M., Abdel-Maksoud, M., 2024. AC75 Aerodynamic Force Prediction Using a 3D Panel Code. 8th High Performance Yacht Design Conference, Auckland, New Zealand.

Norton, F. H., 1921. An investigation on the effect of raked wing tips. Technical Note No. 69. National Advisory Committee for Aeronautics.

Paulin A., Hansen H., Hochkirch K., Fischer M., 2015. Performance Assessment and Optimization of a C-Class Catamaran Hydrofoil Configuration. 5th High Performance Yacht Design Conference Auckland, New Zealand.

Phillips, W. F. and Snyder, O. D., 2000. Modern Adaptation of Prandtl's Classic Lifting-Line Theory. Journal of Aircraft 3, 662-670. https://doi.org/10.2514/2.2649.
Tannenberg, R., Turnock, S.R., Hochkirch, K., Boyd, S.W., 2023. VPP Driven Parametric Design of AC75 Hydrofoils. Journal of Sailing Technology 8, 161-181. https://doi.org/10.5957/jst/2023.8.9.161.

Tannenberg, R., Hochkirch, K., Walther, A., Turnock, S. R., Boyd, S. W., 2025. Development of an Adjoint System-Based Hydrofoil Optimisation Framework using Algorithmic Differentiation. Currently under review at Optimization and Engineering.

Walther, A., Griewank, A. 2012. Getting started with ADOL-C. Combinatorial Scientific Computing, Naumann, U., Schenk, O. Chapman-Hall CRC Computational Science, pp. 181-202.

Wächter, A., Biegler, L., 2006. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. Mathematical Programming 106(1), 25-57. https://doi.org/10.1007/s10107-004-0559-y.

Weather Underground, 2024. Historical Weather Data Auckland, New Zealand. https://www.wunderground.com/history/monthly/nz/auckland/NZAA/date/2020-3 (accessed 15 May 2024).

# Chapter 6 Verification and validation

Every part of the optimisation routine has been verified/validated. This is reported in chapters 3 - 5. An overview of the verification and validation tests is presented in Table 6.1 with references to the relevant sections in the papers. Additional tests, which have been too comprehensive to address within the papers, are discussed in the following.

Table 6.1: Verification and validation studies

| Method | Verification/Validation | Outcome | Reference |
|---|---|---|---|
| **Yacht model - $V_{MG}$** | Comparison with recorded data from the yachts collected during the last America´s Cup, filtered for representative situations | Good agreement | Paper 1 |
| **Lifting line method - L/D-ratio** | Comparison with semi-empirical method by Whicker & Fehlner (1958) | Good agreement | Paper 1 |
| **Lifting line method - Lift-distribution** | Comparison with Athena Vortex Lattice and RANS CFD predictions | Good agreement | Paper 2 |
| **Foil weight computation** | Comparison with commercial CAD-tool | Good agreement | below |
| **Foil CoG computation** | Comparison with commercial CAD-tool | Good agreement | below |
| **Lifting line method - Gradients** | Comparison with finite difference-based gradients | Excellent agreement | below |
| **Yacht model - Gradients** | Comparison with finite difference-based gradients | Good agreement | below |
| **IPOPT** | Used on optimisation test functions | Excellent agreement | below |
| **IPOPT + ADOL-C** | Used on optimisation test functions | Excellent agreement | below |
| **General design optimisation** | Comparison with parametric study and finite difference-based optimisation | Excellent agreement | Paper 2 / below |

## 6.1 Foil weight and centre of gravity

The weight of the foil and its centre of gravity are important aspects for the performance of the hydrofoils. They must be computed based on the current hydrofoil design to be correctly considered in the optimisation.

The weight of the hydrofoil is computed using a 2.5D approach. Therefore, every foil part is discretised into several stations. At these stations, the local chord of the current design is evaluated. The local chord and the sectional area coefficient of the foil cross-section are then used to compute the sectional area at every station. These sectional areas are integrated using Simpson´s rule to obtain an approximation of the volume of the foil part. An initial study for a

simple geometry has shown that 101 stations per foil part deliver accurate results, while still being computationally cheap. The volume of the foil part and the average density of the hydrofoil are used to compute the part´s weight. The weights of the parts are summed to obtain the overall foil weight. A bulb is then automatically sized to meet the weight requirement of 921 kg per foil. The bulb is located at the junction of the extension and the foil wing. The intersection of the junction and the bulb is thereby not considered. This introduces slight inaccuracies. To understand the extent of these inaccuracies, two foils from the parametric study are analysed. Candidate 1 has very low chord, high anhedral and a large bulb, Candidate 70 has no anhedral and no bulb (Figure 6.1). Their weight is computed with the 2.5D approach and a 3D approach integrated in a commercial CAD-program.

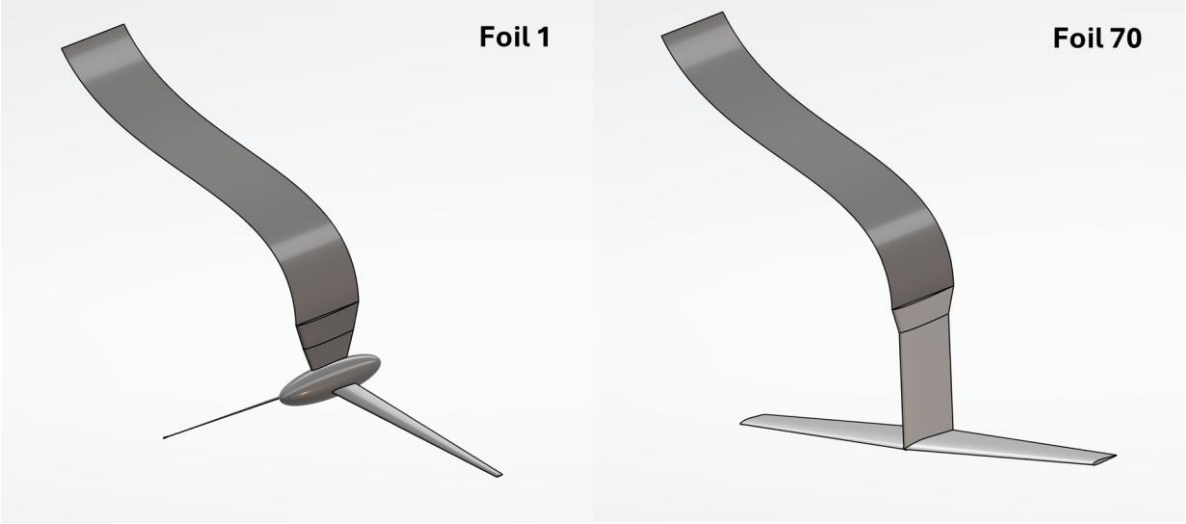



Figure 6.1: Foil candidates used for weight and centre of effort validation

The results of this comparison are presented in Table 6.2 and show an excellent agreement for the foil without bulb (1%) and good agreement for the foil with bulb (4.5%). This is deemed sufficient. Further tests have shown that curvature in the foils is another source of inaccuracy when computing the foil weight with the 2.5D approach. These curvature-related inaccuracies have amounted to differences of 3.2% for extreme cases. It was tested whether the curvature of the optimal foils in the large optimisations stems from the optimiser exploiting these inaccuracies. It has been found that the curvature is introduced for fluid-dynamic reasons and is not related to the weight computation.

Table 6.2: Weight computation comparison

| Candidate | 2.5D internal method [kg] | 3D CAD program [kg] | Difference [%] |
|---|---|---|---|
| **1** | 921 | 881.2 | 4.5 |
| **70** | 995.6 | 985.9 | 0.98 |

The centre of gravity of the foils is computed by taking moments around the sectional areas for every foil part. The lever arms of the parts are then combined for an estimate of the assembly's centre of gravity in a weighted average. The results have been compared to measurements with a commercial CAD program. The results are presented in Table 6.3. They show an excellent agreement with a maximum difference of 0.6%. A similar comparison has been performed for the foil in the canted case (25°). The comparison yielded even better agreement when compared to the non-canted case. Foils with curvature have also been tested, and in extreme cases, this has led to an error of 4%. However, such geometries have not been experienced during the optimisations. It has also been verified that the centre of gravity computation approach is not the reason for the moderate curvature developed in some of the optimisations.

Table 6.3: Centre of gravity computation comparison measured from transom (x), the centre line of the boat (y) and the waterline when floating (z)

| Candidate | 2.5D internal method [m] | 3D CAD program [m] | Difference [%] |
|---|---|---|---|
| **1 - x** | 10.5 | 10.512 | 0.1 |
| **y** | 4.76 | 4.76 | 0.0 |
| **z** | 2.284 | 2.285 | 0.0 |
| **70 – x** | 10.5 | 10.558 | 0.6 |
| **y** | 4.76 | 4.76 | 0.0 |
| **z** | 2.751 | 2.746 | 0.2 |

## 6.2 Gradients of the lifting line method

This section presents two examples of gradients computed using the adjoint lifting line method. The first gradient is computed for an upwind sailing condition, the second one for a downwind condition, both in 11 kts of $V_{\mathrm{TW}}$ and for the same design variables. The gradients are computed for the vertical lift to drag ratio of the hydrofoil with respect to its design variables. The foil shape is defined with six points, each with a x-, y- and z-value. A further three variables are used to define the position of the foil with respect to the transom of the AC75. For more detail see Section A.2. This makes a total of 21 input variables and one output variable. The design variables are shown in Table 6.4.

Table 6.4: Design variables

| Point | x [m] | y [m] | z [m] |
|---|---|---|---|
| **Extension leading edge** | 0.62 | 2.36 | -3.14 |
| **Extension trailing edge** | 0.02 | 2.36 | -3.15 |
| **Port wing leading edge** | 0.36 | 4.36 | -3.56 |
| **Port wing trailing edge** | 0.0 | 4.36 | -3.57 |
| **Starboard wing leading edge** | 0.36 | 0.36 | -3.56 |
| **Starboard wing trailing edge** | 0.0 | 0.36 | -3.57 |
| **Foil reference point** | 10.0 | 2.4 | 0.55 |

The gradients computed with the adjoint lifting line method are validated with finite difference-based gradients. A central finite difference scheme is used modifying each input variable by + and - 0.001 m. Table 6.5 and Table 6.6 show the resulting gradients in comparison with the adjoint gradients. The comparisons show excellent agreement. The highest deviation of any of the gradient components is 0.005%, the differences of the other components are even lower. This confirms the correct functioning of the adjoint lifting line method. The reason these very small differences exist is that the finite difference-based gradients are an approximation, while the gradients from ADOL-C are correct to machine precision. Only the differences of the reference point components are significant, but the absolute values of these components are virtually zero and therefore provide the correct answer. The values in Table 6.5 and Table 6.6 are presented to ten significant figures to provide a level of accuracy where the differences can be observed.

A normal evaluation of the lifting line method solved in 15 milliseconds whereas the computation of the gradient required around 105 milliseconds, so seven times as long. These values were measured for a model where the whole appendage is discretised with 60 vortices. Hence, the adjoint lifting line method is faster than finite differences from seven design variables upwards. A small model with twelve vortices resulted in a factor of only 1.5, while a large model with 120 vortices achieved a factor of nine. The VPP was compiled in debugging mode which is generally slower. The according release version is roughly twice as fast. This is true for the normal evaluation of the method as well as the gradient computation.

Table 6.5: Gradients for condition 1

| Point | Automatic differentiation [1/m] | Finite difference [1/m] | Deviation [%] |
|---|---|---|---|
| **Extension leading edge** | | | |
| **x** | 2.231239784 | 2.231261912 | -0.000992 |
| **y** | 34.17436714 | 34.17435221 | 0.000044 |
| **z** | 393.2305945 | 393.2304572 | 0.000035 |
| **Extension trailing edge** | | | |
| **x** | -5.96523216 | -5.96525455 | 0.000375 |
| **y** | -34.59042482 | -34.59042428 | 0.000002 |
| **z** | -393.0703051 | -393.0701701 | 0.000034 |
| **Port wing leading edge** | | | |
| **x** | 6.891639251 | 6.891650822 | -0.000168 |
| **y** | 43.17735333 | 43.17735466 | -0.000003 |
| **z** | 178.0717702 | 178.0716981 | 0.00004 |
| **Port wing trailing edge** | | | |
| **x** | -4.321591689 | -4.321603096 | -0.000264 |
| **y** | -35.07578983 | -35.07579095 | -0.000003 |
| **z** | -176.2732882 | -176.273223 | 0.000037 |
| **Starboard wing leading edge** | | | |
| **x** | -0.7129217119 | -0.7129217045 | 0.000001 |
| **y** | -30.54552446 | -30.5455252 | -0.000002 |
| **z** | 123.5480926 | 123.5480769 | 0.000013 |
| **Starboard wing trailing edge** | | | |
| **x** | 2.778370154 | 2.778370131 | 0.000001 |
| **y** | 23.84655041 | 23.84655092 | -0.000002 |
| **z** | -123.3571807 | -123.3571705 | 0.000008 |
| **Foil reference point** | | | |
| **x** | -6.435962874e-13 | 2.29327668e-08 | 3563322,357 |
| **y** | 6.328004787e-11 | 1.136619687e-06 | -1796073,874 |
| **z** | 6.725286994e-12 | -2.847500014e-08 | 423502,0075 |

Table 6.6: Gradients for condition 2

| Point | Automatic differentiation [1/m] | Finite difference [1/m] | Deviation [%] |
|---|---|---|---|
| **Extension leading edge** | | | |
| **x** | 0.366701691 | 0.3667195498 | -0.00487 |
| **y** | 38.77812773 | 38.77810991 | 0.000046 |
| **z** | 474.1977527 | 474.1973038 | 0.000095 |
| **Extension trailing edge** | | | |
| **x** | -2.732626028 | -2.732644032 | -0.000659 |
| **y** | -39.33238823 | -39.33238268 | 0.000014 |
| **z** | -474.3387957 | -474.3383504 | 0.000094 |
| **Port wing leading edge** | | | |
| **x** | 4.284403992 | 4.284413299 | -0.000217 |
| **y** | 51.54082568 | 51.54082635 | -0.000001 |
| **z** | 225.6250534 | 225.6248576 | 0.000087 |
| **Port wing trailing edge** | | | |
| **x** | -2.493376352 | -2.493367148 | 0.000369 |
| **y** | -45.39957722 | -45.39957694 | 0.000001 |
| **z** | -223.9677934 | -223.9679829 | -0.000085 |
| **Starboard wing leading edge** | | | |
| **x** | -1.334661691 | -1.3346617 | -0.000001 |
| **y** | -37.98324233 | -37.98324268 | -0.000001 |
| **z** | 166.0199622 | 166.0198443 | 0.000071 |
| **Starboard wing trailing edge** | | | |
| **x** | 2.624366407 | 2.624366406 | 0 |
| **y** | 33.10281958 | 33.10281967 | 0 |
| **z** | -165.8789381 | -165.8788261 | 0.000068 |
| **Foil reference point** | | | |
| **x** | 3.317346398e-13 | 1.887379142e-08 | -5689324.364 |
| **y** | 4.191691438e-11 | 8.25153279e-07 | -1968444,897 |
| **z** | 3.919087277e-12 | -1.856292897e-08 | 473754,3909 |

## 6.3 Gradients of the yacht model

The gradient of the yacht model represents the influence of the design variables (in meters or degrees) on the $V_{\mathrm{MG}}$ of the yacht (in kts). It is computed with the adjoint method for the bottleneck of the routine (lifting line and parametric model) and finite differences for the remainder of the program. The gradient parts are then combined using the chain rule. To validate these combined gradients, they are compared to gradients computed using the finite difference method for the entire yacht model. Since the accuracy of the finite difference method is depending on the step size, a step size study is performed first. This is shown in Table 6.7, where the gradients of the general design variables are presented for different step sizes. It can be observed that the changes in gradients are significant if the step sizes are large (step sizes $1\text{x}10^{-1}$ to $5\text{x}10^{-4}$). This is due to high truncation errors. Below this, the changes become smaller ($1\text{x}10^{-4}$ to $1\text{x}10^{-6}$). Thereafter, the changes are higher again due to errors introduced through subtractive cancellation. Consequently, the step-size of $1\text{x}10^{-6}$ is believed to produce the most accurate approximation of the gradient and is used for the validation of the combined gradient.

Table 6.7: Full finite difference-based yacht model gradient for different step sizes

| Step size: | Extension [kts/m] | Span [kts/m] | Chord [kts/m] | Taper [kts/m] | Rake [kts/°] |
|---|---|---|---|---|---|
| **1.00E-01** | -2.44615 | -0.0752 | -4.6075 | -1.07129 | -0.00394 |
| **1.00E-02** | -2.48574 | 0.022768 | -4.82463 | -1.14867 | -0.00394 |
| **5.00E-03** | -2.54029 | -0.07769 | -4.66158 | -1.24947 | -0.00394 |
| **1.50E-03** | -2.33944 | -0.07771 | -5.13124 | -1.71891 | -0.00394 |
| **1.00E-03** | -3.3454 | -0.07771 | -5.46673 | -2.05428 | -0.00394 |
| **5.00E-04** | -4.35096 | -0.07771 | -6.4732 | -1.04803 | -0.00394 |
| **1.00E-04** | -2.33869 | -0.07771 | -4.45959 | -1.04803 | -0.00394 |
| **1.00E-05** | -2.33871 | -0.07771 | -4.45959 | -1.04803 | -0.00395 |
| **1.00E-06** | -2.33871 | -0.07771 | -4.45959 | -1.04803 | -0.00401 |
| **1.00E-07** | -2.33873 | -0.07776 | -4.45959 | -1.04802 | -0.00468 |
| **1.00E-08** | -2.33765 | -0.07745 | -4.45966 | -1.048 | -0.01326 |

The combined gradient consists of an adjoint part, which is correct to machine precision, and a finite difference part, whose accuracy again depends on the step size used. Therefore, a second study is performed, where different step sizes are tested for the combined gradient. The results are shown in Table 6.8.

Table 6.8: Combined adjoint and finite difference-based gradient for different step sizes

| Step size: | Extension [kts/m] | Span [kts/m] | Chord [kts/m] | Taper [kts/m] | Rake [kts/°] |
|---|---|---|---|---|---|
| **1.00E+02** | -2.33586 | 0.442568 | -2.57788 | -0.41653 | 0.23192 |
| **1.00E+01** | -2.33905 | -0.07793 | -4.45874 | -1.04782 | -0.00387 |
| **1.00E+00** | -2.33914 | -0.07808 | -4.4607 | -1.04838 | -0.00393 |
| **1.00E-01** | -2.33915 | -0.07809 | -4.46087 | -1.04843 | -0.00394 |
| **1.00E-02** | -2.33915 | -0.07809 | -4.46088 | -1.04843 | -0.00394 |
| **1.00E-03** | -2.33915 | -0.07809 | -4.4609 | -1.04844 | -0.00394 |
| **1.00E-05** | -2.3394 | -0.0772 | -4.45763 | -1.04733 | -0.00353 |
| **1.00E-07** | -2.34075 | -0.06159 | -4.4108 | -1.03098 | 0.003451 |
| **1.00E-10** | -41.2304 | 42.65019 | 157.8579 | 55.95726 | 19.85466 |

It is shown that the optimal step size is $1x10^{-3}$, as the truncation error is minimal, and the subtractive cancellation is not yet affecting the gradient. Therefore, this step size is used subsequently. To validate the resulting combined gradient, it is compared to the full finite difference-based gradient computed earlier. The comparison is shown in Table 6.9. The results show excellent agreement for most variables (0.02% – 0.09% of deviation) and good agreement for some (0.36% – 1.61%). The same comparison has been performed for the yacht model gradient with the full 68 design variables. The large gradient showed excellent agreement with the large finite difference-based gradient with all deviations below 0.08%, most of them significantly lower. Just two variables showed a higher deviation with 3.5% and 3.9%. This further confirms the correct functioning of the combined yacht model gradient computation.

Table 6.9: Combined gradient vs full finite difference-based gradient

| Method | Extension [kts/m] | Span [kts/m] | Chord [kts/m] | Taper [kts/m] | Rake [kts/°] |
|---|---|---|---|---|---|
| **Full FD** | -2.33871 | -0.07771 | -4.45959 | -1.04803 | -0.00401 |
| **AD + FD** | -2.33915 | -0.07809 | -4.4609 | -1.04844 | -0.00394 |
| **Deviation [%]** | 0.02 | 0.49 | 0.03 | 0.04 | -1.61 |

## 6.4 IPOPT with ADOL-C

To verify the correct functioning of IPOPT and the correct functioning of IPOPT in combination with ADOL-C, both methods are used to minimise simple functions with known minima. This process is demonstrated at the example of the “Booth”-function, a common optimisation test function. The Booth-function is shown below. It is two-dimensional and typically evaluated within the given bounds. The minimum of the booth function is $f(1,3) = 0$.

$$f(x,y) = (x + 2y - 7)^2 + (2x + y - 5)^2 \quad for \quad -10 \leq x,y \leq 10$$

Equation 3: Booth-function

The Booth function is optimised with IPOPT and gradients computed using the finite difference method and with IPOPT and gradients computed by ADOL-C. The convergence criterium is set to be $1x10^{-8}$. The maximum iteration limit is 50. The first two runs are started from a random initial point $f(-9.17, -0.91)$ selected by IPOPT. The convergence history of both is shown in Figure 6.2.

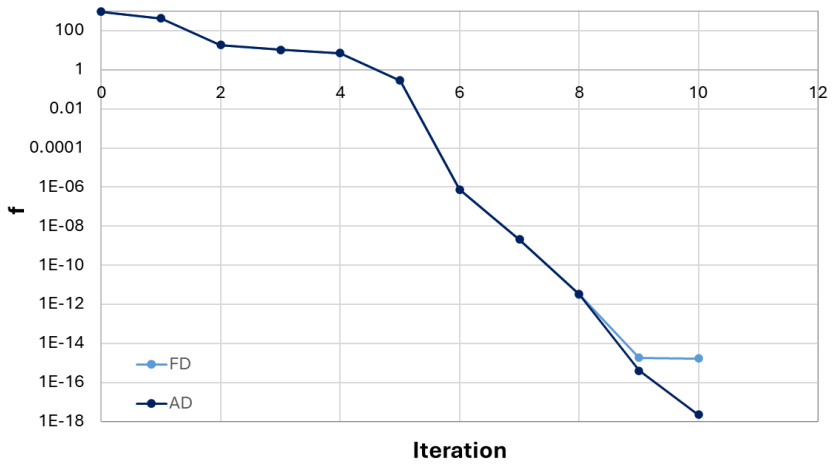



Figure 6.2: Convergence history of Booth-function optimisation with finite-difference-based gradients (FD) and adjoints (AD)

It can be observed that IPOPT identifies the correct optimum within 10 iterations for both methods. This verifies the correct functioning of the optimiser as well as the gradient computation methods. The run which used the adjoint method for gradient computation yields better convergence due to more accurate gradients.

To further verify the optimisation strategy, different starting points are tested. This is shown in Figure 6.3. Again, both methods reach convergence for all starting points, but the higher accuracy of the adjoint method results in better convergence and partly in less required iterations.

This verifies the correct functioning of IPOPT and IPOPT in combination with ADOL-C. This process has been applied to several other test functions with similar outcome.

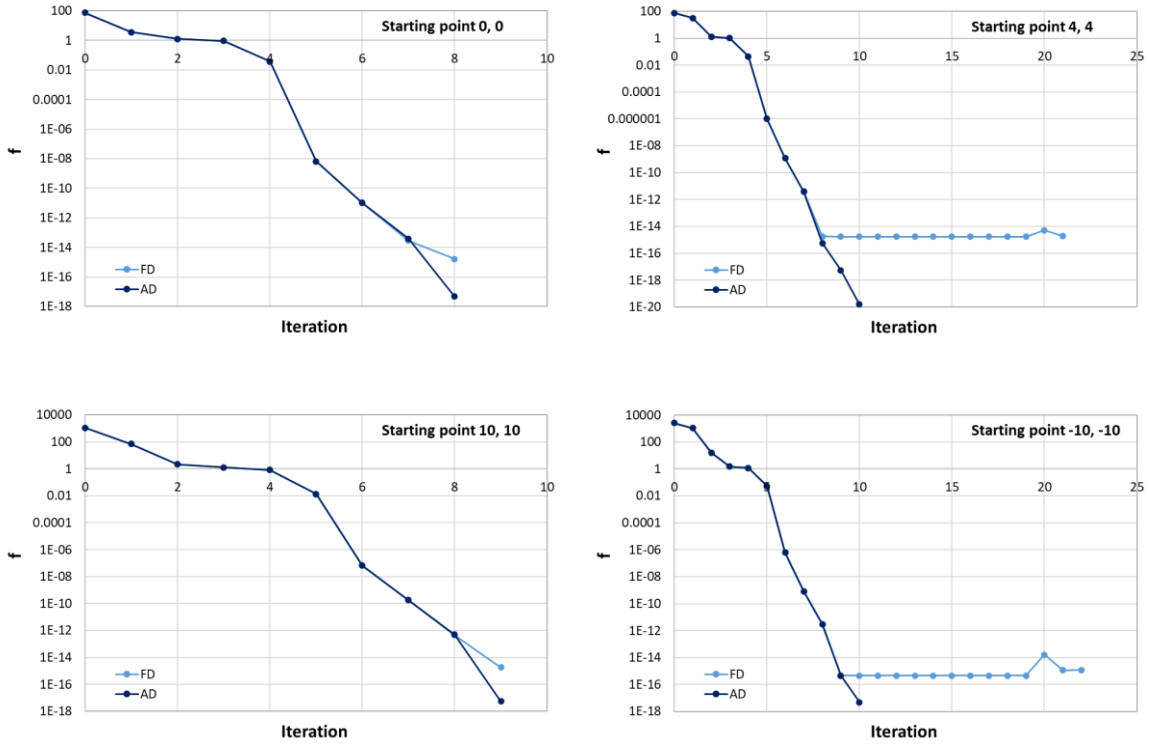



Figure 6.3: Booth-function optimisation with different starting points

## 6.5 General design optimisation

To verify the correct functioning of the entire system-based hydrofoil optimisation routine, the optimisation of the general design variables is benchmarked against a parametric study. This is reported in Paper 2 and shows that the optimiser is able to identify the correct optimum within a few iterations. In addition, the optimisation carried out with IPOPT and the hybrid gradient computation approach is compared to an approach where IPOPT uses full finite difference-based gradients. Both should deliver very similar results as observed in the previous section. The optimisation runs are started from the initial design reported in Paper 2. The convergence criterium is $1\text{x}10^{-4}$. The limit for the maximum number of iterations is 50. The optimisation is performed for a downwind leg ($\beta_t = 140°$ and $V_{\mathrm{TW}} = 11$ kts) with the aim of maximising downwind $V_{\mathrm{S}}$. The resulting convergence histories are presented in Figure 6.4. Both methods identify the correct optimum and converge equally well. This further confirms the correct functioning of the general design optimisation with the hybrid gradient computation.

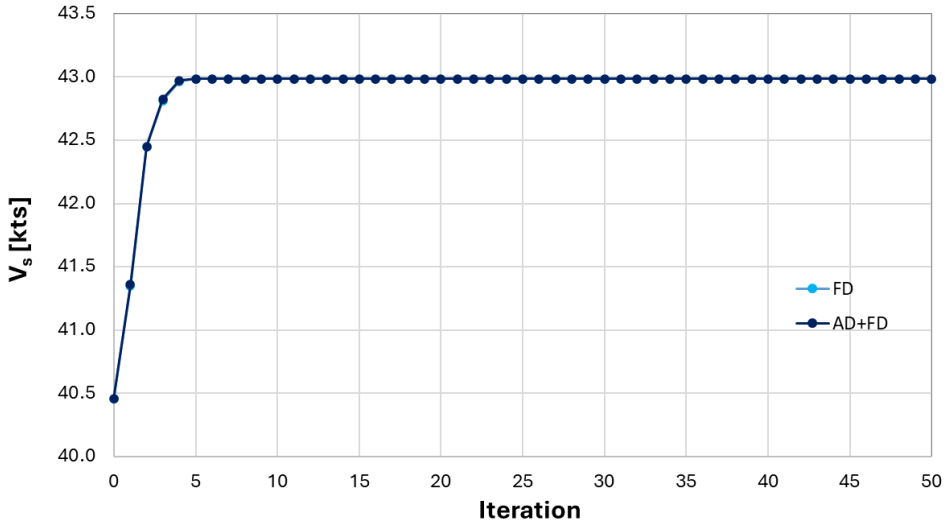



Figure 6.4: Convergence history of the general design optimisation with finite differences (FD) and the hybrid gradient computation (AD+FD)

A further run has been performed without the free-surface image to understand its influence on the optimal design. The run found the exact same optimum as the other runs, but reported slightly higher boat speeds as it did not include the efficiency loss introduced by the proximity of the free surface.

## 6.6 Summary

The developed optimisation framework consists of multiple independent methods. The methods computing the hydrofoil performance and the entire yacht performance have been validated with empirical, semi-empirical and numerical methods and showed good agreement. The routines computing the lifting line- and the yacht model gradient have been validated with finite difference-based gradients and showed excellent and good agreement, respectively. IPOPT and IPOPT in combination with ADOL-C have been tested on common optimisation test functions and showed excellent performance, especially for the combination of IPOPT and ADOL-C. The general design optimisation has been benchmarked against the parametric study in Paper 2 and against a general design optimisation with finite difference-based gradients. All three found the same result. This is the base for trustworthy optimisation results.

# Chapter 7 Conclusions

Many optimisation problems concern the optimisation of a component which is part of a wider system. If the actual aim is maximising system performance and the influence of the component on the system is high, an isolated optimisation is not advisable. It causes inaccuracies, falsifies the optimal solution and leaves potential performance gains unexploitable. Modelling the whole system mitigates these problems but is computationally expensive. A component with a high influence on the system also promises high potential performance gains. This calls for detailed optimisation with a high number of design variables. The computational time required for an optimisation scales with the number of design variables and makes detailed design prohibitively expensive with conventional methods. This is especially the case if the whole system is modelled. The optimal design of a component might also be sensitive to external conditions affecting the state of the system. If a component is to be operated across varying external conditions, this must be accounted for and adds another layer of complexity.

Hydrofoils are the biggest performance differentiators (components) on modern racing yachts (system). Highly optimising them is paramount to staying competitive. The ultimate aim is thereby to maximise the performance of the yacht on the racecourse. The design of the hydrofoils significantly affects the state and the trim of the yacht system. Their optimum design is sensitive to the external conditions such as wind speed and direction. The literature shows, hydrofoil optimisations today are performed in isolation of the yacht system and usually only employ a hand full of design parameters. Furthermore, they are mostly performed for a single condition only, ignoring that the foils are operated across a wide range of conditions in reality. This results in inaccurate optimal designs and leaves potential unexploited.

The aim of this research was to develop a hydrofoil optimisation routine capable of employing a full physics model of the entire yacht in a detailed optimisation routine accounting for the varying conditions. This requires the development of an innovative optimisation strategy that drastically reduces computational time.

The aim was achieved with the development of a stationary physics model of an entire yacht and a detailed parametric model of a hydrofoil integrated into a gradient-based optimisation routine, where the gradient is computed with a combination of the adjoint method and finite differences. The adjoint method is only applied to the bottleneck of the routine where the number of design variables would otherwise influence the run time of the gradient computation. Because of the immense implementation complexity of the adjoint method, it is not used for the remainder of the model, which is instead differentiated using the much simpler finite difference method. This allows the computation of the gradient at small cost and independent of the number of input

variables, while reducing development time. The gradient-parts are combined using the chain rule and are provided to the gradient-based optimisation algorithm IPOPT. This framework enables the optimisation of thousands of design variables in a short time and on a standard PC, hitherto not remotely possible. The framework was benchmarked against a parametric study and known optimum results for a lift distribution and found the expected optimal solutions. Following successful validation, a detailed optimisation with 68 design variables was performed for an upwind condition. An initial discretisation used 40 stations per part and a cosine spacing as recommended by Philips and Snyder (2000) but has led to the development of unrealistic geometries at the junction of the foil. A discretisation with less stations and a linear spacing has solved this problem and was used subsequently. The according run optimised the spineform, the planform and the twist distribution in detail. Bounds were used to ensure the design is within the rule of the yacht class. The run was started from a reasonable design and has shown a trustworthy convergence history. The final foil is perfectly smooth and features the highest anhedral and widest span allowed, with the minimum chord at every station. The routine has designed the twist distribution such that the desired elliptical lift distribution is achieved. This was only possible due to the adjoint method enabling the use of many twist-variables. The final foil has a convex leading edge and is swept aft with the wingtip raked aft even further. This produces induced thrust at the tip of the foil and increases the overall wing efficiency. Identifying this effect has again only been possible through defining the leading edge with a high number of design variables. The final design has led to an increase of 1.58 kts or 6.6 % $V_{\mathrm{MG}}$. This amounts to a 14 s timesaving on a single 1.5 nm upwind leg and is a race-dominating difference. The run solved in roughly 1.5 h on a standard desktop PC. The computation of a gradient was thereby 36 times faster than with a purely finite difference-based method for the 68 design variables. In contrast to any other optimisation approach, increasing the number of design variables would not increase computational cost further. This demonstrates the significant efficiency improvement and enhanced capability of the developed approach.

The framework was extended to multi-condition optimisation to account for the varying wind speeds and courses. The most-likely wind speeds for a specific race venue and time of the year have been determined from historical weather data. This has resulted in three different wind speeds, with a respective up- and downwind course, so six conditions in total. The $V_{\mathrm{MG}}$ achieved in every condition and the according gradient are computed in serial and then averaged at every iteration. This allows the optimisation of the foil for multiple conditions while maintaining a single objective. A multi-condition optimisation was performed starting from the same initial foil as the previous single-condition run. The final foil is perfectly smooth and exhibits the same features as the single-optimisation foil, but with a planform that shifts the centre of lift further aft. This is favorable for the included high-speed cases as it produces more righting moment. The finding has

only been enabled through the use of the full physics model. The multi-condition foil is benchmarked against foils optimised for the six conditions individually. The individual foils perform best in their respective niches whereas the multi-condition foil performs best across the range of conditions. The final multi-condition foil is 1.9 kts or 7.2 % $V_{\mathrm{MG}}$ faster than the initial foil. This amounts to an 81.9 s timesaving over a typical 9 nm race and is again a race-dominating difference. The run solved in roughly 9 h on a standard desktop PC, but computing the gradients in parallel means that this detailed, system-based, multi-condition optimisation can also be performed in about 1.5 h. This in stark contrast to existing strategies that can require several hours for a single iteration on a supercomputer, while only modelling the hydrofoil itself and optimising a small number of design variables. This further proves the enormous efficiency and capability of the developed method.

## 7.1 Novelty

This research has produced several novel contributions to the field of hydrofoil design and optimisation. The first, openly available, systematic investigation into how the design of the America's Cup foils influences the performance of the yacht is presented with the parametric study. It has developed general understanding and serves as a benchmark for current and future optimisations. It has demonstrated the immense potential of hydrofoil design.

For the first time, a physics model of an entire yacht has been differentiated using the adjoint method and finite differences and has been integrated into a gradient-based optimisation routine. This has resulted in a dramatic reduction of computational time allowing a level of optimisation accuracy and design detail hitherto not remotely possible. The hybrid nature of the gradient computation has also resulted in a significant reduction of the development time compared to a full adjoint strategy.

The extension of the framework to multi-condition optimisation marks a further novel contribution. No system-based hydrofoil optimisation has been conducted for multiple conditions. This approach best reflects the reality of an America's Cup race series. The multi-condition optimisation can be performed at no additional cost.

For the first time, the optimal foils for six individual conditions and the optimal foil for six conditions combined have been presented. They showcase the specific optimal solutions and increase understanding of hydrofoil design in general. Furthermore, the optimisation results have shown for the first time that hydrofoils with raked wingtips and aft sweep can produce induced thrust. This increases the overall efficiency of the wing and can be applied to many other hydrofoils. This finding has been made possible by the power of the adjoint method.

## 7.2 Limitations

Several limitations to the developed approach exist. Many of them can be overcome with further work and are presented in the next section. Others are more fundamental and are discussed herein.

**Viscosity –** Including a model of the entire yacht in the optimisation means that the yacht state must be computed numerous times during the optimisation. Solving for a yacht state requires many evaluations of the hydrofoil model itself. Hence, the computational time required to compute the hydrofoil forces cannot be of the order of hours. This rules viscous methods such as RANS CFD out and means that friction drag, flow separation and stall are not directly modelled. The friction drag is accounted for by including viscous drag coefficients, but flow separation and stall are not included. Stall could be modelled to some extend by introducing a non-linear lift-curve-slope to the lifting line method. Nevertheless, viscous methods provide a more accurate prediction of the hydrofoil forces themselves, but they prohibit the use of the whole yacht model and consequently neglect the changes in yacht state during the optimisation. The literature has shown how much the conditions between foil designs can vary. Therefore, using a viscous method, but for a potentially, significantly wrong condition, is expected to cause higher errors, than those introduced by neglecting viscosity. This is especially the case, as no hydrofoils have been observed in the optimisations that are operated close to stall angles.

**Cavitation –** Cavitation is a serious threat for sailing teams in high-speed situations. It can cause a dramatic reduction in lift, accompanied by an increase in drag often resulting in a touch-down of the boat. Modelling cavitation is a challenging task and mostly performed in combination with viscous methods. Cavitation models have occasionally also been linked to potential flow methods, but no such readily available model exists for hydrofoils. This means cavitation cannot be modelled properly, but constraint functions could be used to prohibit designs that cavitate. However, it is unclear to what extend this would influence the design. Furthermore, cavitation is more likely to occur in dynamic scenarios, which are not reflected in the current routine.

**Free-surface effects –** Hydrofoils in the America's Cup are operated close to the free-surface. This causes wave-making with a resulting reduction in lift and an increase in drag. It also causes spray drag and potentially ventilation. The reduction in efficiency due to the presence of the free surface is modelled with the biplane image, the increase in drag due wave making is not modelled. Similarly, the spray drag is not modelled. Empirical models exist that can account for wave making drag and spray drag to some extent but are not yet included. Ventilation is a more serious concern. The current America's Cup has shown several instances of how the ventilation of main foils, rudders and elevators can cause a significant loss of performance. However,

modelling ventilation for sophisticated foil geometries is very complex and exclusively performed in combination with viscous methods. This means ventilation cannot be accounted for directly. Furthermore, it mostly occurs in dynamic scenarios, such as bear-away manoeuvres, that are not captured by the current stationary approach. However, optimal foils could be checked for critical modes of operation with viscous flow methods and these insights could then potentially be used to impose constraints that prohibit designs that ventilate.

**Dynamics –** Sailing a yacht in the America's Cup includes manoeuvres, changes in wind pressure and direction, waves, crew errors and crew latency resulting in a dynamic operation of the yacht. The approach developed in this study uses a stationary model, so it does not capture these dynamic scenarios. However, the boats are operated in a relatively stable and stationary state for most of the time and the differentiation of a dynamic model is, if at all possible, incredibly complex. An associated optimisation would also be significantly more computationally expensive. Nevertheless, the stationary model is not a perfect representation of the yacht sailing in a race. To still consider dynamic effects, an optimal foil can be run in a dynamic VPP to assess its performance in manoeuvres. Should the foil for example not be able to support a full flying tack in low wind speeds, a constraint on the minimum area can be imposed in a following optimisation.

**Implementation complexity –** Modelling the full physics of a sailing yacht is a very complex task, differentiating all of them using the adjoint method is even more so. While the computational efficiency of the adjoint method is the reason many of the physics can be included in the optimisation in the first place, it at the same time hinders and slows down their inclusion. The reason is the associated implementation complexity. Every part of the code that is differentiated using the adjoint method requires to be differentiable, in the form of accessible source code in the right language and must be templated. This means that potentially whole parts of a code must be replaced due to their non-differentiability or due to being part of pre-compiled libraries. Also, templating new models requires significant time and the interfacing with the remainder of the code can necessitate special functions and wrappers. See Appendix A for details. This generally makes the optimisation with the adjoint method much more complex and time-consuming in development than other methods where the physics model can often be treated as a black box. Nevertheless, the adjoint method is the only method that allows truly detailed optimisation.

## 7.3 Further work

The efficiency of the developed approach opens the door for further improvements and additions to the method. This includes the modelling of additional effects to enhance simulation accuracy such as the structural behaviour and the optimisation of further areas such as the trim and the state of the yacht. In the following, the resulting items of further work are presented.

**Hydrofoil immersion and cant optimisation –** The immersion of a hydrofoil has a large influence on the forces it produces and must be accounted for. In the current study, the heel angle, the ride height and the cant angle were fixed. This has resulted in a fixed intersection location of the foil arm and the free surface. The intersection location was determined using a 3D-CAD model. The parametric model was then set-up to just model the underwater part of the foil and arm up to the intersection which does not change. This is valid but prohibits optimising the ride height, heel angle or cant angle during the optimisation. In such a case, the intersection must be computed at every evaluation of the lifting line method. FS-Equilibrium already has an immersion-modelling feature but the underlaying search algorithm cannot be differentiated. This requires finding, implementing and validating an alternative and differentiable method. The cant angle, which is a major variable in hydrofoil operation, can then be included in the optimisation as a hydrofoil design variable with no additional cost. In case of the multi-condition optimisation, six cant variables, one for each course, can be included.

**State variable optimisation –** In the current implementation, the state variables heel angle, pitch angle and ride height are fixed to certain target values. During force balancing, the trim variables are used to achieve these targets. Hence, the optimisation returns the optimal foil geometry for this fixed state. However, another target state could still perform better, for example a state with a lower ride height, which would then also affect the optimal foil design. This can be exploited by including the state variables in the optimisation. The state variables can be directly optimised by the Newton-Raphson method, but that means the state is optimised at every function evaluation. This is likely to be computationally expensive. The preferred option is to use finite differences to compute the influence of each state variable on the $V_{\mathrm{MG}}$ (like for the forces and moments) and then perform the optimisation of state variables alongside the optimisation of the foil. However, both do require the foil immersion feature to be included first, as the state variables change the immersion of the foil. It also requires the sail forces to be provided as a function of the ride height since it affects the end-plating of the main sail. Like the state variables also the course/true wind angle can be optimised to achieve the highest $V_{\mathrm{MG}}$.

**Sectional properties –** Currently, a fixed value for the zero-lift-angle and the drag coefficient is assumed for every section. For force control, these sections are then raked by modifying the angle of attack. In reality however, flaps are used, and the zero-lift-angles and the drag coefficients are not constant. A flap model has been developed as part of this study based on response surfaces. These provide the zero-lift-angle as a function of the flap angle and the span and the drag coefficient as a function of the flap angle, the Reynolds number, the lift coefficient and the span. The underlying data is provided in look up tables. ADOL-C cannot differentiate through look up tables, so the differentiation must be executed externally. The external gradients can then be linked into the ADOL-C gradient computation. The external differentiation can be performed by hand for simple response functions or finite differences and is not a problem, linking the external gradients into the chain is however more complex. Functions have been developed that return the primal values and the derivatives of the response surfaces, but their interfacing with ADOL-C requires further work.

**Separate flap control –** The foils with anhedral angle have two flaps that can be operated independently. However, in the current implementation they are assumed to move simultaneously to produce the required vertical lift. This means the benefit of having two individual flaps is not accounted for. This benefit can be included by using only the more horizontal outboard flap to balance the heave forces. The inboard flap angle can then be treated as a hydrofoil design variable and be optimised for maximum $V_{\mathrm{MG}}$. The multi-condition optimisation can be performed with six individual inboard cant angle variables. Including this does not result in additional computational cost and increases the accuracy of the method. It can be implemented with little effort but should be tested carefully.

**Sail forces –** The forces and moments produced by the sails have a significant effect on the state and trim of the system yacht, in turn affecting the optimal hydrofoil design. The current America's Cup yachts have a sail plan consisting of a conventional jib and a double skin mainsail with internal control mechanisms. These allow deep camber and high lift-coefficients along with a generally higher efficiency when compared to conventional rigs. As no readily available force model for these advanced rigs exists, the sail forces have been modelled with the conventional IMS-module. This does not account for the high possible lift coefficients and the high efficiency. As a result, the performance of the yacht was underpredicted, especially in cases where high lift-coefficients are beneficial. Therefore, the development of a suitable sails model was initiated in the form of a master thesis with the Technische Universität Hamburg-Harbug. In the project a parametric model of the sail plan and a boundary element method were used to predict the sail forces and moments for numerous conditions. The study has also included the influence of the hull on the sail forces. A response surface inside FS-Equilibrium has been fitted to the generated datapoints and returns the six forces and moments produced by the sail set in certain conditions.

The IMS-model can directly be replaced with this new response model, as the sails model is inside the finite difference part of the physics model and does not require the preparation for ADOL-C. It should still be tested carefully and might require improvements/changes.

The following two items of further work are very important aspects to improve modelling accuracy and optimisation capability as well but do require more significant implementation efforts. They both concern the hydrofoil directly, so they must be differentiated with ADOL-C which adds to the complexity. The third item does improve optimisation capabilities and is expected to require serious development work.

**Bulb model –** The bulb is currently modelled with a form factor approach. This does capture the bulb drag to a reasonable level, but it does not capture the influence of the bulb on the lift produced by the hydrofoils. For the lifting line method, the bulb is currently “invisible”. This introduces inaccuracies and finding, implementing and validating a suitable, potential-flow based bulb/fuselage model would be beneficial. The model must then be differentiated with ADOL-C. Since the bulb does not affect the wing geometry in the current implementation, the foil weight can be computed independently of the bulb. The bulb is then automatically sized to meet the weight requirement. However, with a sophisticated model, the wing parts should start at the perimeter of the bulb and not at the junction. Hence, the geometry of the foil is influenced by the bulb size but must also influence the bulb size so that the weight requirement is met. To solve this, the bulb dimensions must be defined as additional design parameters and a constraint function must be introduced that ensures the wing and the bulb meet the weight requirement together.

**Structural model -** Hydrofoils deflect under fluid loading, which changes their shape and hence the forces and moments they produce. This requires a structural model, that coupled to the lifting line method, accounts for the deformation of the foil. A first structural model has been developed as part of this project in the form of a master thesis. The method is based on an Euler-beam model and can compute the bend and twist of the foil as a result of the fluid loading provided by the lifting line method. However, the method required significant tuning with an FEA code. One of the reasons might be that the Euler-beam theory generally neglects shear-deformation. This causes modelling errors for short beams. The discretisation of the foil means that the sections are in fact relatively short. A discretised Timoshenko-beam model does account for this and is the preferred option but is more complex in terms of implementation. Further work is required to develop such a model in C++ and to connect it to the lifting line method. An iterative process is required that computes the foil forces and the resulting changes in shape and feeds the new shape back to the lifting line method to update the forces. This process can be performed until the deformation has converged to a certain extend or a fixed number of iterations is reached. The model must be

written such that it can be differentiated with ADOL-C. The differentiation must only be carried out for the last step of the iteration, as only this step influences the gradient. This saves computational time.

**Section optimisation –** Section design and optimisation is an important part of hydrofoil design. Conventionally, the section optimisation and the 3D-wing optimisation are performed separately. The adjoint method would allow the combination of the two, but this is expected to require significant research and development. It needs a 2D section model to be directly coupled to the lifting line method and differentiated with the adjoint method. It also requires a parametric model of the foil that models the section shape at every station of the foil. The 2D section model must be interfaced with the parametric model and must be able to "read" the section shapes. The 2D section data for this study has been obtained through XFoil. XFoil is the state-of-the-art section analysis tool and its direct integration into FS-Equilibrium and the adjoint optimisation routine would be a significant step forward. However, integrating it in FS-Equilibrium is already complex and differentiating it reversely even more so. Especially, as XFoil is written in Fortran and cannot be differentiated with ADOL-C. In this case, a different tool has to be used that can differentiate Fortran code, for example Tapenade (Hascoet and Pascual, 2013). The gradients can then be linked into the ADOL-C gradient computation using its external functions feature. Alternatively, a C++-based section analysis tool can be found or developed and integrated.

The previous items of further work have concerned the further development of the method. However, the existing method can already be used to investigate and optimise further concepts. Two examples are:

**T-foils –** The current work has assumed a hydrofoil with two flaps and two control systems (although they move simultaneously) and a fixed density reflecting this configuration. In this case, the foil with the highest anhedral angle has performed best, as it has the lowest wetted surface area for the required weight. However, in a T-foil, a single flap can be used with a single control system. This increases the density of the foil and hence also reduces wetted surface area. This effect has first been used by a single team in the last America's Cup, but now seems to be the preferred option of all teams. Nonetheless, this option to reduce wetted surface area is not currently reflected in the optimisation assuming constant density regardless of the anhedral angle. However, it can be addressed in a separate optimisation run with no further development required. In this separate run, a higher density must be assumed, and the spine form must be fixed to a T-Foil. The optimum T-Foil can then be compared to the optimum Y-Foil.

**Winglets -** The current class rule in the current America's Cup defines a box the foil is not allowed to exceed. The box is rectangular with additional "pockets" for winglets. In the current set-up of the optimisation these pockets are not considered, and the tip of the foil is fixed at the bottom of the box prohibiting the forming of potential winglets. This has enabled comparison with the parametric study and has simplified the validation of the optimal designs. However, exploring the effect of winglets and optimising their design is of great interest. For this reason, the method should be extended to optimise foils with winglets. This requires either a change to the parametric model and the introduction of new bounds or the use of the existing parametric model and constraint functions that represent the more complex shape of the foil box.

The optimal foils identified in this study have opened another area of potential research:

**Investigating sweep and wingtip rake –** The optimal foils identified in this study have exhibited aft sweep with an even further aft rake of the wingtip. This feature has resulted in induced thrust at the wingtips and increases the overall efficiency of the wing when compared to a straight foil. This effect could be used to increase the efficiency of many hydrofoils and should be further investigated and understood. This can be achieved with specific analysis and optimisation using the lifting line method and validation with higher fidelity methods.

## 7.4 Other applications

The optimisations in this thesis have been performed for an America's Cup yacht, but the developed approach can also be used to optimise the foils of other sailing yachts such as International Moths, IMOCAS or Ultims. This only requires the adaption of the force models to the new yacht and the modification of the parametric model to represent the according foil type. Similarly, the method can be used to optimise the hydrofoils of the increasing number of foil-assisted power boats to enhance their efficiency and contribute to decarbonisation.

In addition, the developed methodology is not limited to hydrofoil optimisation but can also be directly used for other maritime applications. Wind-assisted ships for example are conventional motor ships with the addition of sails or wings supporting the engines with the aim to reduce fuel consumption. Several such wings can be placed on the deck of a ship. Every wing has numerous trim parameters that allow to manipulate its shape for maximum performance. Hence, optimising the trim of a ship with multiple wings requires a large number of design variables and modelling the whole ship system is important. The developed framework can directly be used for this trim optimisation. The lifting line method can predict the wing forces and only requires a modification of the parametric model to represent the wings instead of the foil. Suitable force models for the other components of the ship, such as the hull, must be included.

The general methodology can also be applied to other problems of similar class outside of the maritime field. This class of problems is defined by having a component within a system in an equilibrium state. The component must have a high influence on the system, otherwise isolated optimisation is to be preferred. The component model must be computationally cheap, as it is evaluated many times for computing the system state in every iteration. An obvious example are the wings of an airplane.

# Appendix A Algorithmic differentiation of the lifting line method and the parametric model

The key to the immense efficiency of the developed approach is the adjoint method. The adjoint method is the reverse differentiation of a function. This reverse differentiation can be performed by hand or with the help of algorithmic/automatic differentiation tools. However, algorithmic/automatic differentiation is by no means purely automatic and requires significant implementation efforts. These are discussed in this appendix which is meant for the reader who wishes to acquire a deeper understanding of the implementation process. The adjoint method is applied to the lifting line method and the parametric model, so their original implementation is presented first. Thereafter, the required changes for differentiation are discussed.

## A.1 Lifting line implementation

In this study, FS-Equilibrium's lifting line method is used to compute the forces and moments produced by the hydrofoil. Its implementation is based on the advanced lifting line approach developed by Phillips and Snyder (2000). It can be set up in the graphical user interface (GUI) of FS-Equilibrium as shown in Figure 7.1. The tab allows to specify the type of fluid, the solver, the number of individual parts the appendage system shall have and others. For the purposes of this study, three foil parts are defined, the port wing, the starboard wing, and the extension of the foil arm. The properties of these individual foil parts can be assigned in further tabs that are available on the left in Figure 7.1 (Foil0, Foil1 and Foil2). The free-surface effects can be modelled using the biplane- and the inverted foil analogy. This can be specified in the "Mirror" tab.

The geometry of a foil part is provided as an IGES file and must be a "ruled surface" representing the planform and the spineform of the part. The surfaces are assigned to the foil parts in the foil-tabs shown in Figure 7.2 at the top. In these tabs the user can also assign the number of vortices used to discretise the foil part (NumSections), the sectional properties along the span of the foil part and other relevant metrics.

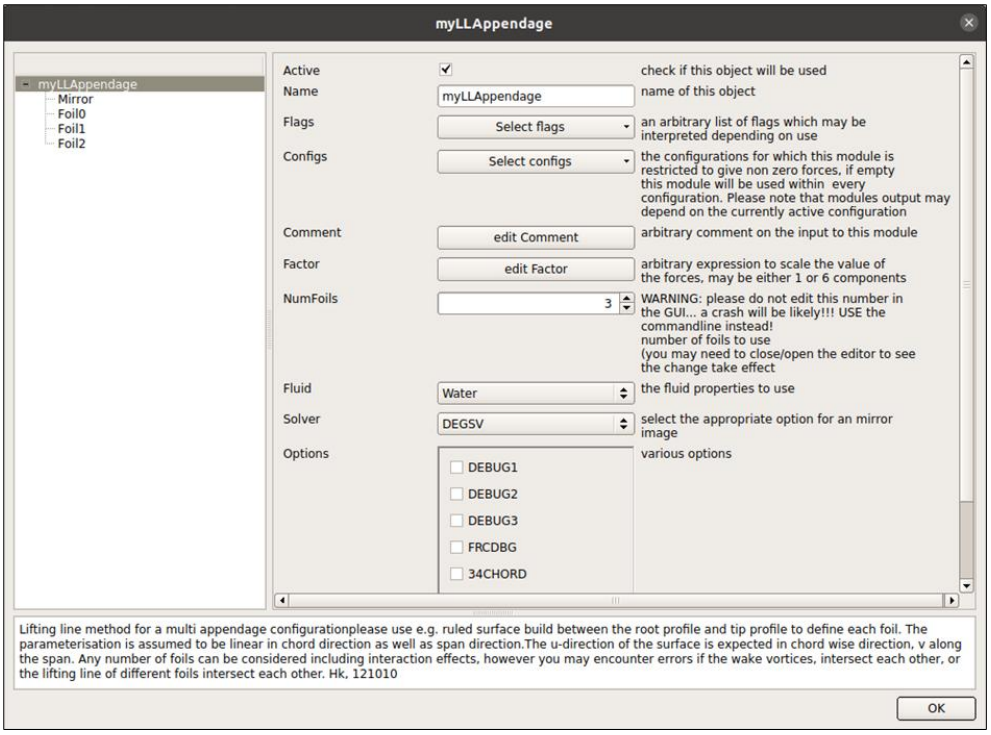



Figure 7.1: Lifting line module tab in FS-Equilibrium's GUI

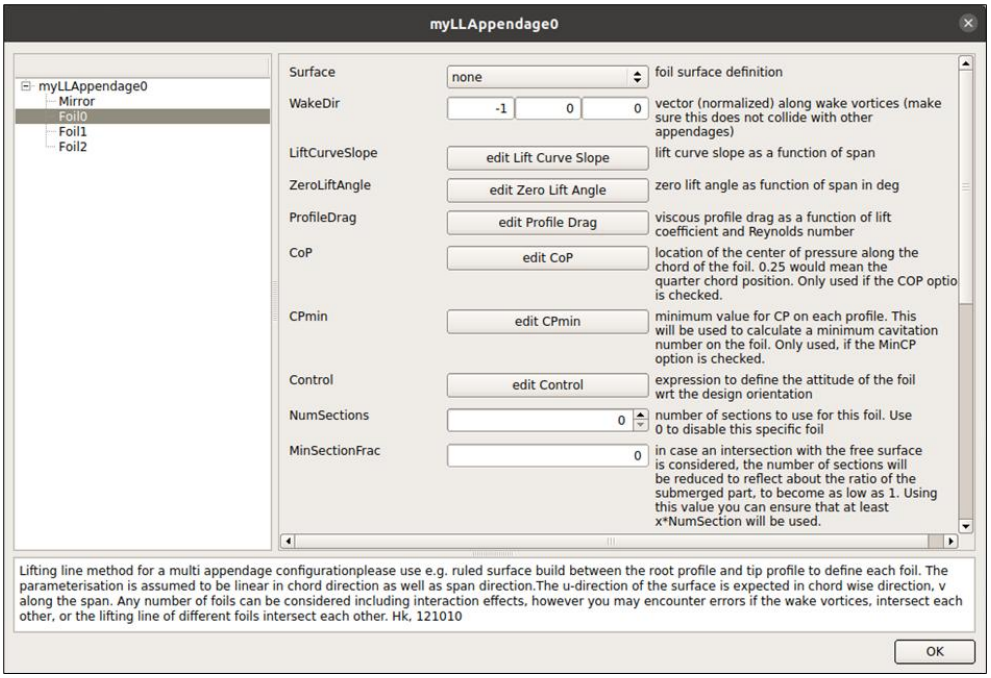



Figure 7.2: Individual foil part tab in FS-Equilibrium's lifting line method

For the differentiation of the lifting line method, the underlaying implementation is important. The general set-up is presented in Figure 7.3, where any red font colour can be ignored for now. The whole implementation is executed within a class called *FLLAppendage*. Inside *FLLAppendage* two structures exist, *TAppendage* and *TGrid*. *TAppendage* stores information specific to the foil parts. This includes the number of stations used and the sectional properties. *TGrid* stores the geometric information of the whole appendage system, e.g., the chord at every station and the

location of the vortices (variables *chord* and *vtxPts*). Upon initialisation three instances of *TAppendage* are created for the three foil parts and stored in *mAppendage*. A single instance of *TGrid* is generated and saved as *mpGrid*. Inside the structures the member variables and methods are shown, for example *getChord()* in *TAppendage*, computing the chord at a given span location. *FLLAppendage* also includes its own methods for example *solve()* which solves the lifting line for a given condition. *solve()* is called by the VPP solver at every step of the force-balancing routine via the function *computeForcesA()*.

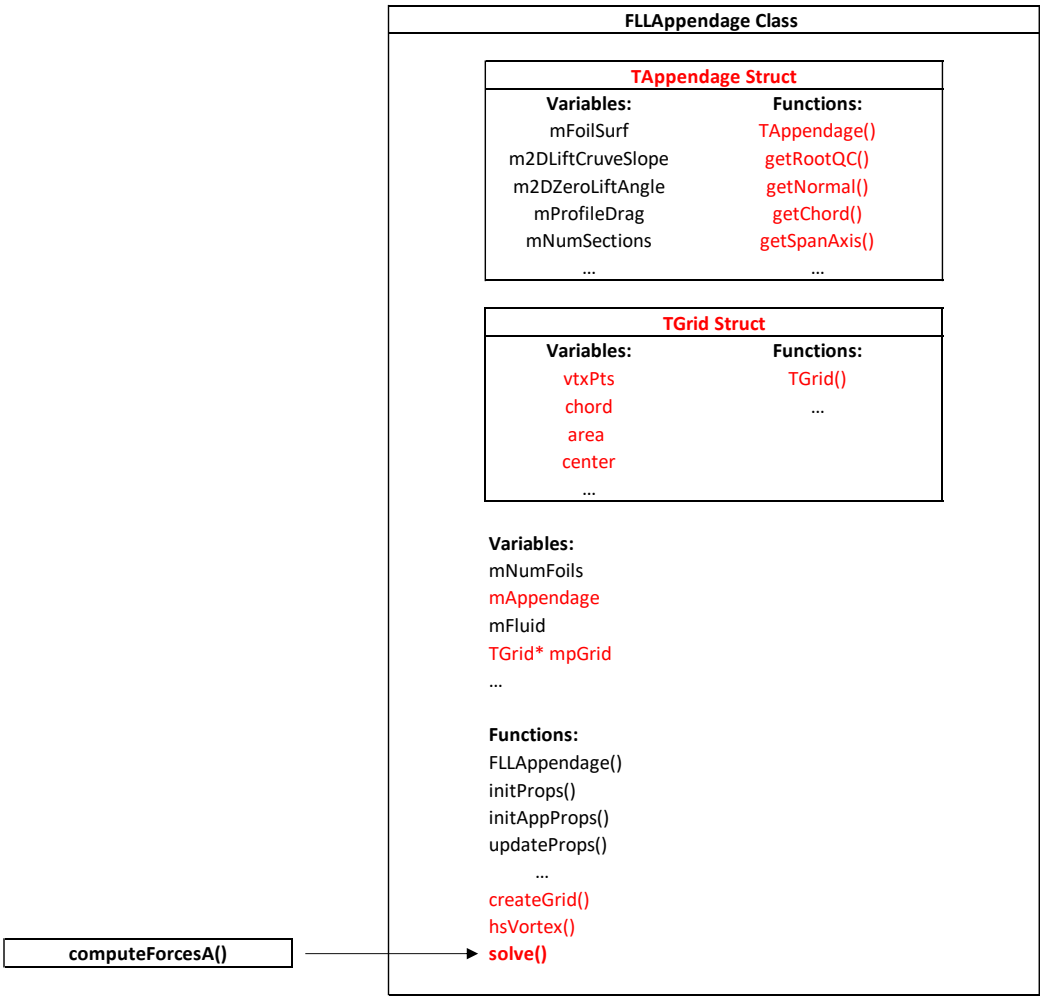



Figure 7.3: Architecture of the *FLLAppendage* module showing the incorporated structures, functions and variables

The workflow of the lifting line method is presented in Figure 7.4. The VPP solver (Newton-Raphson) calls *computeForcesA()* for the current condition, which calls *solve()* and starts the solving process of the lifting line method. First, the foil surfaces are loaded and transformed according to the current cant angle (Steps 1 -3). Thereafter, the grid is generated for the current condition (Step 4) by calling the function *createGrid()*. *createGrid()* obtains the information about the foil parts from the *TAppendage* objects and executes steps 4.1 to 4.5 to compute the grid and the lifting line. For example, this includes the coordinates of the vortices and the chord at every

station. This information is stored on *mpGrid*. In steps 5 to 6, the influence matrix is set up based on *mpGrid*. It is solved for the vortex strength distribution using a Gaussian Elimination approach. Subsequently, in step 10, the force per station is computed based on the density, the local velocity and the local chord. Finally, the force distribution is integrated to obtain the overall forces and moments (Step 11). The forces and moments are transformed into the global coordinate system (Step 12) and returned to the Newton-Raphson.

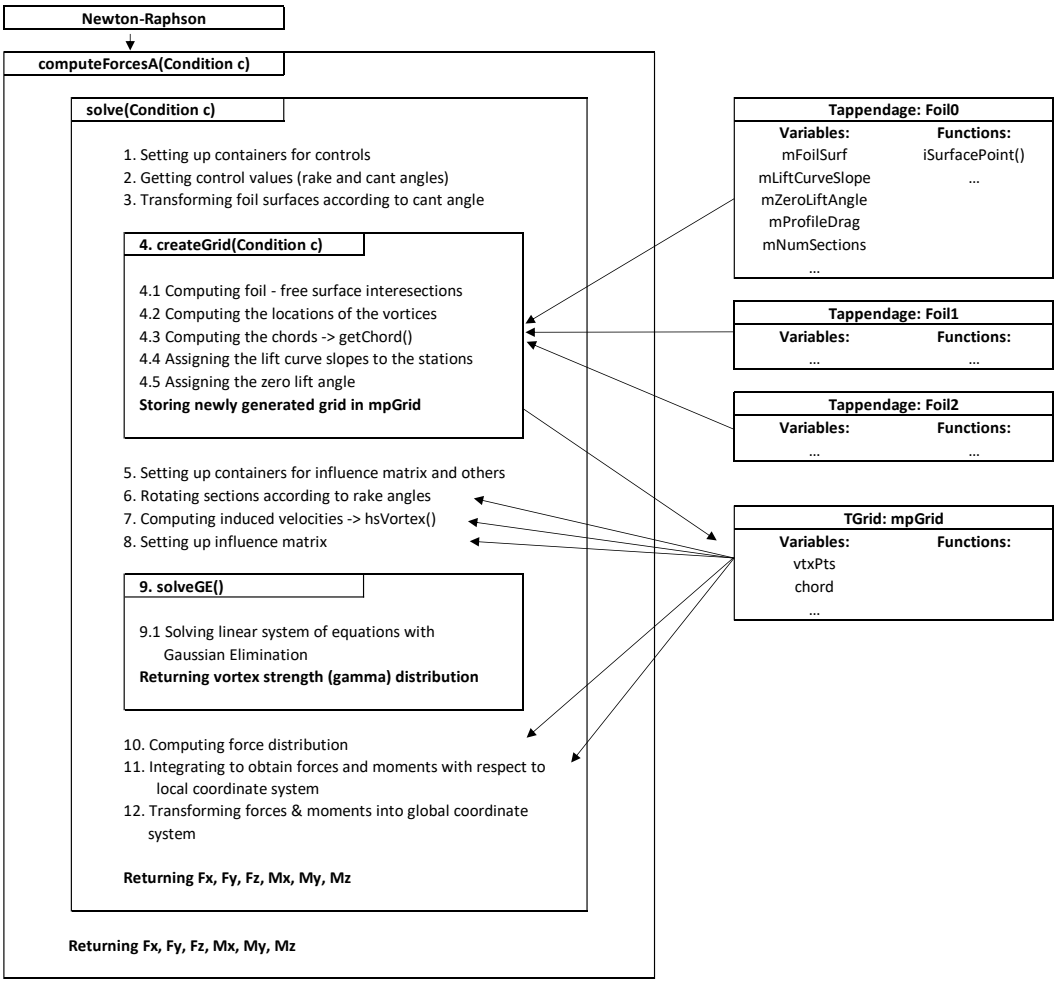



Figure 7.4: Flow of the lifting line's solve function and its calls to the structures of *FLLAppendage*

## A.2 Parametric model implementation

The parametric model of the hydrofoil serves to generate the shape of a hydrofoil from a set of design variables. FS-Equilibrium's lifting line module requires ruled surfaces to be provided in the form of NURBS surfaces. From these surfaces the method can derive the geometrical data such as the span, the chord along the span, the twist along the span, the sweep, and the anhedral angle of the foils. NURBS surfaces (Figure 7.5) are defined by control points ($P_{n,m}$) and the degrees of the surface that is fit between the points in the *u*- and *v*-direction. A knot vector defines if the surface is clamped at its endpoints and weights can be applied to the control points to increase or decrease their influence on the curve. For further reference see Piegl and Tiller (1997).

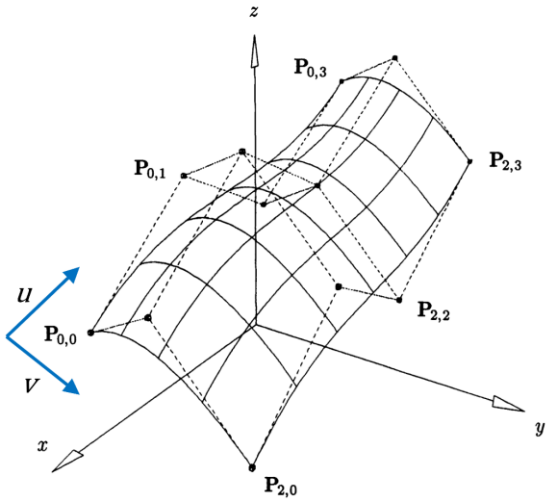



Figure 7.5: NURBS surface set up by control points ($\boldsymbol{P}_{i,j}$) derived from Piegl and Tiller (1997). The surface can be evaluated at the point u, v to return the coordinates of this point on the surface

The parametric model of the AC75 hydrofoil is developed based on NURBS surfaces. It is used to define the three foil parts which are the foil arm extension and the port- and starboard wing. The first version of the simple parametric model is shown in Figure 7.6. The surface of the foil arm is fixed by the AC75 rule and not submerged at any time, so it is only included for visualisation purposes. Just the two points that it shares with the extension are required. The extension of the foil arm is defined with two additional points at the leading and the trailing edge where the extension and the wings meet (*mExtensionLead(x,y,z)* and *mExtensionTrail(x,y,z)*). Two further points are used to define the wings representing the leading- and trailing edge points at the tip of the foil (*mPortWingLead(x,y,z)*, *mPortWingTrail(x,y,z)* and *mStbWingLead(x,y,z)*, *mStbWingTrail(x,y,z)*). The root is already defined by the location of the extension. This results in six design points. A seventh point (*mDelta*) is used as a reference point that determines the position of the foil with respect to the transom of the AC75 at the waterline. All other points are

measured from this reference point. These seven points each have three components which result in a total of 21 design variables.

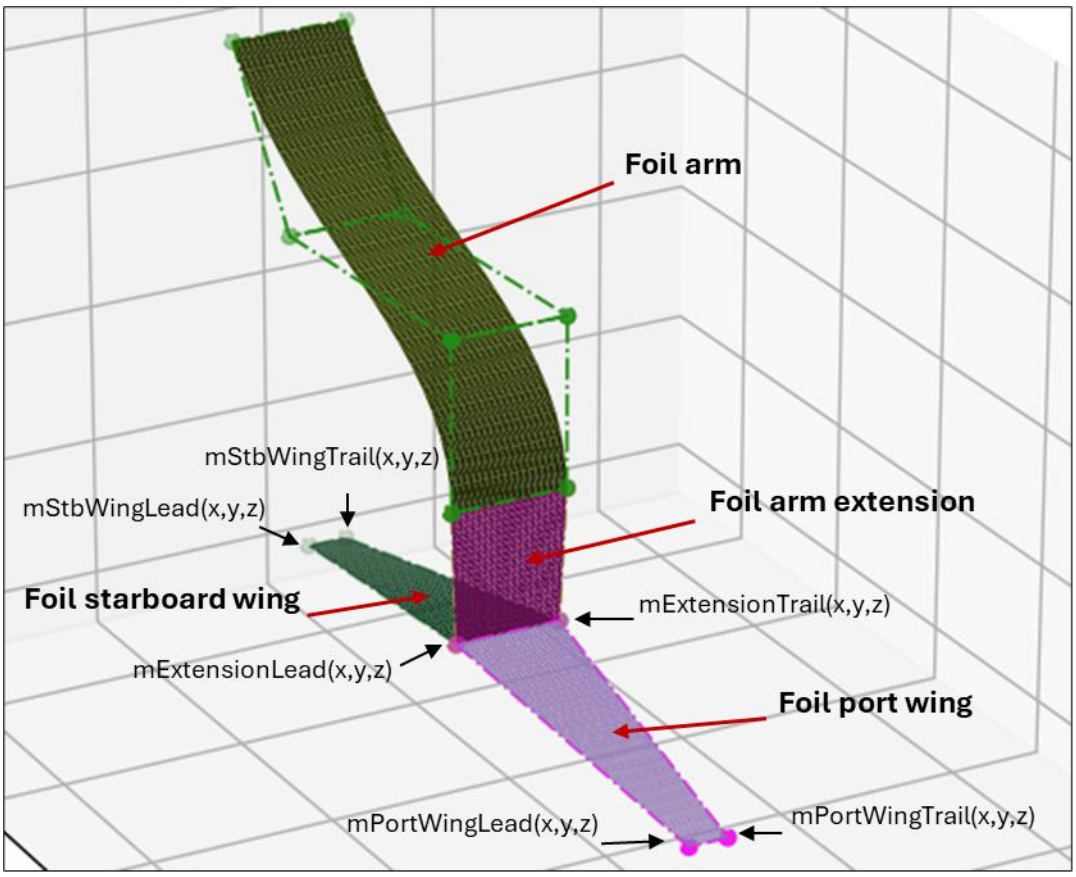



Figure 7.6: Simple parametric model of the AC75 hydrofoil showing the control points employed to define the hydrofoil geometry; visualised with the NURBS-Python visualisation feature (Bingol and Krishnamurthy, 2019)

The implementation of the model is based on the C++ NURBS library TinyNURBS (Jayaraman). The library is open source and header-only, which allows access to the full source code. The library is incorporated in a new structure called *MySurface* as shown in Figure 7.7. The structure stores information for a single surface and means to evaluate this surface. As stated before, a NUBRS surface is defined with the degrees, the knot vectors, the control points and the weights. However, the modelling capabilities of the model are deemed sufficient without including the weighting feature, so the weights were not considered. This is equal to setting the value of every weight to one and essentially reduces the NURBS surface to a B-Spline surface. The degrees, knot vectors and control points are stored in *MySurface* using the according variables. Based on this information the surface can be evaluated using the function *isurfacePoint()* which returns the coordinates of the surface at the location *u*, *v*. *isurfacePoint()* requires *ibsplineBasis()* and *ifindSpan()* to compute the coordinates.

```
struct MySurface
{
    unsigned int degree_u, degree_v;
    std::vector<double> knots_u, knots_v;
    std::vector<std::vector<std::vector<double>>> control_points;


    mySurface() = default;


    std::vector<double> ibsplineBasis(double u);

    double ifindSpan(double u);

    std::vector<double> isurfacePoint(double u, double v);
};
```

Figure 7.7: *MySurface* structure for a single NURBS surface, storing all required information to define the surface and functions to evaluate it

The parametric model consists of three *MySurface* objects which are called *extension*, *portWing* and *stbWing*. A surface can be evaluated for example with *extension.iSurfacePoint(u, v)* at the location on the surface u, v. The three surfaces are integrated into a single method called *getMyPoint()* that represents the whole parametric model. The function is a method of *TAppendage* and is shown in Figure 7.8.

```
FPoint3 FLLAppendage::TAppendage::getMyPoint(TConfiguration points, double u, double v) const
{
    FPoint3 pt_o;

    double cant_deg = mpParentAppendage->mCant;
    double a = cant_deg * (PI / 180);


    double dx = points.mDelta[0];
    double dy = points.mDelta[1];
    double dz = points.mDelta[2];

    // Translating all points by dx, dy, dz (measurement from AC75 transom to reference point)
    // Rotating all points by a (cant angle in radians)

    initialiseSurfaces(TConfiguration points);

    int idx=getIndex();

    switch (mpParentAppendage->mGeoIdentifier)
    {
        case 0:     // extension model
            pt_o = extension.isurfacePoint(u, v);
        break;
        case 1:     // port wing model
            pt_o = portWing.isurfacePoint(u, v);
        break;
        case 2:     // stb wing model
            pt_o = stbWing.isurfacePoint(u, v);
        break;
    }

    return pt_o;
}
```

Figure 7.8: getMyPoint() function that incorporates the parametric model

The arguments of the method include the design points presented in Figure 7.6 (*TConfiguration points)* and the location on the surface to be evaluated (*u, v*). The function translates and rotates the points to their correct position (based on *mDelta* and the cant angle) and initialises the three surfaces. A switch-statement is used so the foil surface currently of interest is evaluated. The method returns the coordinates of the evaluated point in 3D space.

## A.3 Differentiation with ADOL-C

The differentiation of a function with ADOL-C requires several general steps. These ensure that ADOL-C recognises the input and output variables and that it obtains all relevant information about the function to compute its gradient. To obtain the function information, ADOL-C uses a specific datatype called *adouble* (active *double*). This datatype allows to store the value of a variable, and the information required to compute the derivatives. It is used to "tape" the function while it is evaluated. This "tape" is an internal representation of the function and is differentiated later. Hence, every variable that influences the gradient must be an *adouble*. The datatype is defined in the ADOL-C header along with the ADOL-C functions and the overloaded operators. The full set of steps required is presented below.

1. Include the ADOL-C header
2. Specify the number of input and output variables
3. Define the input and output variables in the normal *double* datatype
4. Also define the input and output variables separately in the a*double* datatype
5. Define all intermediate variables that influence the gradient as *adoubles*

6. Start the gradient computation with the function *trace_on()*
7. Initialise the input variables (*double* to *adouble,* <<= *operator*)
8. Evaluate the function to differentiate
9. Deinitialise the output variables (*adouble* to *double,* >>= *operator*)
10. Stop the gradient computation with *trace_off()*

11. Declare containers that store the gradient or jacobian
12. Compute the gradient with *gradient()* or the jacobian with *jacobian()*

An example of how these steps are applied to a simple function is given in Figure 7.9. For a more in-depth introduction see Walther and Griewank (2012).

```cpp
#include <iostream>

#include "adolc/adolc.h"        // Including the ADOL-C header



int main()
{
    int n = 2;                  // Specifying the number of input variables
    int m = 1;                  // Specifying the number of output variables
    int tag = 1;                // Specifying the tape the trace should be recorded on


    // Defining the input and output variables
    double x1_in = 1.5;
    double x2_in = 0.5;
    double y_out;

    // Defining the corresponding adoubles
    adouble x1;
    adouble x2;
    adouble y;


    //-------------------------------------------------------------------------------------
    trace_on(tag);              // Starting the tape

    x1 <<= x1_in;               // Initialising the active input variables
    x2 <<= x2_in;


    y = (sin(x1/x2) + (x1/x2) - exp(x2)) * ((x1/x2) - exp(x2)); // Equation 3


    y >>= y_out;                // Deinitialising the active output variables

    trace_off();                // Stopping the tape
    //-------------------------------------------------------------------------------------


    // Declaring containers to store inputs and outputs
    double* x = new double[n];
    x[0] = x1_in;
    x[1] = x2_in;
    double* g = new double[n];


    gradient(tag, n, x, g);     // Computing the gradient

    std::cout << "Gradient: " << g[0] << ", " << g[1] << std::endl;  // Printing to console


    return 0;
}
```

Figure 7.9: Steps 1 – 12 applied to a simple function for gradient computation

## A.4 Differentiation of the lifting line method and the parametric model

Figure 7.9 has shown the steps required to differentiate a function with ADOL-C. What is straight forward for a simple function is much more complex for sophisticated programs and requires additional development. The lifting line method is integrated into FS-Equilibrium and must communicate with other parts of the program and the GUI. Furthermore, it draws on numerous libraries. This must be accounted for. Additionally, the existing method includes sections of the code that cannot be differentiated. These need to be replaced or excluded.

### A.4.1 Replacing non-differentiable features

The existing lifting line method includes two parts or features that cannot be differentiated for different reasons. This must be resolved in a first step. The features must be either replaced, differentiated externally, if possible, or excluded entirely.

The first instance is the feature that computes the intersection of the free-surface and the hydrofoil. This is used to model hydrofoil immersion. The intersection is computed with Brent's method which is non-differentiable by nature. Therefore, the feature cannot be used and is excluded. However, differentiable approaches may exist that could be used in the future.

The sectional properties of the hydrofoil are provided by response surfaces that are created based on look-up tables upon initialisation. Since the sectional properties influence the hydrofoil performance and are depending on the hydrofoil design, they affect the gradient and must be included in the differentiation. However, ADOL-C cannot differentiate look-up tables, so the response surfaces have been replaced with constant values for the lift-curve-slope, the zero-lift-angle and the profile drag coefficient. As discussed before, an external differentiation of the response surfaces would allow their inclusion, but this requires further work.

### A.4.2 Replacing pre-compiled libraries

For a computer code to be differentiated with ADOL-C, it must be in the form of source code. Compiled code cannot be differentiated. Hence, pre-compiled parts or libraries must be replaced or excluded. The lifting line method uses NURBS-Surfaces to represent the geometry of the hydrofoil. These and the means to evaluate them are defined in a pre-compiled library. Since the geometry definition and evaluation is vital, the original library must be replaced. Therefore, the developed parametric model is directly and internally coupled to the lifting line method, replacing the original library. This has been an integral and substantial part of the differentiation work.

The pre-compiled library provides the function *getPoint(u,v)* returning the coordinates of the resulting point. This function is used to extract any relevant information from the surfaces, for example the position of the quarter chord at the root of the foil (Figure 7.10). This function is replaced with the function *getMyPoint(u,v)* which represents the parametric model (Figure 7.8). It is used in the same manner as the original function, see for example Figure 7.11.

```
FPoint3 FLLAppendage::TAppendage::getRootQC() const
{

    return mFoilSurf.getPoint(0.25,0.);

}
```

Figure 7.10: Original definition of the function *getRootQC()*

```
FPoint3 FLLAppendage_int2::TAppendage::getRootQC(TConfiguration points) const
{

    return getMyPoint(points, 0.25, 0.);

}
```

Figure 7.11: Updated definition of the function *getRootQC()*

getMyPoint() is used in several other simple functions computing for example the chord at a given station (*getChord()*), but is also required in more complex functions. One example is a function computing the direction of the span (*getSpanAxis()*). This has again used a pre-compiled function differentiating the surface using finite differences (*dV()*). This is shown in Figure 7.12 and required a new implementation of the surface differentiation feature (Figure 7.13). Numerous such changes had to be made.

```
FVector3 FLLAppendage::TAppendage::getSpanAxis(double t) const
{

    GEM::Magnet mag(&mFoilSurf,0.25,t);
    FVector3 span=mFoilSurf.dV(mag);
    span.normalize();
    return span;

}
```

Figure 7.12: Original definition of *getSpanAxis()*

```
FVector3 FLLAppendage_int2::TAppendage::getSpanAxis(TConfiguration points, double t) const
{
    FVector3 init = getMyPoint(points, 0.25, t);
    FVector3 finite;
    double delta = 1e-4;
    double x,y,z;

    if((t <= (1.0 - delta)) && (t >= 0.0))
    {
        finite = getMyPoint(points, 0.25, t + delta);
        x = (finite[0] - init[0]) / delta;
        y = (finite[1] - init[1]) / delta;
        z = (finite[2] - init[2]) / delta;
    }
    else if((t <= (1.0)) && (t >= 0.0 + delta))
    {
        finite = getMyPoint(points, 0.25, t - delta);
        x = (init[0] - finite[0]) / delta;
        y = (init[1] - finite[1]) / delta;
        z = (init[2] - finite[2]) / delta;
    }
    else
    {
        x = 0.0;
        y = 0.0;
        z = 0.0;
    }

    FVector3 span = {x, y, z};

    span.normalize();

    return span;
}
```

Figure 7.13: Updated definition of *getSpanAxis()*

In the original implementation, the surfaces were defined in IGES-files containing the points spanning them. In the new implementation the points must be stored elsewhere. This is done in an instance of a new structure called *TConfiguration points*. A new tab was created that allows the manipulation of the points in the GUI.

### A.4.3 Different options for turning *doubles* into *adoubles*

With all non-differentiable parts of the code replaced or excluded, the differentiation process can be started. This requires the "taping" of the whole function. This "tape" is an internal representation of the function which is then differentiated in a second step. The taping process requires any variable that influences the gradient to be recorded. This is achieved with the *adouble* datatype. Hence, all variables that influence the gradient must be declared as *adoubles*. These are the input variables (design variables) and output variables (forces and moments), but also most intermediate variables, functions and classes. This includes base classes performing for example point, vector and matrix operations and the main method.

Three options exist for turning *doubles* into *adoubles*. The simplest one is to replace any *double* with an *adouble* by hand or with the command *#define double adouble*. This is shown in Figure 7.14 for a function in the vector library computing the dot product. The command automatically redefines every *double* type to an *adouble* in a given file. This approach is simple and fast but means the lifting line method can only be run in *adoubles* from there on. However, the lifting line

method must still run in *doubles* for normal VPP solving. Therefore, an approach must be taken that allows the lifting line method to be run with different data types. This can either be achieved by overloading or by templating functions.

```
#define double adouble

double FVector3::dotProduct(const FVector3 & v1, const FVector3 & v2)
{
    double p = 0.;
    for (int i = 0; i < 3; i++)
    {
        p += (v1.x[i] * v2.x[i]);
    }

    return p;
}
```

Figure 7.14: Redefining to *adoubles* for the dot product function in FS-Equilibrium's vector class

Overloading allows to define a single function several times with different datatypes. The function must therefore be declared and defined twice as shown in Figure 7.15. While functions can be overloaded, this is not the case for variables, structures and classes. The lifting line method contains several member variables of classes and structures that must be defined in *doubles* and *adoubles*. As neither the variables itself can be overloaded, nor the overlying structures or classes, overloading cannot be used for the given problem. A further downside of overloading is that the whole program must be developed and maintained twice which promotes bugs and increases development time. Templating instead can be applied to structures and classes as well and does not require the implementation of multiple definitions.

```
double FVector3::dotProduct(const FVector3 & v1, const FVector3 & v2)
{
    double p = 0.;
    for (int i = 0; i < 3; i++)
    {
        p += (v1.x[i] * v2.x[i]);
    }

    return p;
}


adouble FVector3::dotProduct(const FVector3 & v1, const FVector3 & v2)
{
    adouble p = 0.;
    for (int i = 0; i < 3; i++)
    {
        p += (v1.x[i] * v2.x[i]);
    }

    return p;
}
```

Figure 7.15: Overloading the dot product function for use with *doubles* and *adoubles*

Templating allows the creation of generic structures, classes and functions that can store and solve in different data types. A function is turned into a templated function by adding *template<typename T>* to it. The variables inside the function that must be generic are then defined with the data type *T* instead of *int*, *double* or *adouble*. This can also be applied to the

return type and the arguments of functions and variables inside structures and classes. With templating being the only possible option, it is applied to the whole method. This is explained in detail in the next sections. Nevertheless, templating requires significant implementation time. Every relevant function or class must be altered to accommodate the generic data type *T* and whenever an object of a templated class is generated or a member function is called, the datatype must be specified. Figure 7.16 shows the templated version of *dotProduct* and how it is run in *doubles*.

```
template<typename T>
T FVector3::dotProduct(const FVector3<T> & v1, const FVector3<T> & v2)
{
    T p = 0.;
    for (int i = 0; i < 3; i++)
    {
        p += (v1.x[i] * v2.x[i]);
    }

    return p;
}


FVector3<double> vectorOne = {1., 1., 1.};
FVector3<double> vectorTwo = {2., 2., 2.};
double dotProduct = FVector3<double>::dotProduct(vectorOne, vectorTwo);
```

Figure 7.16: Templating and solving of the dot product function

### A.4.4 Templating the base classes

The first step in templating the lifting line method is templating the base classes used by the method. FS-Equilibrium has several such classes that are employed everywhere in the program. The classes relevant for the lifting line method are *FVector3*, *FPoint3*, *FMatrix4* and *FMotor*. *FVector3* is the class for vectors with a size of three. It includes functions to compute the dot product (Figure 7.14), the cross product, the length of the vector and others. *FPoint3* is used to store coordinates or points with their x, y, and z-components. It is a child-class of *FVector3* and shares all the features, while additionally providing functions to compute the distance between two points or the distance of a point to the origin. *FMatrix4* is the class for 4x4 matrices containing means to compute for example transformation matrices, projection matrices and the inverse of a matrix. *FMotor* is used to store the three force components $f_x$, $f_y$, $f_z$ and the three moment components $m_x$, $m_y$, $m_z$. Additionally, all classes have getter-, setter-, constructor-, and destructor functions and numerous operators.

In a first step, the classes are copied and renamed to *TFVector3*, *TFPoint3*, *TFMatrix4* and *TFMotor* to separate them from the original base classes. Although the original classes could be run in doubles after templating, the templating of these original classes would mean that the data type *double* would have to be specified everywhere in the whole VPP. By copying and renaming the classes, and only using those new classes in the lifting line method, this is not necessary.

However, it means that wrappers must be used whenever the lifting line method is communicating with other parts of the VPP. This is explained in more detail in the next section. Figure 7.17 shows an example of a templated function from the *TFMatrix4* class that includes *TFVector3s* and functions like *crossProduct()*. This process has been applied to hundreds of functions in the base classes. Additionally, all base classes had to be turned into header-only libraries.

```
template<typename T>
TFMatrix4<T> TFMatrix4<T>::getTransformationMatrixByXY(const TFVector3<T>& _xdir, const TFVector3<T>& _ydir)
{
    TFVector3<T> zdir=TFVector3<T>::crossProduct(_xdir,_ydir);

    TFVector3<T> ydir=TFVector3<T>::crossProduct(zdir,xdir);

    TFMatrix4<T> mat=getIdentity();
    for (unsigned i=0;i<3;++i)
    {
        mat[i][0]=xdir[i];
        mat[i][1]=ydir[i];
        mat[i][2]=zdir[i];
    }

    return mat;
}
```

Figure 7.17: Templated *getTransformationMatrixByXY()* function from *TFMatrix4* class

### A.4.5 Implementing wrappers

The lifting line method must communicate with other parts of the program in several situations. It uses the templated base classes, whereas the remainder of the program uses the original classes. These are not compatible. Hence, wherever these interfaces occur, the information must be converted from an object of the original base classes to an object of the templated base classes or vice versa. These conversions can be executed by wrapper functions that pass the individual components to the other class. Several wrapper functions had to be developed. One example is *computeForcesA()*. The function is the called by the Newton-Raphson and must return an *FMotor*. However, *computeForcesA()* runs *solve()* that returns *TFMotor<double>*. The function passes the individual components of *TFMotor<double>* that are both of type *double* and are therefore compatible. Similar situations appear in the communication between the GUI and the lifting line method and the condition and the lifting line method. An example of the latter is given in Figure 7.18, where the components of a *FVector3* are passed to a *TFVector3<double>*.

```
template<typename T>
TFVector3<T> FLLAppendage_int2::AppendageData<T>::getLocalVelocityB(const TFPoint3<T>&p, Condition con)const
{
    // linear motion of the origin
    TFVector3<T> vel;
    vel[0] = con.v[0];
    vel[1] = con.v[1];
    vel[2] = con.v[2];
    vel -= TFVector3<T>::crossProduct(p, getOmega(con));
    return vel;
}
```

Figure 7.18: Function to compute the local velocity based on the condition, vel is a *TFVector3<T>*, *con.v* an *FVector3*

### A.4.6 Templating the FLLAppendage module

After completing the templating of the base classes and their integration using wrappers, the next step is the preparation of the lifting line method itself. Every variable and every function of the method that influences the gradient must be templated to be available in *doubles* and *adoubles*. Functions can be templated directly, variables not. Variables can only be templated by templating their parent class or structure. If a structure or class is templated this also directly templates its function declarations. The function definitions still must be templated separately. This process must be applied to the whole lifting line module. Parts of classes that do not affect the gradient, can be separated and can remain of the *double* type.

Figure 7.3 has shown the set-up of the lifting line module. The variables and functions marked in red are those that influence the gradient and require templating. The ones in black can remain of type *double* as they do not influence the gradient. Since both structures, *TAppendage* and *TGrid*, contain active variables and functions, both structures must be templated. However, *TAppendage* also contains variables that are not influencing the gradient as for example the number of sections used (*mNumSections*). These can be separated into a different class that uses *doubles*. This class is named *TAppendageParameter* and stores all passive variables and functions that concern a foil part. This is shown in Figure 7.19, which displays the architecture of the modified lifting line module.

Figure 7.3 has also shown that the overlaying *FLLAppendage* class contains active variables and functions. These are the functions *createGrid()*, *hsVortex()* and *solve()* and the variables *mAppendage* and *mpGrid*. Since variables cannot be templated, this requires the templating of the entire *FLLAppendage* class. However, the *FLLAppendage* class is a child of the *LiftForceModule* class that is used throughout the VPP. The templating of *FLLAppendage* would require significant modifications to the *LiftForceModule* class and other parts of the VPP. Therefore, a new template class called *AppendageData* is introduced as part of the *FLLAppendage* class instead. This class is used to store all active parts such as *TAppendage*, *TGrid*, *mAppendage*, *mpGrid* and the function declarations and automatically templates them. Consequently, *FLLAppendage* can remain a normal class and the remainder of the VPP doesn't have to be changed. *AppendageData* also contains the two newly introduced structures *TConfiguration,* which stores the design points, and *MySurface,* which defines the surfaces for the parametric model. This is shown in Figure 7.19, where parts marked with green colour are templates. *FLLAppendage* is renamed to *FLLAppendage_int2* representing the lifting line module with the internal parametric model in its second version.

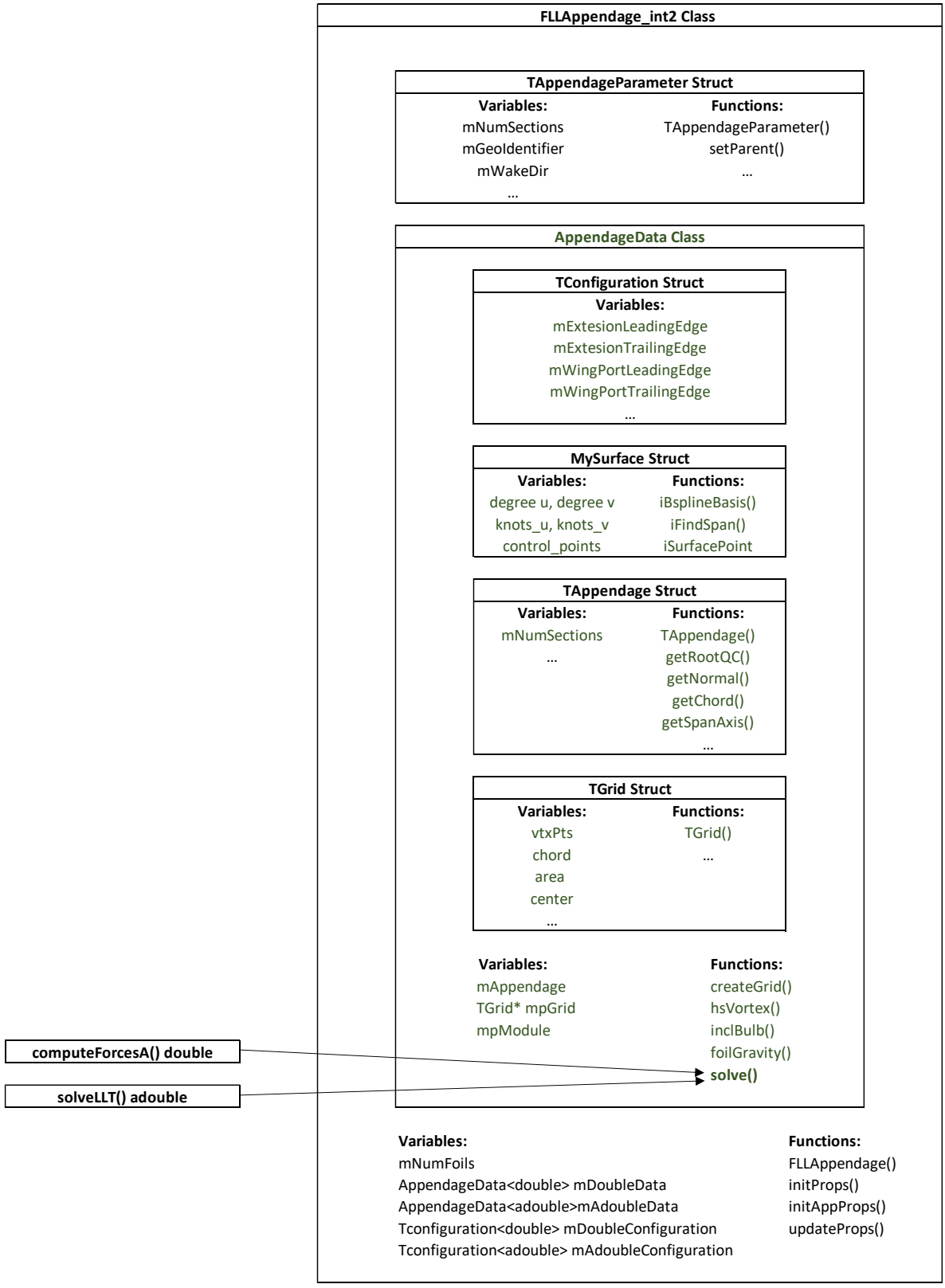



Figure 7.19: Architecture of *FLLAppendage* module in differentiable form called *FLLAppendage_int2*

With all variables and function declarations templated, the next step is to template the function definitions. An example of this is given in Figure 7.20. It shows the function *getChord()* as a member function of *TAppendage*. *TAppendage* is a member of the template class *AppendageData*. The type of *AppendageData* defines the datatype of the arguments of the function, of the employed *TFPoint3s* and of the variable inside the function as well as the return type of the function. This approach is applied to all functions inside *AppendageData*, including *createGrid()*, *hsVortex()* and *solve()*.

```
template<typename T>
T FLLAppendage_int2::AppendageData<T>::TAppendage::getChord(TConfiguration points, T t) const
{
    TFPoint3<T> le= getMyPoint(points, 0., t);
    TFPoint3<T> te= getMyPoint(points, 1., t);

    T distance = le.getDistance(te);

    return distance;
}
```

Figure 7.20: Templated *getChord()* function

In a last step, two objects of type *TConfiguration* are created in *FLLAppendage_int2*, one in *doubles* (*AppendageData<double>::TConfiguration mDoubleConfiguration*) and one in *adoubles* (*AppendageData<adouble>::TConfiguration mAdoubleConfiguration*). These store the design points in the respective types. Furthermore, two objects of AppendageData exist, *AppendageData<double> mDoubleData* and *AppendageData<adouble> mAdoubleData*. These store all active information in *double*s and *adoubles*. This is shown at the bottom of Figure 7.19.

To solve the lifting line method in normal mode, the double type is used (*mDoubleData.solve(mDoubleConfiguration, condition)*). The function is called by the VPP solver through the function *computeForcesA()*. The gradient on the other hand is computed with a function called *solveLLT()*. *solveLLT()* calls *mAdoubleData.solve(mAdoubleConfiguration, mLastCondition)* for solving with *adoubles*. This is discussed in detail in the next section.

### A.4.7 Computing the gradient

With all templating completed the differentiation with ADOL-C is set up in the function *solveLLT()* as shown in Figure 7.21. First, the number of independent and dependent variables is declared. The independent variables are the design variables and the dependent variable is here defined as the lift to drag ratio of the hydrofoil. After declaring the output variable *liftToDragOut*, *convertGuiData()* is called. *convertGuiData()* is a wrapper converting the design variables set in the GUI in *FVector3s* to the *TConfiguration mDoubleConfiguration* based on *TFVector3<doubles>*. Next, the tape is started and the function *doubleToAdouble()* is called.

*doubleToAdouble()* initialises the *mDoubleConfiguration* to the *mAdoubleConfiguration* using the <<= operator. This sets the design variables as the input variables. Subsequently, the lifting line method is run in *adoubles* for the current design. The design is assessed for the condition that was solved last in the *double* mode. After computing the lift to drag ratio and deinitialising it (with the >>= operator), the tape is stopped, and the gradient is evaluated.

```
void FLLAppendage_int2::solveLLT()
{
    int n = 21;          // Specifying the number of independent variables
    int m = 1;           // Specifying the number of dependent variables
    int tag = 1;         // Specifying tape identifier


    double liftToDragOut;


    convertGuiData();


    //-----------------------------------------------------------------------------------------------
    trace_on(tag);       // Starting the tape


    doubleToAdouble();  // Initializing the independent variables


    TFMotor<adouble> output = mAdoubleData.solve(mAdoubleConfiguration, mDoubleData.mLastCondition);

    adouble liftToDrag = (output[2] / output[0]) * -1;


    liftToDrag >>= liftToDragOut;   // Deinitializing the dependent variable

    trace_off();         // Stopping the tape
    //-----------------------------------------------------------------------------------------------



    // result container declaration...

    gradient(tag, n, in, g);    // Computing the gradient

    // printing results...
}
```

Figure 7.21: Implementation of *solveLLT()* for solving lifting line method in *adoubles*

For an exemplary gradient computation using this set-up see Section 6.2. This includes the validation of the gradients with gradients computed using the finite difference method.

## A.5 Summary

This chapter has presented the implementations of the lifting line method and the parametric model, introduced the general steps required to differentiate a function with ADOL-C and discussed the preparation of the models for differentiation. The preparation included replacing parts of the code that are non-differentiable or were contained within pre-compiled libraries. With the whole code visible and differentiable, every variable and function that affects the gradient then had to be redefined to be of the datatype *adouble*. Different options for redefining the existing variables from *doubles* to *adoubles* were presented. Since the code must run in both types, *double* for VPP solving and *adouble* for gradient computation, templating is the only option. Therefore, a new template class (*AppendageData*) was introduced as part of the *LLAppendage* module. All active variables, functions, structures and classes have been made part of this template class. Additionally, all function definitions inside the lifting line module and all relevant base classes defined in external libraries have been templated amounting to around 8000 lines of code. The base classes included for example point-, vector- and matrix libraries. Wrappers have been implemented to enable communication between the VPP, which uses standard base classes, and the adjoint lifting line module, which uses the templated base classes. Two instances of the template class *AppendageData* have been created, one in *doubles* and one in *adoubles*. The VPP solver uses the *double*-based instance, the function computing the gradient the *adouble*-based. This function has been set-up to compute the gradient of the lift to drag ratio with respect to 21 design variables.

# List of References

Albring, T. A., Sagebaum, M. and Gauger, N. R. (2015) 'Development of a Consistent Discrete Adjoint Solver in an Evolving Aerodynamic Design Framework', *AIAA Aviation: 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Dallas, TX.

Alimohammadi, S., Beyhaghi, P., Meneghello, G. and Bewley, T. (2017) 'Delaunay-based optimization in CFD leveraging multivariate adaptive polyharmonic splines (MAPS)', *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. Grapevine, Texas. Reston, Virginia, American Institute of Aeronautics and Astronautics.

America's Cup (2020a) *The AC75: Designed to Fly* [Online]. Available at https://www.youtube.com/watch?v=VQUl_hf6yo8 (Accessed 11 July 2024).

America's Cup (2020) *Virtual Eye* [Online]. Available at https://ac36.americascup.com/en/advanced-dashboard (Accessed 12 August 2022).

America's Cup (2021) *Prada Cup Final and 36th America's Cup Match Amended Race Conditions and Sailing Instructions* [Online]. Available at https://www.americascup.com/news/1040_PRADA-CUP-FINAL-AND-36TH-AMERICAS-CUP-MATCH-AMENDED-RACE-CONDITIONS-AND-SAILING-INSTRUCTIONS (Accessed 15 May 2024).

Andersson, J., Åkesson, J. and Diehl, M. (2012) 'CasADi: A Symbolic Package for Automatic Differentiation and Optimal Control', in Forth, S., Hovland, P., Phipps, E., Utke, J. and Walther, A. (eds) *Recent Advances in Algorithmic Differentiation,* Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 297–307.

Artemis Technologies (2021) *6 DoF RANSE Simulation for Velocity Prediction Video* [Online], LinkedIn. Available at https://www.linkedin.com/company/artemis-technologies-ltd/videos/native/urn:li:ugcPost:6673584814135795712 (Accessed 31 January 2021).

Aubert, P., Di Césaré, N. and Pironneau, O. (2001) 'Automatic differentiation in C++ using expression templates and application to a flow control problem', *Computing and Visualization in Science*, vol. 3, no. 4, pp. 197–208.

Banović, M., Mykhaskiv, O., Auriemma, S., Walther, A., Legrand, H. and Müller, J.-D. (2018) 'Algorithmic differentiation of the Open CASCADE Technology CAD kernel and its coupling with an adjoint CFD solver', *Optimization Methods and Software*, vol. 33, 4-6, pp. 813–828.

Banović, M., Vasilopoulos, I., Walther, A. and Meyer, M. (2020) 'Algorithmic differentiation of an industrial airfoil design tool coupled with the adjoint CFD method', *Optimization and Engineering*, vol. 21, no. 3, pp. 1221–1242.

Baron Von Schertel, H. (1973) 'The Design and Application of Hydrofoils and their Future Prospects', *Joint Meeting with the International Hydrofoil Society,* 30/10/1973.

Besnard, E., Schmitz, A., Kaups, K., Tzong, G., Hefazi, H., Kural, O., Chen, H. and Cebeci, T. (1998) 'Hydrofoil Design and Optimization for Fast Ships', *ASME International Congress and Exhibition*. Anaheim, CA.

Bingol, O. R. and Krishnamurthy, A. (2019) 'NURBS-Python: An open-source object-oriented NURBS modeling framework in Python', *SoftwareX*, vol. 9, pp. 85–94.

Binns, J. R., Hochkirch, K., Bord, F. D. and Burns, I. (2008) 'The Development and Use of Sailing Simulation for IACC Starting Manoeuvre Training', *3rd High Performance Yacht Design Conference*. Auckland, New Zealand.

Bischof, C. H., Carle, A., Corliss, G. F., Griewank, A. and Hovland, P. D. (1992) 'ADIFOR-Generating Derivative Codes from Fortran Programs', *Scientific Programming*, vol. 1, pp. 11–29.

Bischof, C. H., Corliss, G. F., Green, L., Griewank, A., Haigler, K. and Newman, P. (1992) 'Automatic Differentiation of Advanced CFD Codes for Multidisciplinary Design', *Computing Systems in Engineering*, vol. 3, pp. 625–637.

Boegle, C., Hansen, H. and Hochkirch, K. (2012) 'Speed Vs. Stability - Design Considerations and Velocity Prediction of a Hydro-Foiled International Moth', *4th High Performance Yacht Design Conference*. Auckland, New Zealand, pp. 18–27.

Böhm, C. (2014) *A Velocity Prediction Procedure for Sailing Yachts with a Hydrodynamic Model Based on Integrated Fully Coupled RANSE-Free-Surface Simulations*, PhD, TU Delft.

Brockett, T. (1966) *Minimum pressure envelopes for modified NACA-66 sections with NACA A=0.8 camber and bushipstype 1 and type 2 sections: No. DTMB–1780*.

Bücker, M., Hovland, P., Wente, C. and Bach, H. (2024a) *autodiff.org* [Online]. Available at https://www.autodiff.org/?module=Tools (Accessed 11 December 2024).

Bücker, M., Hovland, P., Wente, C. and Bach, H. (2024b) *autodiff.org - Application Database* [Online]. Available at https://www.autodiff.org/?module=Applications (Accessed 11 December 2024).

Buermann, T. M. and Hoerner, S. F. (1964) 'Commercial Hydrofoils', *Naval Engineers Journal*, vol. 76, no. 2, pp. 191–202.

Djavareshkian, M. H. and Esmaeili, A. (2014) 'Heuristic optimization of submerged hydrofoil using ANFIS–PSO', *Ocean Engineering*, vol. 92, pp. 55–63.

Drela, M. (1989) 'XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils', in Müller, T. J. (ed) *Low Reynolds number aerodynamics: Proceedings of the conference, Notre Dame, Indiana, USA, 5 - 7 June 1989,* Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Springer, pp. 1–12.

Drela, M. and Youngren, H. (2021) *Athena Vortex Lattice* [Computer program]. Available at http://web.mit.edu/drela/Public/web/avl/ (Accessed 29 May 2021).

Eggert, F., Henrichs, H., Hansen, H. and Hochkirch, K. (2020) 'Flight Dynamics and Stability Assessment for an International Moth', *5th Int. Conference on Innovation in High Performance Sailing Yachts and Sail-Assisted Ship Propulsion.* Gothenburg, Sweden.

Eppler, R. and Shen, Y. T. (1979) 'Wing Sections for Hydrofoils—Part 1: Symmetrical Profiles', *Journal of Ship Research*, vol. 23, no. 03, pp. 209–217.

Faltinsen, O. M. (2005) *Hydrodynamics of high-speed marine vehicles* [Online], Cambridge, Cambridge Univ. Press. Available at http://www.loc.gov/catdir/enhancements/fy0633/2005006328-d.html.

Fischer, M. (2021) Unpublished interview conducted by R. Tannenberg, 2021.

Forrester, A., Sóbester, A. and Keane, A. (2008) *Engineering design via surrogate modelling: A practical guide* [Online], Chichester, Hoboken, N.J, John Wiley distributor; Wiley. Available at https://onlinelibrary.wiley.com/doi/book/10.1002/9780470770801.

Garg, N., Kenway, G. K., Martins, J. R. and Young, Y. L. (2017) 'High-fidelity multipoint hydrostructural optimization of a 3-D hydrofoil', *Journal of Fluids and Structures*, vol. 71, pp. 15–39.

Gattini, S. (2020) *PRADA ACWS - Day 1 Photo Gallery* [Online]. Available at https://www.lunarossachallenge.com/en/gallery/614_PRADA-ACWS-Day-1-Photogallery (Accessed 22 May 2021).

Gauger, N., Walther, A., Özkaya, E. and Moldenhauer, C. (2012) 'Efficient aerodynamic shape optimization by structure exploitation', *Optimization and Engineering*.

Giering, R., Kaminski, T. and Slawig, T. (2005) 'Generating efficient derivative code with TAF: Adjoint and tangent linear Euler flow around an airfoil', *Future Generation Computer Systems*, vol. 21, no. 8, pp. 1345–1355.

Gill, P. E., Murray, W. and Saunders, M. A. (2005) 'SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization', *SIAM Review*, vol. 47, no. 1, pp. 99–131.

Gilruth, R. (1951) *Hydrofoil craft, USA* US2703063A.

Graf, K., Freiheit, O., Schlockermann, P. and Mense, J. C. (2020) 'VPP-Driven Sail and Foil Trim Optimization for the Olympic NACRA 17 Foiling Catamaran', *Journal of Sailing Technology*, vol. 5, no. 01, pp. 61–81.

Griewank, A. and Walther, A. (2008) *Evaluating derivatives: Principles and techniques of algorithmic differentiation* [Online], 2nd edn, Philadelphia, SIAM. Available at http://bvbr.bib-bvb.de/F?func=service&doc_library=BVB01&doc_number=016738224&line_number=0002&func_code=DB_RECORDS&service_type=MEDIA.

Guida, L., Marimon Giovannetti, L. and Boyd, S. W. (2020) 'Three-Dimensional Variations of the Nacra 17 Main Foil for Benchmarking Shape Optimisations', *5th Int. Conference on Innovation in High Performance Sailing Yachts and Sail-Assisted Ship Propulsion*. Gothenburg, Sweden.

Hagemeister, N. and Flay, R. G. J. (2019) 'Velocity Prediction of Wing-Sailed Hydrofoiling Catamarans', *Journal of Sailing Technology*, vol. 4, no. 01, pp. 66–83.

Hansen, H., Hochkirch, K., Burns, I. and Ferguson, S. (2019) 'Maneuver Simulation and Optimization for AC50 Class', *Journal of Sailing Technology*, vol. 4, no. 01, pp. 142–160.

Hart, F. P., Kriplani, N., Luniya, S. R., Christoffersen, C. E. and Steer, M. B. (2006) 'Streamlined Circuit Device Model Development with fREEDAR® and ADOL-C', in Bücker, M., Corliss, G., Naumann, U., Hovland, P. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 295–307.

Hascoet, L. and Pascual, V. (2013) 'The Tapenade automatic differentiation tool', *ACM Transactions on Mathematical Software*, vol. 39, no. 3, pp. 1–43.

Hazen, G. S. (1980) 'A Model of Sail Aerodynamics for Diverse Rig Types', *The New England Yacht Symposium*.

Hochkirch, K. (2018) *FS-Equilibrium User Manual,* DNV GL SE.

Hochkirch, K. and Bertram, V. (2022) 'Wind assisted propulsion: Economic and ecological considerations', *Maritime Technology and Research*, vol. 4, no. 3, Manuscript.

Hodson, J., Hunsaker, D. F. and Spall, R. (2017) 'Wing Optimization using Dual Number Automatic Differentiation in MachUp', *55th AIAA Aerospace Sciences Meeting*. Grapevine, Texas. Reston, Virginia, American Institute of Aeronautics and Astronautics.

Hoerner, S. F. (1965) *Fluid-Dynamic Drag*, Published by the Author.

Hollenbach, U., Hansen, H., Hympendahl, O., Reche, M. and Carrio, E. R. (2020) 'Wind assisted propulsion systems as key to ultra-energy efficient ships', *12th Symposium on High-Performance Marine Vehicles - Hiper´20*.

Horel, B. and Durand, M. (2019) 'Application of System-based Modelling and Simplified-FSI to a Foiling Open 60 Monohull', *Journal of Sailing Technology*, vol. 4, no. 01, pp. 114–141.

Hospodář, P., Drábek, A. and Prachař, A. (2022) 'Aerodynamic Design and Strength Analysis of the Wing for the Purpose of Assessing the Influence of the Bell-Shaped Lift Distribution', *Aerospace*, vol. 9, no. 1, p. 13.

Hovland, P., Bischof, C., Spiegelman, D. and Casella, M. (1997) 'Efficient Derivative Codes through Automatic Differentiation and Interface Contraction: An Application in Biostatistics', *SIAM Journal on Scientific Computing*, vol. 18, no. 4, pp. 1056–1066.

Hovland, P. D., Naumann, U. and Norris, B. (2002) 'An XML-Based Platform for Semantic Transformation of Numerical Programs', *Software Engineering and Applications*. Cambridge, USA.

Jayaraman, P. *Github - tinynurbs* [Online]. Available at https://github.com/pradeep-pyro/tinynurbs (Accessed 31 May 2022).

Jee, K. W., McShan, D. L. and Fraass, B. A. (2005) 'Implementation of Automatic Differentiation Tools for Multicriteria IMRT Optimization', in Bücker, M., Corliss, G., Hovland, P., Naumann, U. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Springer.

Kandasamy, M., Peri, D., Ooi, S. K., Carrica, P., Stern, F., Campana, E. F., Osborne, P., Cote, J., Macdonald, N. and Waal, N. de (2011) 'Multi-fidelity optimization of a high-speed foil-assisted semi-planing catamaran for low wake', *Journal of Marine Science and Technology*, vol. 16, no. 2, pp. 143–156.

Katz, J. and Plotkin, A. (1991) *Low-speed aerodynamics: From wing theory to panel methods*, New York, McGraw-Hill.

Keane, A. J. and Prasanth, N. B. (2005) *Computational approaches for aerospace design: The pursuit of excellence* [Online], Chichester, England, Hoboken, N.J, Wiley. Available at https://onlinelibrary.wiley.com/doi/book/10.1002/0470855487.

Kennedy, G. J. and Martins, J. R. (2014) 'A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures', *Finite Elements in Analysis and Design*, vol. 87, pp. 56–73.

Kenway, G. K., Mader, C. A., He, P. and Martins, J. R. (2019) 'Effective adjoint approaches for computational fluid dynamics', *Progress in Aerospace Sciences*, vol. 110, p. 100542.

Kerwin, J. E. (1978) *A Velocity Prediction Program for Ocean Racing Yachts,* Massachusetts Institute of Technology, Report No.78-11.

Keuning, J. A. and Sonnenberg, U. B. (1978) 'Approximation of the Calm Water Resistance on a Sailing Yacht Based on the "Delft Systematic Hull Series"', *The Fourteenth Chesapeake Sailing Yacht Symposium.* Annapolis.

Kim, J. G., Hunke, E. C. and Lipscomb, W. H. (2006) 'Sensitivity analysis and parameter tuning scheme for global sea-ice modeling', *Ocean Modelling*, vol. 14, 1-2, pp. 61–80.

Kinnas, S. A. and Fine, N. E. (1993) 'A numerical nonlinear analysis of the flow around two- and three-dimensional partially cavitating hydrofoils', *Journal of Fluid Mechanics*, vol. 254, pp. 151–181.

Knudsen, S. S., Walther, J. H., Legarth, B. N. and Shao, Y. (2023) 'Towards Dynamic Velocity Prediction of NACRA 17', *Journal of Sailing Technology*, vol. 8, no. 01, pp. 1–23.

Kontogiannis, A. and Laurendeau, E. (2021) 'Adjoint State of Nonlinear Vortex-Lattice Method for Aerodynamic Design and Control', *AIAA Journal*, vol. 59, no. 4, pp. 1184–1195.

Kostas, K. V., Ginnis, A. I., Politis, C. G. and Kaklis, P. D. (2017) 'Shape-optimization of 2D hydrofoils using an Isogeometric BEM solver', *Computer-Aided Design*, vol. 82, pp. 79–87.

Larsson, L., Eliasson, R. E. and Orych, M. (2016) *Principles of yacht design*, Camden, Maine, International Marine/McGraw-Hill Education.

Lavimi, R., Le Hocine, A. E. B., Poncet, S., Marcos, B. and Panneton, R. (2024) 'Hull shape optimization of autonomous underwater vehicles using a full turbulent continuous adjoint solver', *Ocean Engineering*, vol. 312, p. 119256.

Lee, K.-J., Hoshino, T. and Lee, J.-H. (2014) 'A lifting surface optimization method for the design of marine propeller blades', *Ocean Engineering*, vol. 88, pp. 472–479.

Liao, Y., Yildirim, A., Martins, J. R. and Young, Y. L. (2022) 'RANS-based optimization of a T-shaped hydrofoil considering junction design', *Ocean Engineering*, vol. 262, p. 112051.

Lyu, Z., Xu, Z. and Martins, J. (2014) 'Benchmarking Optimization Algorithms for Wing Aerodynamic Design Optimization', *8th International Conference on Computational Fluid Dynamics*. Chengdu, Sichuan, China.

Martins, J. R. R. A. and Ning, S. A. (2022) *Engineering design optimization*, Cambridge, Cambridge University Press.

Melis, F. M., Tannenberg, R., Boyd, S. W., Scharf, M. and Abdel-Maksoud, M. (2024) 'AC75 Aerodynamic Force Prediction Using a 3D Panel Code', *8th High Performance Yacht Design Conference*. Auckland, New Zealand.

Melis, M. F., Hansen, H., Fischer, M. and Abdel-Maksoud, M. (2022) 'Velocity Prediction Program for a Hydrofoiling Lake Racer', *Journal of Sailing Technology*, vol. 7, no. 01, pp. 255–275.

Meneghello, G., Beyhaghi, P. and Bewley, T. (2016) *Simulation-based optimization of the hydrofoil of a ying catamaran* [Online]. Available at https://www.lmd.ens.fr/mgl/pdf/meneghello0000simulation.pdf (Accessed 11 May 2024).

Mishima, S. and Kinnas, S. A. (1996) 'A numerical optimization technique applied to the design of two-dimensional cavitating hydrofoil sections', *Journal of Ship Research*, 40(1), pp. 28–38.

Molland, A. F., Hudson, D. A. and Turnock, S. R. (2017) *Ship resistance and propulsion: Practical estimation of ship propulsive power*, Cambridge, Cambridge University Press.

Molland, A. F. and Turnock, S. R. (2022) *Marine rudders, hydrofoils and control surfaces: Principles, data, design and applications*, Oxford, Cambridge, Butterworth-Heinemann an imprint of Elsevier.

Mykhaskiv, O., Banović, M., Auriemma, S., Mohanamuraly, P., Walther, A., Legrand, H. and Müller, J.-D. (2018) 'NURBS-based and parametric-based shape optimization with differentiated CAD kernel', *Computer-Aided Design and Applications*, vol. 15, no. 6, pp. 916–926.

Ng, G. W., Liao, Y., Yildirim, A. and Martins, J. R. (2025) 'Hydrostructural optimization of subcavitating cambered and symmetric composite foils', *Composite Structures*, vol. 351, p. 118545.

North Sails (2024) *North VPP* [Computer program]. Available at https://www.northsails.com/de-dach/pages/sail-design.

Offshore Racing Congress (2016) *ORC VPP Documentation*.

Offshore Racing Congress (2024) *ORC VPP* [Computer program]. Available at Offshore Racing Congress.

Patterson, N. and Binns, J. (2022) 'Development of a Six Degree of Freedom Velocity Prediction Program for the foiling America's Cup Vessels', *Journal of Sailing Technology*, vol. 7, no. 01, pp. 120–151.

Paulin, A. (2013) *Performance Assessment and Optimisation of a C Class Catamaran Hydrofoil Configuration with the Software Tool FS-Equilibrium*, MSc.

Paulin, A., Hansen, H., Hochkirch, K. and Fischer, M. (2015) 'Performance Assessment and Optimization of a C-Class Catamaran Hydrofoil Configuration', *5th High Performance Yacht Design Conference*. Auckland, New Zealand.

Peri, D. (2023) 'Design Optimization of a Cavitating 2D Profile in Proximity of the Free Surface', *Journal of Sailing Technology*, vol. 8, no. 01, pp. 143–160.

Phelivan Solak, H., Wackers, J., Pellegrini, R., Serani, A., Diez, M., Perali, R., Sacher, M., Leroux, J., Augier, B., Hauville, F. and Bot, P. (2023) 'Hydrofoil Optimization via Automated Multi-Fidelity Surrogate Models', *10th Conference on Computational Methods in Marine Engineering,* CIMNE.

Phillips, W. F. and Snyder, D. O. (2000) 'Modern Adaptation of Prandtl's Classic Lifting-Line Theory', *Journal of Aircraft*, vol. 37, no. 4, pp. 662–670.

Piegl, L. and Tiller, W. (1997) *The NURBS Book*, 2nd edn, Berlin, Heidelberg, Springer Berlin / Heidelberg.

Ploe, P. (2018) *Surrogate-based optimization of hydrofoil shapes using RANS simulations*, PhD.

Prandtl, L. (1921) *Applications of modern hydrodynamics to aerodynamics: NACA Report No. 116*.

Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T. (1988) *Numerical recipes in C*, New York, Cambridge University Press.

Reid, J. T. (2020) *A General Approach to Lifting Line Theory, Applied to Wings with Sweep*, PhD.

Revels, J., Lubin, M. and Papamarkou, T. (2016) *Forward-Mode Automatic Differentiation in Julia* [Online]. Available at http://arxiv.org/pdf/1607.07892v1.

Richardt, T., Harries, S. and Hochkirch, K. (2005) 'Maneuvering simulations for ships and sailing yachts using FRIENDSHIP-Equilibrium as an open modular workbench', *International*

*EuroConference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2005)*. Hamburg, Germany.

Röbenack, K. (2007) 'Controller design for nonlinear multi-input – multi-output systems based on an algorithmic plant description', *Mathematical and Computer Modelling of Dynamical Systems*, vol. 13, no. 2, pp. 193–209.

Robin, P., Leroyer, A., De Premorel, D. and Wackers, J. (2023) 'Tackling Modern Sailing Challenges with a CFD-based Dynamic VPP', *6th Int. Conference on Innovation in High Performance Sailing Yachts and Sail-Assisted Ship Propulsion*. Lorient, France.

Rodriguez, R., Wang, Y., Ozanne, J., Sumer, D., Filev, D. and Soudbakhsh, D. (2022) 'Adaptive Takeoff Maneuver Optimization of a Sailing Boat for America's Cup', *Journal of Sailing Technology*, vol. 7, no. 01, pp. 88–103.

Roth, R. and Ulbrich, S. (2013) 'A Discrete Adjoint Approach for the Optimization of Unsteady Turbulent Flows', *Flow, Turbulence and Combustion*, vol. 90, no. 4, pp. 763–783.

Rousselon, N. (2020) 'The Use of Flow Simulations at Artemis Racing for the 35th America's Cup', *5th Int. Conference on Innovation in High Performance Sailing Yachts and Sail-Assisted Ship Propulsion*. Gothenburg, Sweden.

Roux, Y., Leroyer, A., Raymond, J. and Hauville, F. (2008) 'Strongly coupled VPP and CFD RANSE code for Sailing Yacht Performance Prediction', *3rd High Performance Yacht Design Conference*. Auckland, New Zealand.

Royal New Zealand Yacht Squadron and Circolo Della Vela Sicilia (2020) *AC75 Class Rule v1.16*.

Royal New Zealand Yacht Squadron and Royal Yacht Squadron (2023) *AC75 Class Rule v2.9*.

Sacher, M., Durand, M., Berrini, E., Hauville, F., Duvigneau, R., Le Maitre, O. and Astolfi, J. A. (2017) 'Flexible hydrofoil optimization for the 35th America's cup with constrained Ego Method', *4th Int. Conference on Innovation in High Performance Sailing Yachts*.

Sagebaum, M., Albring, T. and Gauger, N. R. (2019) 'High-Performance Derivative Computations using CoDiPack', *ACM Transactions on Mathematical Software*, vol. 45, no. 4, pp. 1–26.

Sarikaya, B. and Tuncer, I. H. (2022) 'Adjoint Based Design Optimization of Subsonic Airfoils using a Panel Code Together with a RANS Solver', *AIAA SCITECH 2022 Forum*. San Diego, CA & Virtual. Reston, Virginia, American Institute of Aeronautics and Astronautics.

Schlenkrich, S., Walther, A., Gauger, N. R. and Heinrich, R. (2008) 'Differentiating Fixed Point Iterations with ADOL-C: Gradient Calculation for Fluid Dynamics', in Bock, H. G., Kostina, E.,

Phú, H. X. and Rannacher, R. (eds) *Modeling, simulation and optimization of complex processes: Proceedings of the third International Conference on High Performance Scientific Computing, March 6 - 10, 2006, Hanoi, Vietnam,* Berlin, Heidelberg, Springer, pp. 499–508.

Schnerr, G. H. and Sauer, J. (2001) 'Physical and Numerical Modeling of Unsteady Cavitation Dynamics', *4th International Conference on Multiphase Flow.* New Orleans, USA.

Schoenholz, S. S. and Cubuk, E. D. (2020) 'JAX, M.D.: A Framework for Differentiable Physics', *Advances in Neural Information Processing Systems 33 ([Online]*. Available at http://arxiv.org/pdf/1912.04232.

Shen, Y. T. and Eppler, R. (1981) 'Wing Sections for Hydrofoils—Part 2: Nonsymmetrical Profiles', *Journal of Ship Research*, vol. 25, no. 03, pp. 191–200.

Siemens Digital Industries Software (2020) *Simcenter STAR-CCM+* [Computer program].

Silva, S. (2014) *C-Class Catamaran Daggerboard: Analysis and Optimization* [Online]. Available at https://fenix.tecnico.ulisboa.pt/downloadFile/844820067124325/63294_paper.pdf (Accessed 11 April 2024).

Specialist Committee of 23rd ITTC (2002) 'Recommended Procedures and Guidelines - Testing and Extrapolation Methods, Resistance Uncertainty Analysis, Example for Resistance Test', *International Towing Tank Conference*.

SumToZero (2021) *GOMBOC Designer* [Online]. Available at http://sumtozero.com/products/gomboc-designer/ (Accessed 31 January 2021).

Tadjouddine, M., Forth, S. A. and Keane, A. J. (2006) 'Adjoint Differentiation of a Structural Dynamics Solver', in Bücker, M., Corliss, G., Naumann, U., Hovland, P. and Norris, B. (eds) *Automatic Differentiation: Applications, Theory, and Implementations,* Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 309–319.

Tannenberg, R., Turnock, S. R., Hochkirch, K. and Boyd, S. W. (2023) 'VPP Driven Parametric Design of AC75 Hydrofoils', *Journal of Sailing Technology*, vol. 8, no. 01, pp. 161–182.

Tozzi, G. M. (2004) *Hydrofoil Shape Optimization by Gradient Methods*, MSc.

Tyde (2024) *Electric hydrofoiling yachts* [Online]. Available at https://tyde.one/ (Accessed 25 November 2024).

Wächter, A. and Biegler, L. T. (2006) 'On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming', *Mathematical Programming*, vol. 106, no. 1, pp. 25–57.

Walther, A. and Griewank, A. (2012) 'Getting started with ADOL-C', in Naumann, U. and Schenk, O. (eds) *Combinatorial Scientific Computing,* Chapman and Hall/CRC, pp. 181–202.

Walther, A. and Griewank, A. (2022) *ADOL-C: A Package for the Automatic Differentiation of Algorithms Written in C/C++* [Online]. Available at https://github.com/coin-or/ADOL-C/blob/master/ADOL-C/doc/adolc-manual.pdf (Accessed 11 December 2024).

Weather Underground (2024) *Historical Weather Data Auckland, New Zealand* [Online]. Available at https://www.wunderground.com/history/monthly/nz/auckland/NZAA/date/2020-3 (Accessed 15 May 2024).

Whicker, F. and Fehlner, L. F. (1958) *Free-Stream Characteristics of a Family of Low-Aspect Ratio, All Moveable Control Surfaces for Application to Ship Design: Report No. 933, David Taylor Model Basin*.

Wilkins, I. *Gomboc: A design high-flier for ETNZ* [Online]. Available at https://gomboc.eu/docs/AMERICACUP_GOMBOC.pdf (Accessed 14 November 2024).

Wolfson Unit (2024) *WinDesign4* [Computer program]. Available at https://www.wumtia.soton.ac.uk/softwares/sailing-velocity-prediction-programme/.

Zeng, Z. and Kuiper, G. (2012) 'Blade Section Design of Marine Propellers with Maximum Cavitation Inception Speed', *Journal of Hydrodynamics*, vol. 24, no. 1, pp. 65–75.