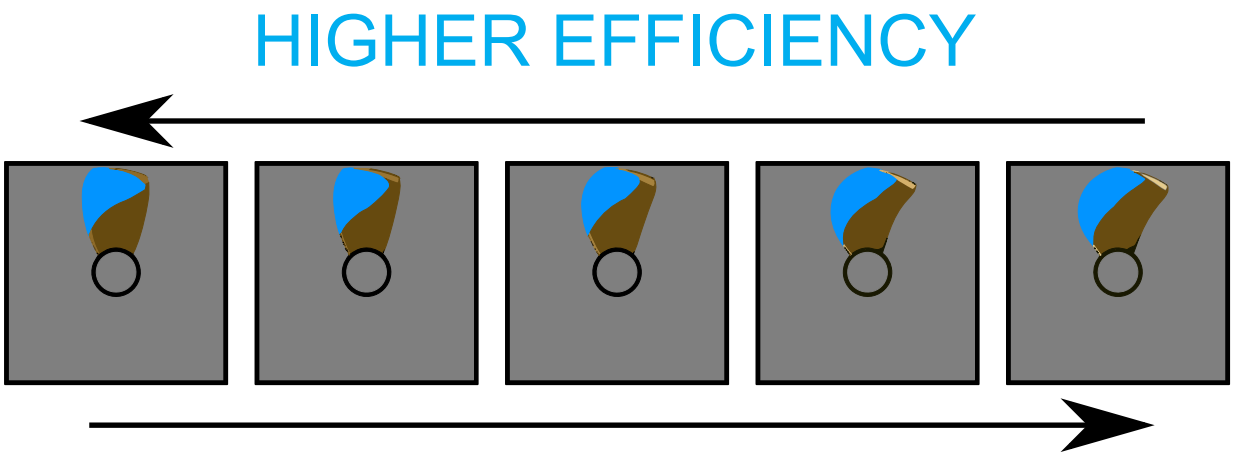
Master of Science Thesis




# Development of a Multiobjective Design Optimization Procedure for Marine Propellers

## A Case Study to the Trade-Off between Efficiency and Cavitation Performance

**C.P. Pouw**

December 19, 2007

Hydrodynamics

TU Delft

Faculty of Marine Engineering

Delft University of Technology

# Development of a Multiobjective Design Optimization Procedure for Marine Propellers

## A Case Study to the Trade-Off between Efficiency and Cavitation Performance

Master of Science Thesis

For obtaining the degree of Master of Science in Marine Technology at Delft University of Technology

C.P. Pouw

December 19, 2007

Faculty 3ME · Delft University of Technology

**Delft University of Technology**

# Preface

Now, after more than ten years of studying Marine Engineering in Delft, I'm at the verge of concluding my final thesis. A crown-achievement as we would say and to me sure feels like it. A lot of things happened before I came this far. An overview of my extensive student life in Delft is a little too far fetched but an introduction to this work is not misplaced.

My interest for propeller design started when I was involved in the human powered submarine team WaSub II. One of my tasks was to designed the propeller and I found out that estimating the wake field was not as easy as I thought. I also found out that there is more than the Wageningen B-Series that every student gets to learn during his/her studies. So WaSub was like a stepping stone for me to come to MARIN. Here they were interested to investigate if something interesting could be done with a propeller panel code in combination with a genetic algorithm on the other side.

This work could not have been done without the support of the people at MARIN. Gratitude goes to my supervisor Gert Jan Zondervan with whom I could exchange ideas and who helped me out with the programs they use here at MARIN. To my professor Tom van Terwisga who always has a positive point of few on things and who also supported me with general thesis troubles. To the rest of the people working at MARIN, in most cases it is not a problem just to walk into someones office and ask your question(s).

Last but not least to the fellow students here with whom I could have fun during the day but also after the day has finished. To be more specific:
Maarten, for cleaning his desk which would later become mine.
Bastien, if someone taught me how to pronounce *putain*, it is him.
Frans Hendrik, for the hours of endless music.
Emanuel, it is always good to have a German in your room.
Charles, who somehow got my mummies email address...
Mayumi, the only person who really understood me!
Aline, who is never late except in Holland.
Thijs, got me back into the blues.
Jasper, just being a guitar player is already worth mentioning.
Jorrit Jan, are there already bananas growing in your office?
Wim van Rees, who got addicted to juggling, don't ask me why.
Arnoud, lekkerrr! bier.

# Summary

**propeller design problem** The design of a marine propeller is characterized by its complexity rather then its shortcoming of knowledge how to asses its performance. There are several constraints to satisfy and all are in a different field. The three major considerations are strength, performance (efficiency) and cavitation behaviour. There is not a perfect *design methodology* for the engineer how to use the different *analysis tools* available to come to a final design. The process is in most cases an iterative one that ends with a *satisfying* design rather then an optimal design. Optimal in this case means optimal in the sense of the best compromise possible.

**application of a multiobjective optimiser** The application of a multiobjective optimiser makes it possible to visualise the trade off among different conflicting objectives to guide the engineer in making his compromise. Furthermore it gives more insight in the problem at hand. The goal of this thesis is therefore two-fold. First one is the implementation of a multiobjective optimiser to show what the gains are when an analysis tool can be turned into a design tool. The second one is to apply it and investigate what the trade-off is between cavitation performance and efficiency for a test case based on a container vessel.

The implemented algorithm is the *non-dominated sorting genetic algorithm* (NSGA) which is currently used in many other practical design problems. Genetic Algorithms are in general robust but not very efficient when it comes to the number of design evaluations it has to do. The algorithm is able to give a good approximation of the trade-off with a good diversity among its solutions. The propeller is analysed by the lifting surface program ANPRO which has an extension to predict sheet cavitation and bubble cavitation at both sides of the blade. In the test case considered there were additional constraints for a minimum blade thickness, the avoidance of bubble cavitation and a maximum allowance of sheet cavitation at the pressure side.

**results** The algorithm is able to converge to a trade-off between the two conflicting objectives. The size of the population is chosen as 80 and the algorithm run for 300 generations. From statistical data of the average age of the individuals in the population, converge could be determined. The simulations were not able to converge completely but sufficiently enough to obtain an approximation of the Pareto front. The trade-off show a decrease from

maximum efficiency of 4% while gaining a reduction of sheet cavitation at the suction side of 27%. The amount of reduction of efficiency was roughly the same for different test cases but the cavitation percentages showed more variation. The method showed to be sensitive to the interpretation of the cavitation data produced by the propeller analysis program ANPRO.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

**Motivation** In the detailed design of a propeller, several requirements have to be satisfied. The two most important ones are efficiency and cavitation performance. Efficiency is important from an economical viewpoint. The less fuel that is consumed, the cheaper the price that has to be paid for the transport. Lower fuel consumption also means lower emission of exhaust gasses. Lower cavitation on the other hand is important to reduce the radiated noise from a propeller, to prevent possible erosion caused by implosions of cavities and to reduce structural vibrations in the aft part of the ship.

These two goals are said to be conflicting as they can't be satisfied fully without degrading the other. It is the task of the designer to find a trade-off between these two and come up with a 'compromise' that gives a satisfying design. The motivation of this work is to better understand this trade-off and this thesis tries to investigate these two conflicting objectives by approaching the problem from a multiobjective point of view. This means that the optimisation is done with regard to both objectives at the same time which should result in a better insight in the trade-off between these two.

**Background** In the preliminary design, the designer is interested in defining the overall parameters of a propeller like diameter, pitch-ratio, blade area ratio and the number of blades. Systematic series, like for example Kuiper [21] and Gawn [15], give a fast and reliable prediction. They are easy to apply and implement as they nowadays consist of a set of polynomials. An assessment of the cavitation can only be done by a rough approximation based on the loading of the propeller and the cavitation number. The result is a minimum required area to prevent cavitation like the criteria as given by Keller [20].

A more advanced approach would be to include knowledge about cavitation behaviour together with propeller performance data. There are two cases found in literature where these two are combined and where an optimiser based on a genetic algorithm is applied to find optimal solutions.

The first application is from Suen and Kouh [30] where they combine the cavitation criteria from Keller with the B-Series propeller data. They have a number of constraints to keep

the variables within the validity of the propeller series and to implement the criterion from Keller. Their optimisation is a single-objective one and it searches for the optimal propeller satisfying the given cavitation criteria. With their article they show that the application of a GA is a good alternative to the method as originally described in Kuiper [21].

The second application is a multiobjective design optimization by Benini [4] where he uses an Evolutionary Algorithm to visualise the trade-off between cavitation and efficiency. He also makes use of the B-Series propeller data and the Keller formula for minimum blade area ratio. The trade-off is shown for propeller loading $K_T$ and propeller efficiency $\eta$ where the minimum blade area according to Keller is taken into account as a constraint. His multiobjective approach is new in propeller design.

Both references apply a more integrated approach of the propeller design but the next step would be to implement a more realistic cavitation model or a more accurate propeller model. The application of a panel method is for example done by Caponnetto [7] but this is without a cavitation model. Instead he used the condition of shock free entrance of the blade sectional profiles to reduce the risk of cavitation.

**Marin** This thesis is done for the Maritime Research Institute Netherlands, an institute that provides the industry (like commercial shipbuilders, fleet owners, navies, offshore companies) with maritime expertise. Their mission is to provide the industry with innovative design solutions and to carry out advanced research for the benefit of the maritime sector as a whole. Part of the section Ships/Powering makes propeller designs on a regular basis and applies their knowledge which they gained in the field of cavitation in the form of numerical tools and design guidelines.

**Outline** This thesis is setup to first give an overview of the 'classical' propeller design process in the first chapter. The problem formulation together with the approach taken to tackle this problem is given here too. Chapter 3 introduces the different theoretical models that are used. These models are divided into the propeller analysis models, cavitation models and the optimisation algorithm. To do an optimisation the geometry of the propeller has to be described by variables. How this parameterisation is done is described in chapter 4. The different modules (optimiser and analysis tools) have to be implemented and connected together which is described in Chapter 5. Once the framework is set, a preliminary design case and a detailed design case are treated in chapter 6. These cases deal with the propeller design of a 6800 TEU container vessel. The results are discussed in chapter 7 and chapter 8 concludes this work, gives recommendations and a future outlook.

# Chapter 2

# Problem formulation

This chapter consists of four parts. The first part elaborates on the propeller design process as it is undertaken nowadays by most designers. It gives a rough overview of the different steps that are undertaken to come to a final design. The second part identifies the different conflicting objectives that can be found in propeller design. The third part is devoted to formulate an optimisation problem from a mathematical point of view. It also introduces the concept of Pareto optimal solutions which is helpful in multiobjective design optimisation. The last part combines propeller design and multiobjective design optimisation into one and points out what will be focused on.

## 2.1 Classical propeller design process

Bertram [5] gives a good overview of the design process which takes places at HSVA in Hamburg. It is a general outline which only in detail differs from other designers. Especially how to decide between trade-offs (how to make compromises) and some design philosophies (to what extend is pressure side sheet cavitation allowed, which rules and regulations to apply).

**1 Preparation of model experiments** Known at this stage are the rpm, ship speed, estimate of delivered power, ship hull form and which classification society to apply regulations from. Often also the number of blades and the diameter is known. The model experiments take place with a stock propeller that resembles the initial design and is meant to get information about the propeller hull interaction and information about the wake field.

**2 Estimate effective wake distribution full scale** The measured wake field at model scale should be transformed to full scale. In most cases only the axial velocity component is scaled and there are several methods to do this scaling. The problem lies in properly scaling the boundary layer from model scale to full scale. An alternative approach to get the full scale wake field is to do a RANS computation.

**3 Determine profile thickness according to classification society** A minimum thickness is required by the chosen classification society which is straight forward to apply.

**4 Lifting-line and lifting-surface calculations** These calculations are undertaken to determine the camber distribution and pitch distribution. The lifting-line method needs an initial starting value and takes the optimal circulation distribution for this. The results are then input for a lifting surface *design* code to determine again the camber and pitch distribution but now the method is a three dimensional model.

**5 Smoothing results of Step 4** The results of the three-dimensional panel code are generally not smooth and feature singularities at the hub and tip of the propeller. The designer deletes these points and specifies values at the hub and the tip based on experience.

**6 Final hydrodynamic analysis** In this stage the design is analysed in all operating conditions using a lifting surface *analysis* program and the results incorporate cavitation performance and vibrational aspects of the design. In this stage the designer, based on its experience, iterates within this step until he/she finds a *satisfying* design.

**7 Check against classification society rules** In this stage a FEM is applied to determine the stresses in the propeller. This is checked against maximum allowed stresses according to the chosen classification society. In most cases the stresses are within the limits of the society. If not, the design is changed and step 6 is repeated again.

This overview separates the design in roughly three different parts: model tests, design part and an iterative analysis part. It is in the *last part* where still decisions are made concerning conflicting objectives like cavitation performance but more on a detailed level. It is also the part that is the most cumbersome to undertake, the design is more detailed in describing the geometry and it requires special numerical tools.

## 2.2 Conflicting objectives

The efficiency of a propeller is mainly determined by two phenomena, induced velocities and frictional losses where the latter one is a viscous effect. Lerbs [24] showed how to derive an optimal circulation distribution to obtain minimum induced losses in the slipstream and thereby maximising the efficiency. He did this by modeling the propeller by bound vortices representing the blades and free vortices trailing from the blades. He could now solve for a minimisation of the induced velocities in the slipstream and came up with an 'optimal distribution' to achieve this.

An example to illustrate the deviation from this optimal distribution is the suppression of the tip vortex. The strength of the tip vortex is directly related to the gradient of

the circulation distribution near the tip and to influence the strength, changes can be made to the shape of the circulation distribution. The optimal circulation distribution therefore changes to a less optimal shape and hence results in a reduction in efficiency. The improvement of efficiency on one side or the reduction of the tip vortex on the other side are therefore said to be conflicting objectives.

Another important parameter in the design of a propeller is the blade area. A larger area means a reduction in the loading per unit blade area which results in a lower maximum pressures at the suction side. Lower maximum pressures reduce the risk of cavitation and improve the cavitation behavior. A larger area on the other hand means more surface friction which reduces the effective thrust and increases the required torque. Here we see again that a reduction of cavitation conflicts with an increase in efficiency.

## 2.3 Optimisation

This section explains some concepts of optimisation. It starts with a mathematical description of an optimisation problem and after that explains by means of an simple example how to deal with multiobjective problems.

### 2.3.1 Definition of an optimisation problem

Before we distinguish between single- and multiobjective optimisation problems, the mathematical formulation of a general optimisation problem is given in formulae 2.1 to 2.4.

$$\begin{aligned} &f_1(\mathbf{x}) \rightarrow min! \\ &\quad\vdots \\ &f_n(\mathbf{x}) \rightarrow min! \end{aligned} \qquad \text{Objectives} \tag{2.1}$$

$$\begin{aligned} &g_1(\mathbf{x}) \leq 0 \\ &\quad\vdots \\ &g_i(\mathbf{x}) \leq 0 \end{aligned} \qquad \text{Inequality constraints} \tag{2.2}$$

$$\begin{aligned} &h_1(\mathbf{x}) = 0 \\ &\quad\vdots \\ &h_j(\mathbf{x}) = 0 \end{aligned} \qquad \text{Equality constraints} \tag{2.3}$$

$$\mathbf{x} \in \mathcal{X} = [\mathbf{x}^{min}, \mathbf{x}^{max}] \subset \mathbb{R}^{n_z} \times \mathbb{Z}^{n_z} \qquad \text{Box constraints} \tag{2.4}$$

Here the objectives $f_n(\mathbf{x})$ have to be minimised or maximised which depends on the problem formulation. A maximisation problem can easily be transformed into a minimisation problem by multiplying it with $-1$. A problem consists of design variables that make up the *design space*. The design variable $\mathbf{x}$ is written in vector form to emphasise that there is in most cases more than one variable that makes up the *design space*.

The *search space* $\mathcal{X}$ is the space that is made up by the design variables $\mathbf{x}$ but which is kept within certain boundaries by the so called box constraints of formula 2.4. The variables can either be real or integer. The (in)equality constraints of formulae 2.3 and 2.2 limit the *search space* even more to a so called *feasible search space*. Optimisation is a search within this *feasible space* for candidates with the lowest objective evaluation.

### 2.3.2 Multiobjective optimization

When there is one objective, it is easy to find the minima. In practice, design problems almost always have more than one objective to minimise for. Most optimization algorithms are however laid out to solve single objective problems. Therefore practical design problems are transformed into a single objective form by using so called weighting factors that combine all different objectives into a single one. The trade-off between two or more objectives is lost the moment a choice is made for the weighting factors. The following example illustrates this.

**simple problem - definition** This example problem consists of only two variables and two objective functions as given by formulae 2.5 and 2.6. The *design space* is 2-dimensional, the box constraints define the *search space* to be in the domain of -3 to 3 for both variables. There are no constraints in this problem and the objective functions are both parabola.

$$f_1(\mathbf{x}) = (x_1 + 0.5)^2 + (x_2 + 1)^2 + 5 \qquad \text{objective 1} \tag{2.5}$$

$$f_2(\mathbf{x}) = 0.75(x_1 - 0.5)^2 + {x_2}^2 + 5 \qquad \text{objective 2} \tag{2.6}$$

$$\mathbf{x} \in [-3, 3]^d \quad \text{with} \quad d = 2 \qquad \text{box constraints} \tag{2.7}$$

**objective function visualisation** The minima for both objectives can both be found analytically and are $\mathbf{x} = [-0.5, -1]$ and $\mathbf{x} = [0.5, 0]$. If we make a landscape plot of both objective functions we can draw the same conclusions (see figure 2.1 and figure 2.2). We immediately see that it is impossible to find an $\mathbf{x}$ that could satisfy both minima of the two objectives, the objectives are said to be *conflicting* with each other.

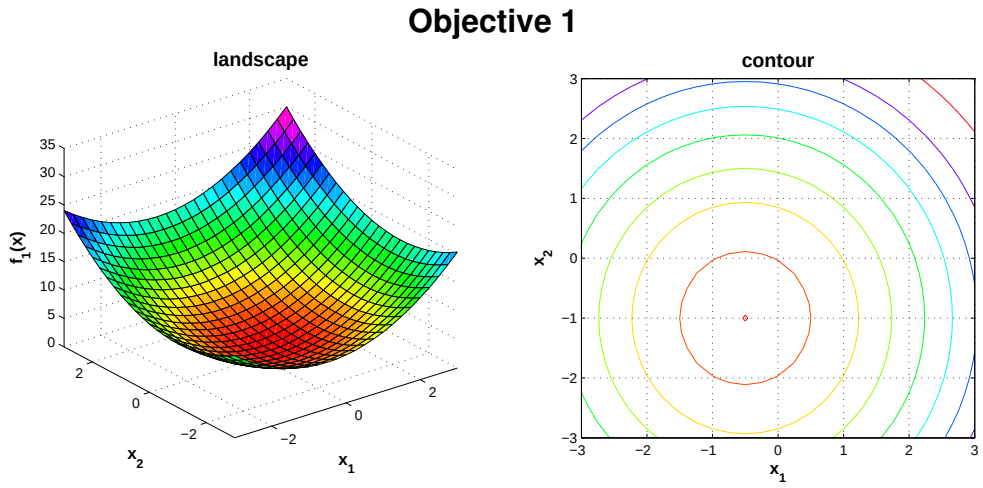



**Figure 2.1:** landscape and contour plot of the first objective function, its minimum is located at $\mathbf{x} = [-0.5, -1]$

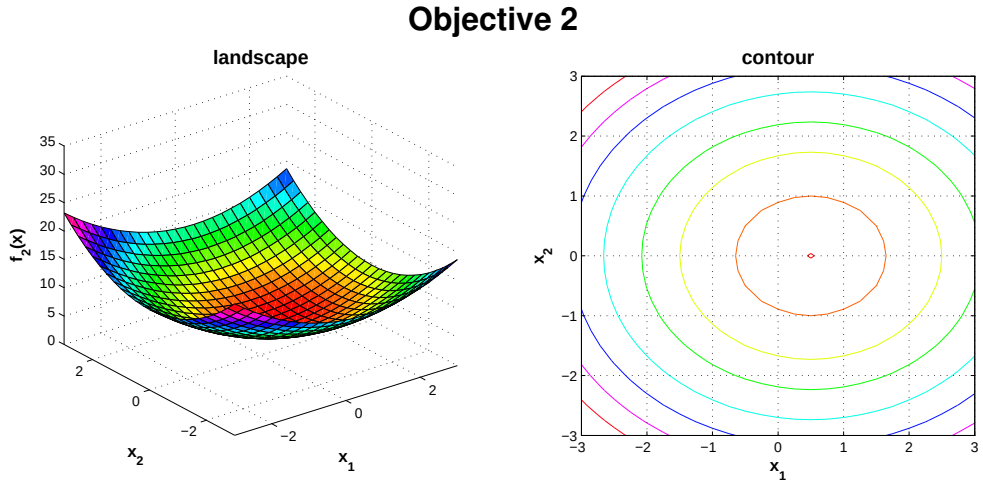



**Figure 2.2:** landscape and contour plot of the second objective function, its minimum is located at $\mathbf{x} = [0.5, 0]$

**weighting factor** A solution would be to combine both objectives into one with the aid of a weighting factor as given in formula 2.8. A possible choice for a weighting factor of $w_1 = 0.65$ and $w_2 = 0.35$, is plotted in figure 2.3. The minima that corresponds to this new objective function is approximately $[-0.212, -0.598]$.

$$f = w_1 f_1 + w_2 f_2 \tag{2.8}$$

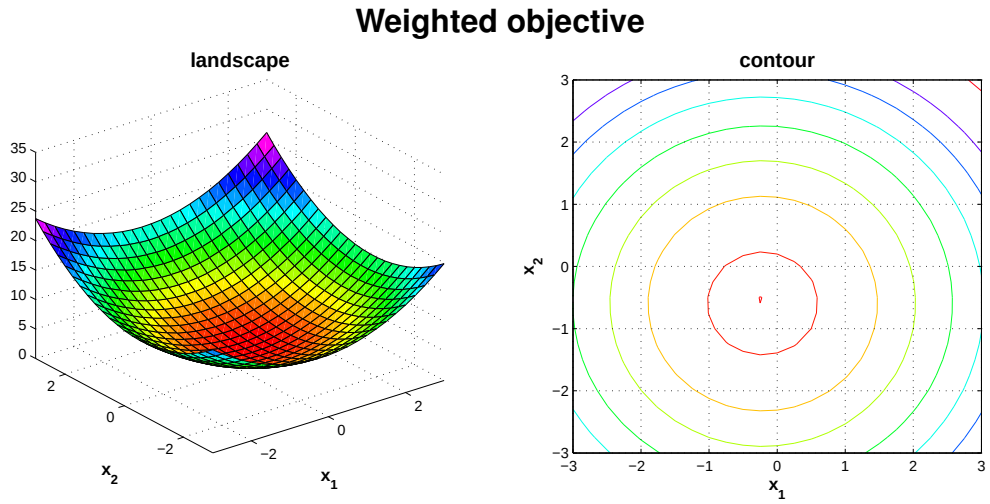



**Figure 2.3:** landscape and contour plot for a weighted objective function of $0.65f_1 + 0.35f_2$, its minimum is located at $\mathbf{x} = [-0.212, -0.598]$

**optimal solutions** If we would calculate a series of optimal solutions for different choices of weighing functions, we can construct a line which resembles the *trade-off* between these two objectives. This is shown in figure 2.4 and table 2.1 gives an overview of these solutions for the different weighing factors. These two figures also show that both objectives conflict with each other. Improvement in one objective means moving away from the optima of the other objective. Note that only a part of the domain of $\mathbf{x}$ is now plotted in both figures. This to make the curve better visible.

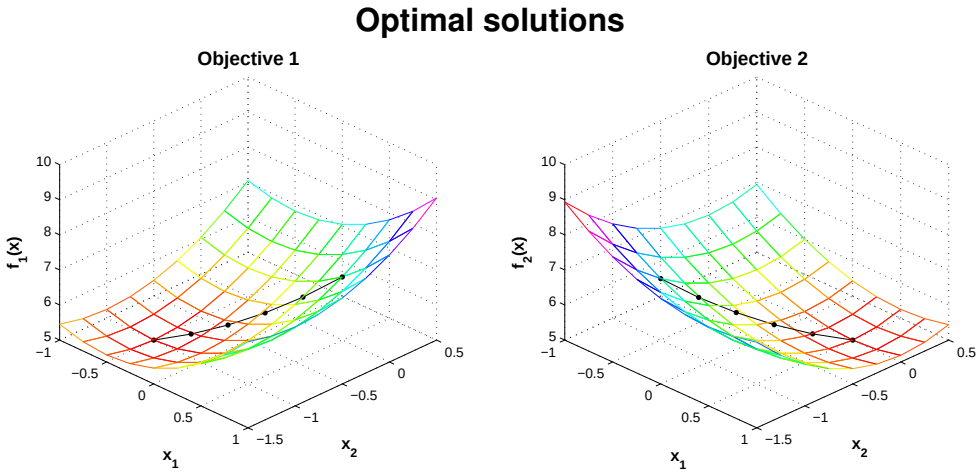



**Figure 2.4:** landscape plot of both objective functions with added the curve of optimal solutions for changing weighing factors

**Table 2.1:** Overview of optimal solutions for both objective functions

| $w_1$ | $w_2$ | $x_1$ | $x_2$ | $f_1(\mathbf{x})$ | $f_2(\mathbf{x})$ | $w_1 f_1 + w_2 f_2$ |
|---|---|---|---|---|---|---|
| 0.0 | 1.0 | 0.500 | 0.000 | 6.800 | 5.000 | 5.000 |
| 0.2 | 0.8 | 0.250 | -0.167 | 6.118 | 5.075 | 5.283 |
| 0.4 | 0.6 | 0.029 | -0.348 | 5.621 | 5.287 | 5.421 |
| 0.6 | 0.4 | -0.167 | -0.545 | 5.276 | 5.631 | 5.418 |
| 0.8 | 0.2 | -0.342 | -0.762 | 5.070 | 6.112 | 5.279 |
| 1.0 | 0.0 | -0.500 | -1.000 | 5.000 | 6.750 | 5.000 |

**Pareto optimal solutions** If we plot the objective values $f_1$ and $f_2$ of these optimal solutions against each other in one plot we get something like figure 2.5. By doing so we obtain a set of so called *Pareto optimal solutions* which is named after the Italian *Vilfredo Pareto* who introduced the concept of *Pareto efficiency*. A multiobjective optimisation algorithm tries to approximate these solutions but uses a different approach to obtain these solutions.

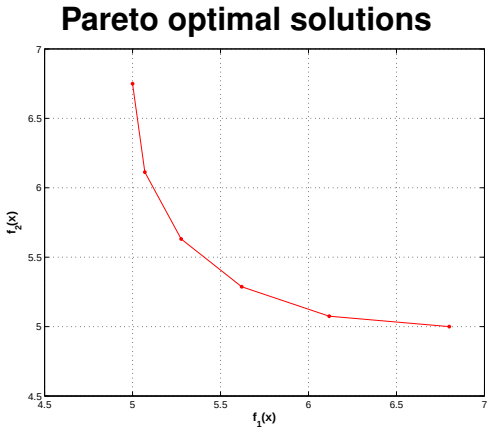



**Figure 2.5:** Pareto optimal solutions for the simple problem, the curve represents the trade-off between the two conflicting objectives of the problem

**domination of solutions** The approach of a multiobjective algorithm is based on domination of solutions rather then the use weighing functions. A solution is said to dominate an other solution if it performs better with regard to *all* objectives. Non-domination occurs when a candidate can not be improved any further in one objective while degrading in another objective. Figure 2.6 explains this in more detail. The figure shows that solutions in quadrant I are dominated by the solution $y(x_4)$. This solution is able to improve in both objectives with regard to the two solutions in green. The solutions in quadrant II and IV are non-dominated by $y(x_4)$.

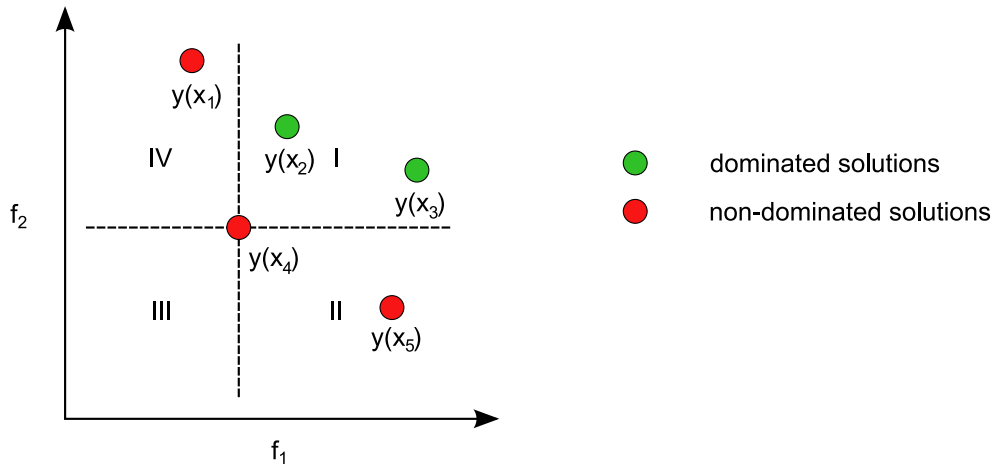



**Figure 2.6:** in red solutions that are part of the non-dominated set, in green solutions that are dominated by this set

Solutions that lye on a trade-off curve as shown in figure 2.5 are characterised by the fact that they are all non-dominating solutions. This set of non-dominated solutions is called a *Pareto front*. A multiobjective algorithm selects and improves solutions based on this domination principle to approximate the real trade-off curve. According to Deb [11], a good approximation should yield solutions that lay along the whole length of the front (including extremal solutions) and it should obtain an even spacing of solutions along this front.

## 2.4 Propeller design problem

As outlined in paragraph 2.2, the propeller design problem is a multiobjective problem that exhibits conflicting objectives and we want to approximate the Pareto front of this problem for three reasons.

**economical** As mentioned in the introduction, the Pareto front could tell us how much could be gained in terms of cavitation behaviour and what the price is that we pay in terms of a reduction in efficiency. Here the Pareto front is also an aid to the designer that gives additional information that can help him decide on a compromise between these two objectives.

**designer** The propeller design is currently done by an expert that makes decisions based on experience. It is unknown whether this expert is making a design that can not be improved upon any further with regard to *both* objectives. If the approximation of the Pareto front is good we could compare this to the design of the expert. Is there maybe a design that dominates the design of the expert?

**problem insight** To conclude, an analysis of the parameters can reveal information about which parameters play a role in forming this trade-off and which don't. As outlined in paragraph 2.2 we expect that the expanded blade area ratio plays an important role but to what extent?

The formulation of the problem is as follows. The objective functions in this thesis are defined as being efficiency and cavitation. How to judge the different forms of cavitation on a blade will be discussed later in paragraph 5.3.2. The design variables that describe a design are only related to the geometrical shape of the propeller. The shape of a sectional profile is not treated in this thesis but is an important part of the geometry that normally is taken care of in the detailed design. To conclude, there are a number of constraints that should be satisfied, mainly avoidance of bubble cavitation and assuring a minimum blade thickness from strength considerations.

# Chapter 3

# Theory

This chapter gives a review of the programs and methods that will be used in the design stage when the optimiser is applied to the propeller design problem. These programs are divided in propeller analysis, cavitation analysis and the optimization algorithm.

## 3.1 Propeller analysis

There are two common ways to analyse a propeller for its performance, by means of experiments and by a physical model trying to capture the physics behind the problem. We'll first discuss the experimental approach by the Wageningen B-Series but there are other frequently used series from for example Gawn [15] or the Newton-Rader series (Newton and Rader [26]). The advantage of using series data is that it is fast and accurate because the series are based on test data. The disadvantage however is that most series only contain a few parameters and is limited to the geometry that the series is able to describe.

### 3.1.1 B-Series propeller data

The Wageningen B-Serie is a collection of systematically varied propellers to predict the open water performance for use in (mainly) preliminary ship design. The series varies the pitch $P/D$, the expanded blade area ratio $A_E/A_0$ (also known as $EAR$) and the number of blades $Z$. The first experimental data of the series originates from Troost [31] and only contained 2 series of 4-bladed propellers. The series is extended in the years after and an overview of the complete series is given in table 3.1.

**Table 3.1:** Overview of the Wageningen B-Series.

| Z\EAR | 0.30 | 0.35 | 0.38 | 0.40 | 0.45 | 0.50 | 0.55 | 0.60 | 0.65 | 0.70 | 0.75 | 0.80 | 0.85 | 1.00 | 1.05 | P/D |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | X | | X | | | | | | | | | | | | | 0.6 - 1.4 |
| 3 | | X | | X | | X | | | X | | | X | | | | 0.5 - 1.4 |
| 4 | | | | X | | | X | | | X | | | X | X | | 0.5 - 1.4 |
| 5 | | | | | X | | | X | | | X | | | | X | 0.5 - 1.4 |
| 6 | | | | | | X | | | X | | | X | | | | 0.5 - 1.4 |
| 7 | | | | X | | | X | | | X | | | X | | | 0.5 - 1.4 |

The open water characteristics of the series is represented in performance charts but is since 1969 available in polynomial form from van Lammeren et al. [32]. Kuiper [21] gives a good overview of the complete B-Series and its updates. The polynomial form of the thrust and torque coefficients are described in the following way:

$$\begin{aligned} K_T =& f_1(J, P/D, A_E/A_0, Z, Re) \\ =& \sum_i C_{t1,i}\, J^{st1,i}\, (P/D)^{tt1,i}\, (A_E/A_0)^{ut1,i}\, Z^{vt1,i} \\ &+ \sum_j C_{t2,j}\, J^{st2,j}\, (P/D)^{tt2,j}\, (A_E/A_0)^{ut2,j}\, Z^{vt2,j}\, (\log_{10} \mathrm{Re} - 0.301)^{wt2,j} \end{aligned} \tag{3.1}$$

$$\begin{aligned} K_Q =& f_2(J, P/D, A_E/A_0, Z, Re) \\ =& \sum_i C_{q1,i}\, J^{sq1,i}\, (P/D)^{tq1,i}\, (A_E/A_0)^{uq1,i}\, Z^{vq1,i} \\ &+ \sum_j C_{q2,j}\, J^{sq2,j}\, (P/D)^{tq2,j}\, (A_E/A_0)^{uq2,j}\, Z^{vq2,j}\, (\log_{10} \mathrm{Re} - 0.301)^{wt2,j} \end{aligned} \tag{3.2}$$

where $J = V_A/(nD)$ is the advance coefficient and Re the characteristic Reynolds number at $0.75R$. The Reynolds number at that particular section is defined as:

$$\mathrm{Re} = \frac{c_{0.75R}\sqrt{{V_A}^2 + (0.75\pi n D)^2}}{\nu} \tag{3.3}$$

here $c$ is the chord length at that section, $V_A$ the undisturbed inflow velocity, $n$ the rotation rate in rps and $\nu$ the kinematic flow viscosity. The propeller open water efficiency is the ration between $K_T$ and $K_Q$ and can be determined according to:

$$\eta = \frac{J}{2\pi}\frac{K_T}{K_Q} \tag{3.4}$$

### 3.1.2 Propeller Analysis ANPRO

ANPRO consists of three theoretical models (see Assenberg et al. [3]) to calculate the characteristics of a propeller. These models are a lifting surface model, a model to incorporate the thickness of the profile and a model to calculate the length of the sheet cavity on the propeller blades. The first two models are described here shortly, the cavitation model is described at page 19 in the paragraph on cavitation models.

**Lifting surface model** The lifting surface model of ANPRO consists of a distribution of pressure dipoles and a vortex sheet that are both placed on a helicoidal plane with a constant pitch. The distribution of the dipoles is over the camberline and are described by so called mode-functions. The unknown coefficients of these functions are solved for by applying the boundary conditions. These conditions are applied at the so called collocation points and they require that the normal velocities at these points are zero.

Once the strength of the pressure dipoles is known, the loading at the geometrical camber line is also known in the sense of a pressure jump across the camber line. Figure 3.1 shows the forces acting on a flat plate where $F_{PRESS}$ is the force that results from the integrated pressures over the camber line. On the leading edge there is normally a suction force due to the very low pressures that are present there. This force can be determined using the condition that the lift force should be perpendicular to the inflow velocity and that there is equilibrium among these forces.

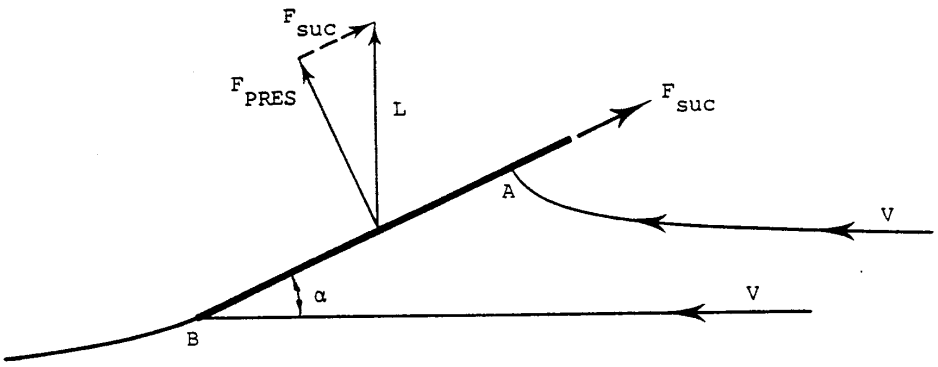



**Figure 3.1:** simplified forces on a flat plate under an angle of attack

**Blade thickness model** The cavitation module requires the pressures on the blade as input parameters. It is therefore important to determine these pressures correctly and to do so, the blade thickness should be taken into account. To pressures determined by the dipoles are the average pressures present on the profile, indicated in figure 3.2 by the mean value line.

Because the flow over the propeller section is three dimensional, the camber line is first corrected to an effective camber line. The pressures on both suction and pressure side are now determined by a conformal mapping method and are then added or subtracted from the mean value. This is a 2D analysis and is repeated for the different radii along the radius of the propeller blade.

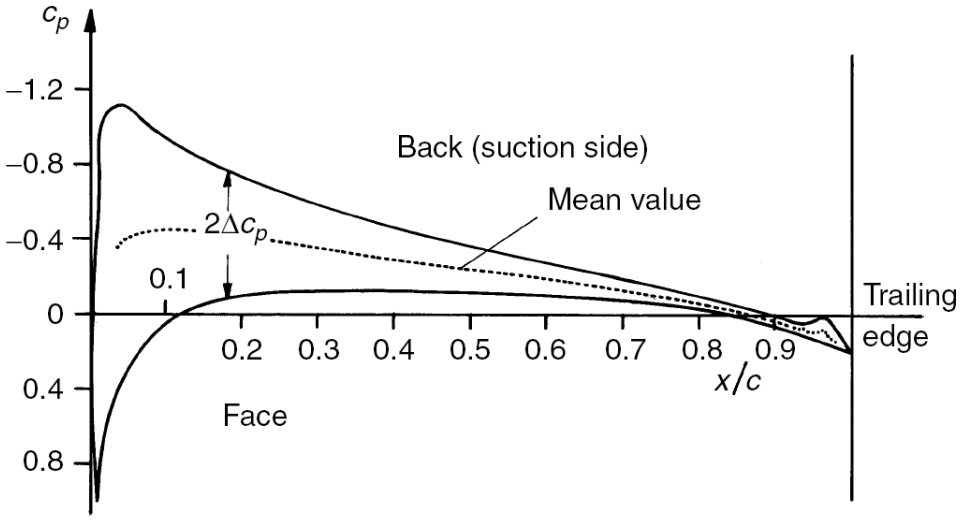



**Figure 3.2:** pressure distribution on a profile section of a propeller blade

## 3.2 Cavitation models

When a fluid is subjected to a drop in pressure or a rise in temperature close to the evaporation limit, it will change phase and vapourises. This phenomena occurs when the actual pressure drops below the vapour-pressure or the temperature rises so that the ambient pressure becomes equivalent to the vapour-pressure, small cavities (bubbles) are then formed in the fluid. On propellers, the pressure on a blade will drop due to high suction peak at the leading edge of a sectional profile which can result in sheet cavitation. If the pressure drops halfway the chord of a sectional profile, bubble cavitation is most likely to happen. These are two different types of cavitation and we can distinguish four different types of cavitation on a propeller blade, Carlton [8] gives the following overview:

**Sheet cavitation** is caused by the high suction peak occurring at the leading each of a profile when this profile is not working at a shock free angle of attack. The suction peak of low pressure initiates cavitation and depending on ambient conditions like angle of attack and local pressure, the cavitation extends into a sheet.

**Cloud cavitation** mainly occurs behind a breaking up sheet cavitation. The sheet develops into a cloud of very small bubbles and hence the name.

**Bubble cavitation** forms when the local pressure is below the vapour pressure for an extended period of time so that nuclei in the flow has time to grow into a bubble. If the local pressure rises again above the vapour pressure, the bubbles will collapse with the risk of causing damage (pitting which is also referred to as erosion).

**Vortex cavitation** follows from the vortices shed from the propeller. These vortices are present as they're inherent to lift generating bodies and are the strongest at the tip

and the hub of the blade where there are large gradients in the circulation distribution. The low pressure in the core is responsible for the initiation of the cavitation.

Figure 3.3 gives an overview of the different forms of cavitation.

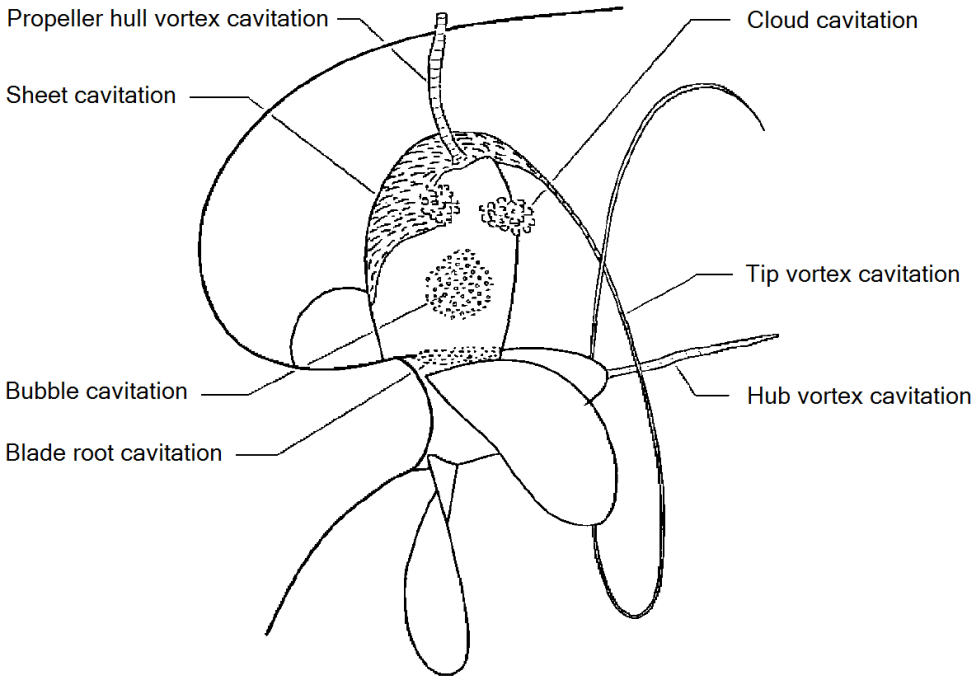



**Figure 3.3:** overview of the different types of cavitation that can be observed on a propeller

The occurrence of cavitation is not only dependant on the pressure but also on the number of particles (nuclei) and the amount of gas that is dissolved in water. The dimensionless parameter that estimates the likelihood that cavitation will occur is defined as:

$$\sigma = \frac{p_0 - p_v}{1/2\rho {V_0}^2} \tag{3.5}$$

where $p_0$ is the ambient pressure, $V_0$ a corresponding reference speed and $p_v$ is the local vapour pressure. The cavitation number for which the local pressure is lower than the vapour pressure is called $\sigma_v$. Cavitation now occurs when $\sigma > \sigma_v$.

### 3.2.1 Burrill's cavitation chart

In the design of a propeller, not all cavitation needs to be avoided because not all types of cavitation are regarded as harmful. As cavitation is (still) difficult to asses, there are not so many aids for the designer to assist in the design of a propeller. Bubble cavitation

is in general avoided as this is regarded as being erosive. Sheet cavitation is allowed to a certain extent on the suction side and is in most cases only allowed for a few percent on the pressure side of the blade.

One of the early aids were the design charts of Burrill and Emerson [6] which give an indication of the amount of cavitation that could be observed at the propeller blade. Burrill gives an empirical relationship of data that is based on observations in a cavitation tunnel with uniform flow. His chart is based on the propeller loading and the cavitation number which are given in 3.6 and 3.7. The chart is shown in figure 3.4.

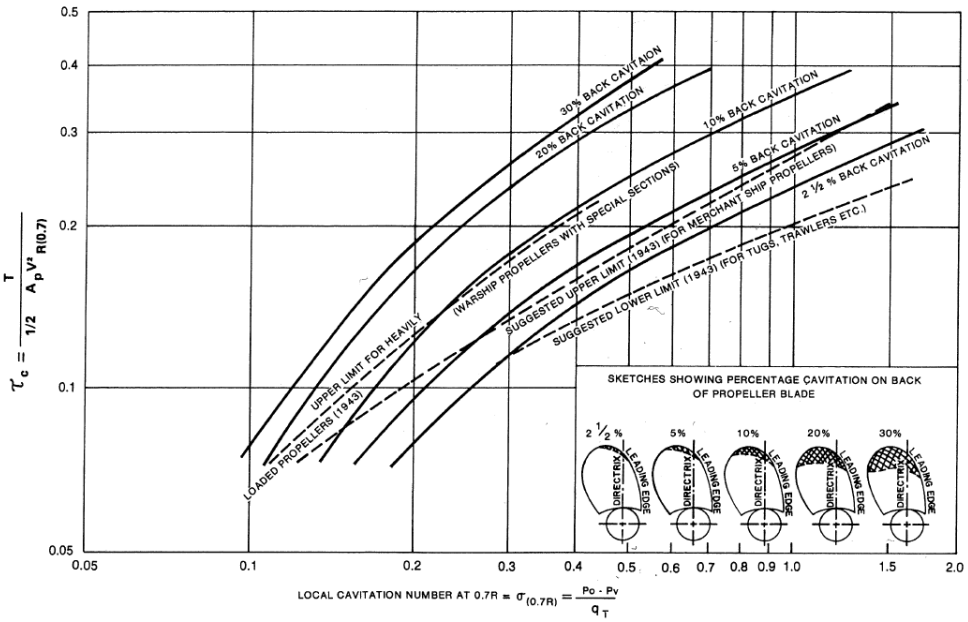



**Figure 3.4:** Simple cavitation diagram from Burrill

$$\tau_c = \frac{T}{1/2\rho A_p (V_{0.7R})^2} \qquad \text{(propeller loading)} \tag{3.6}$$

$$\sigma_{0.7R} = \frac{p_0 - p_v + \rho g h}{1/2\rho (V_{0.7R})^2} \qquad \text{(cavitation number)} \tag{3.7}$$

$$\text{with} \quad (V_{0.7R})^2 = {V_A}^2 + (\pi n 0.7D)^2$$

From this data, Keller [20] made a derivation to give an estimate of the minimum required

area to avoid cavitation. Keller's formula is given in 3.8

$$A_E/A_0 = \frac{(1.3 + 0.3Z)T}{(p_0 + \rho gh - p_v)D^2} + 0.20 \tag{3.8}$$

### 3.2.2 ANPRO cavitation model

The cavitation module in ANPRO is more advanced and tries to model two different types of cavitation instead of giving a percentage like Burrill does. The modelling of sheet cavitation and bubble cavitation is implemented in ANPRO.

The local pressures as calculated by the lifting surface model from ANPRO is used to determine where the local pressure is lower then the vapour pressure. If this occurs in the first 2.5% of the chord length, sheet cavitation is predicted. If this occurs after 10% length of the chord, bubble cavitation is predicted. The prediction of sheet cavitation is based on the theory by Geurst [16]. This theory neglects the effect of profile thickness and is two dimensional. There are two formulations for the cavity length and distinction is made between two types; when the cavity is shorter than the blade chord and when it extends over the blade, see figure 3.5.

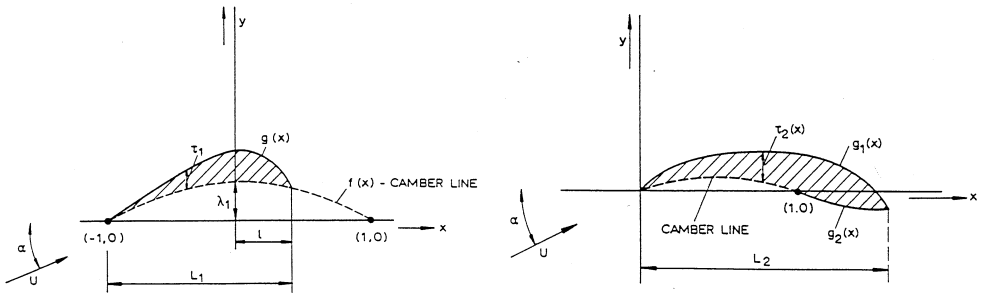



**Figure 3.5:** the two types of sheet cavitation that are modelled by the theory of Geurst, left a sheet that stays on the blade, right a sheet that extends from the blade

Bubble cavitation is predicted for al places on the chord where the local pressure is lower than the vapour pressure. In terms of the local pressure coefficient, this is where $-C_{pmin} > \sigma$.

## 3.3 Optimisation Algorithm

As is mentioned in paragraph 2.4 we want to use an optimisation algorithm that is capable of approximating the Pareto front. In this section such an algorithm is discussed which happen to be an Evolutionary Algorithm (EA). First a general introduction of an EA is given followed by an overview of algorithms that are capable of dealing with multiple objectives. Finally the algorithm that is picked to approximate the Pareto front is explained in more detail.

### 3.3.1 General Outline of an EA

An Evolutionary Algorithm is a generic population-based metaheuristic optimisation algorithm or search algorithm. Ashlock [2] gives the basic steps that will make up any EA, see figure 3.6. There are many ways to fill in these steps, depending on the problem at hand but they all exhibit application of biological theories like reproduction, mutation, recombination, natural selection and survival of the fittest.

```
Generate a population of structures
Repeat
    Test the structures for quality
    Select structures to reproduce
    Produce new variation of selected structures
    Replace old structures with new ones
Until Satisfied
```

**Figure 3.6:** basic loop of the steps that make up an Evolutionary Algorithm

The term Genetic Algorithm (GA) is commonly used to denote the same but actually is a popular sub-class of EA. A GA refers to the *structure* representing the individuals in a population to be of a genetic type. In an optimisation problem, the genomes are represented by bits and operations like crossover and mutation are easily implemented on this type of structures. Optimisation problems in most cases not only consists of binary valued variables but also of real valued variables and a different implementation of crossover and mutation has to be sought.

**purpose of each step** Each step identified in the general loop of an EA serve a certain purpose, some of these are explained a little further.

**generate** a population should start somewhere. In most cases the population is initiated based on a random process but it is also possible to start with an already existing population from a previous run.

**test** individuals that make up a population need to get a fitness value assigned. These are required to evaluate an individual by a selection or replacement operator. In an engineering problem, assigning a fitness value is similar to evaluating a design problem for its objective functions.

**selection** the purpose of selection is to obtain *on average* a bias to the more fit solutions to generate offspring from and to discard the worst solutions. The fitness value is used to distinguish between these solutions.

**produce** the selected individuals are subjected to *variation* to produce new offspring. The two possible operations that can introduce variation are *mutation* and *crossover*. Mutation is an operation that works on an individual, crossover operates on two individuals or more. Variation is important to make the algorithm able to *search* its way through the domain of feasible solutions.

**replace** the offspring can completely replace the previous generation but a mix of both the old generation and the offspring can also make up a new generation. A selection procedure is applied again to determine which individuals will be part of the new generation and which don't.

**diversity** The aforementioned steps will iterate in a loop until a certain amount of iterations are passed or until an optimal (set of) solution(s) is found according to certain criteria. Variation plays an important role in finding this optimum. Reproduction operators like crossover and mutation introduce variation into a population, selection takes out part of this variation and there should be a good balance between these two. If there is not enough variation the algorithm is not able to find the *global* optimum but will probably get stuck in local optima. If there is too much variation the algorithm behaves rather erratic, it is able to find the global optimum but it isn't able to converge to it.

### 3.3.2 Overview of Multiobjective Evolutionary Algorithms

There are a number of multiobjective optimization algorithms (MOEA) available and all are able to approximate the Pareto front with good quality and Kunkle [23] gives an good overview of the different algorithms available. A number of these algorithms stick out above the rest, the most popular ones being PESA II (Corne et al. [9]), NSGA-II (Deb et al. [14]) and SPEA2 (Zitzler et al. [34]). They all incorporate crossover, selection, mutation and replacement but they differ in the way they deal with multiple objectives and the preservation of diversity in the solutions.

**PESA II** Pareto Envelope-based Selection Algorithm, selection in this algorithm is based on a hyperbox instead of individuals. The front is divided in regions called hyperboxes and can contain a number of individuals. When a box is selected, an individual is randomly selected from that box. The idea is to prevent clustering of individuals by selecting only one from a box.

**SPEA2** Strength Pareto Evolutionary Algorithm, this algorithm keeps an archive of optimal solutions that it has found so far. Newly created individuals are compared with this archive for domination and not with other new individuals that are just created. Furthermore, the algorithm uses the number of individuals it dominates or is dominated by in the selection procedure to assure diversity.

**NSGA-II** Non-dominated Sorting Genetic Algorithm, this algorithm sorts the population based on non-dominated fronts. The first front found is ranked the highest and the last one the lowest. This ranking is used in the selection process. In addition to this a crowding distance measure is used to assure diversity in a population.

In this thesis the NSGA-II algorithm is used mainly because of its relative simple way to implement it in Matlab and the availability of the source code. A more detailed description of the algorithm is given in the next section.

### 3.3.3 NSGA-II overview

The algorithm contains the basic parts that were given in figure 3.6 but it distinguish itself in the way it deals with selection and replacement to cope with multiobjectivity. NSGA stands for *Non-dominated Sorting Genetic Algorithm* and this characteristic can be found in the selection and replacement part of the algorithm. The main steps as identified earlier are now explained in more detail.

**test** In this part not only the objective functions (including the constraints) are evaluated but also the crowding distance is calculated. This distance is used to measure the diversity of the solutions along a front of non-dominated solutions. It is a measure how close neighbouring solutions are, see figure 3.7. In this figure the crowding distance for solution $y(x_4)$ is calculated relative to the solutions from the same front which are all colored red. The distance is the sum of the length and width of a cubical that can be drawn through these two closest neighbours.

**selection** The selection of the parents to generate the offspring is based on a tournament selection. Two pairs of individuals are selected each time and the most fit of each pair is selected to produce offspring, see figure 3.8. To determine which individual is the most fit, there are two criteria to check for: feasibility and the Pareto front.

First of all, the feasible individual is always preferred above the infeasible one. If both solutions are feasible then the individual is selected based on the highest Pareto front. If both individuals belong to the same front, selection is based on crowding distance. For individuals that are both infeasible, selection is based on how severe constraints are violated. Figure 3.9 explains this ordering of individuals.

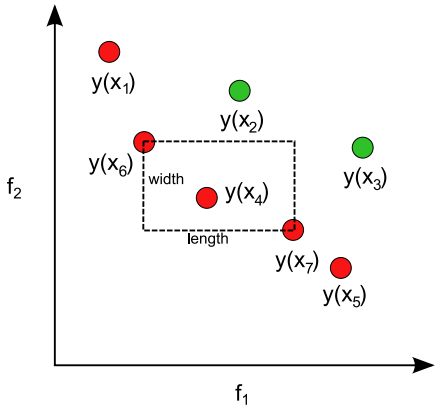



**Figure 3.7:** crowding distance measure

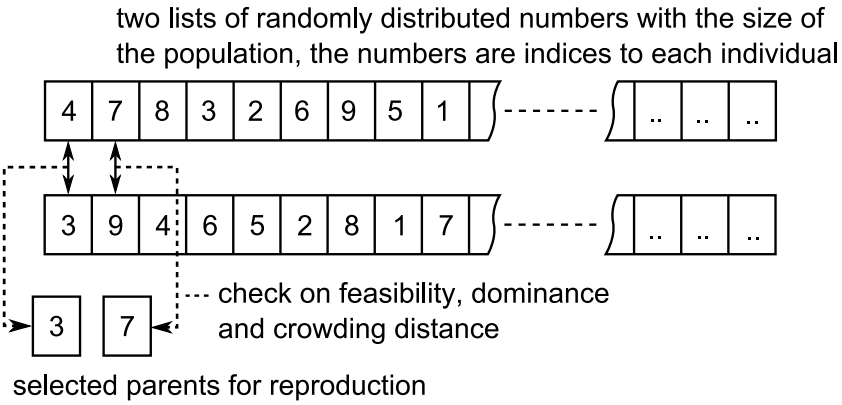



**Figure 3.8:** selection procedure

**crossover and mutation** These two operations are responsible for the generation of new offspring. On the two parents just selected, crossover takes place on every variable of the parents and two children are created. The crossover of every variable occurs with a certain probability, the crossover probability. This probability is related inversely to the number of variables. After crossover, mutation takes place on every variable of the created children. The probability for mutation to occur is also inversely proportional to the number of variables.

In case of a binary valued variable - like an integer which is represented by a number of bits - crossover takes place by swapping part of the bits from one parent with the bits of the other parent. This is similar to what happens in biology with the genes of a section of DNA, see figure 3.10.

For real valued variables this is not going to work because the coding of the bits for a real valued variable is completely different. The crossover is now formed by a probability distribution which is based on binary crossover. It is therefore called *Simulated Binary*

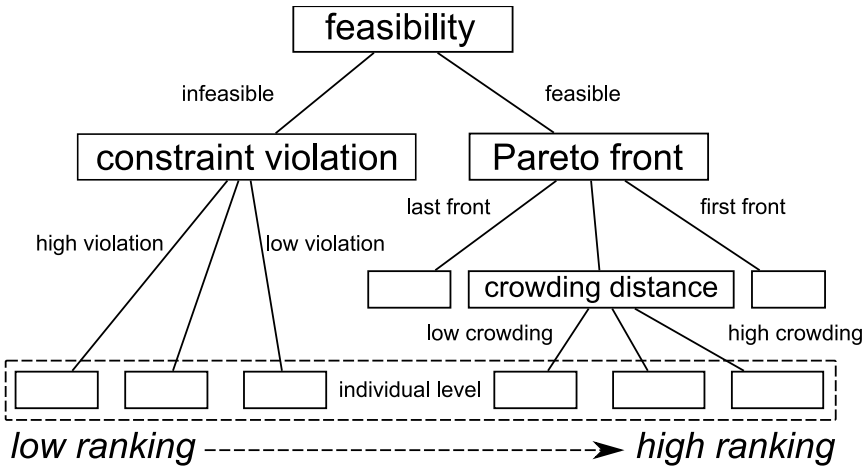



**Figure 3.9:** global overview of the sorting of the individuals into ranks

*Crossover* or SBX, see Deb and Agrawal [12]. The probability to generate an offspring close to one of the parents is higher than far away but is not zero. This makes the algorithm able to generate completely different individuals and gives it the ability to *search* outside its pool of already found solutions. By doing so it increases the possible chance of finding even better individuals and it avoids getting stuck in local optima.

Mutation adds a random disturbance to a variable and is done by flipping the value of one of the genes (bits). The point where this occurs is randomly chosen and is called single-point mutation. Multiple-point mutation consists of more than one point for mutation but is not applied here. For real valued variables, the same problem exists as with crossover that the coding of the bits doesn't map with the real valued variable anymore. A polynomial probability distribution is applied to introduce mutation. Mutation only brings in small variation which is close to its ancestors. It is therefore seen as a local search mechanism that helps the algorithm to converge better to the optimum once it is near it.

**replacement** The selection algorithm is based on the sorting explained in 3.9 and first merges the parent population and the offspring that is just created, see figure 3.11. This merged population is then sorted in Pareto fronts. All the fronts are incrementally added until the normal size of a population is reached. To stay within this fixed population size, the last front is split according to the crowding distance measure.

Constraints are taken into account by an added step in the sorting algorithm, see figure 3.12. In case there are only individuals that violate one or more constraints, the new population is filled according to how severe a constraint is violated. If there are only feasible solutions, the population is filled according to figure 3.11.

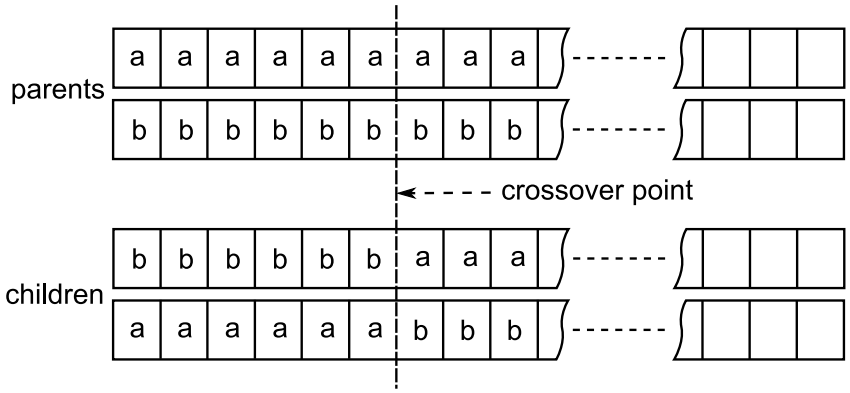



**Figure 3.10:** 1 point crossover of a binary variable (integer), the letters *a* and *b* represent the genes or bits that make up a variable

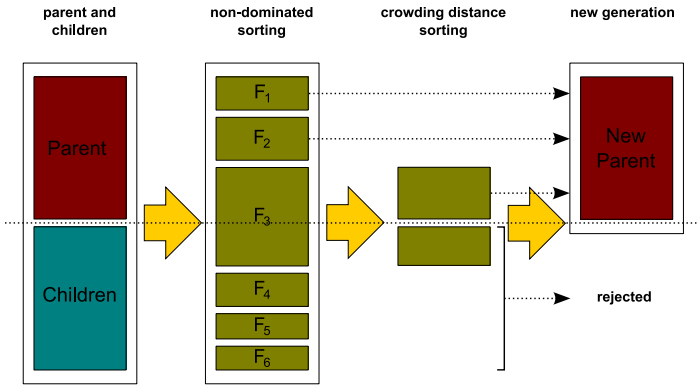



**Figure 3.11:** replacement procedure

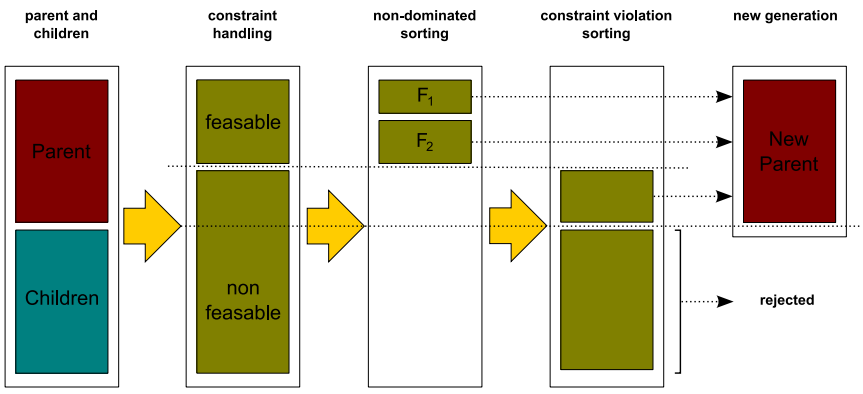



**Figure 3.12:** replacement procedure including constraint handling

# Chapter 4

# Propeller parameterization

This chapter gives a short overview of the nomenclature that is used to describe the propeller geometry. After that the propeller parametrization is discussed and the chapter is concluded with a description of some typical characteristics that result from the propeller geometry.

## 4.1 Propeller nomenclature

The ITTC [19] gives a clear definition of the propeller geometry. The parts that are relevant in this thesis are stated here again for completeness. Figure 4.1 gives a sketch of a propeller and the definition of pitch. The geometrical pitch is defined as the helix generated by the nose-tail line also named the chord line.

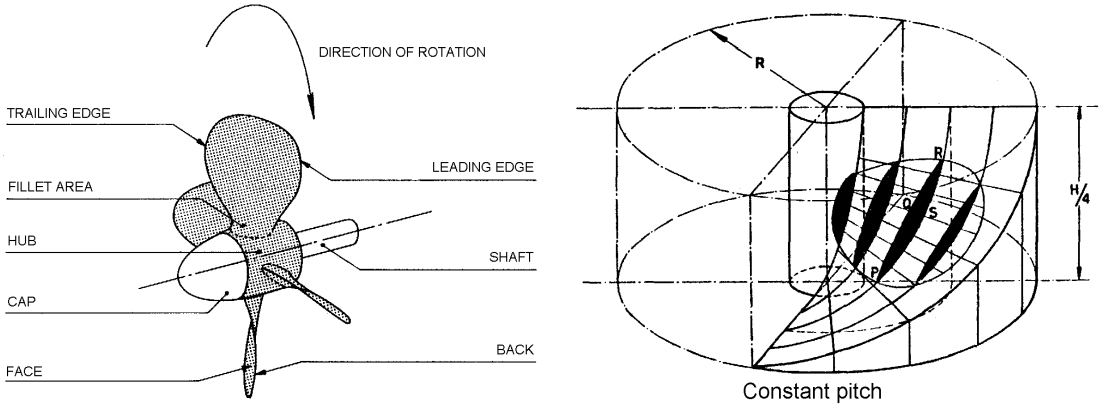



**Figure 4.1:** sketch of a propeller (left) and definition of pitch (right)

The propeller is build up of profile sections along the radius. These sections have a certain chord, camber and thickness (see figure 4.2 on the left).

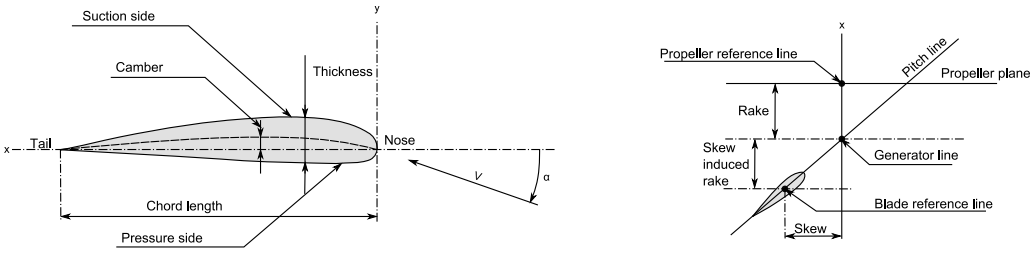



**Figure 4.2:** geometry of a propeller blade section (left) and a cylindrical cross section (right)

The chord length, pitch, camber and thickness can change over the radius and is then referred to as a distribution. When a section is shifted in axial direction (this is downwards on the right in figure 4.2) it is referred to as rake. If the section is shifted in the direction of the helix it is referred to as skew. Skew can also be defined as an angle and is then the angular displacement along the longitudinal propeller axis. The undisturbed inflow relative to the nose-tail line is called the angle of attack $\alpha$.

Two additional lines are defined to introduce a coordinate system to be able to relate the position of a blade section in space. The first one is the *generator line* which is formed by the intersection of the pitch helices and the plane containing the shaft axis and the propeller reference line. The second one is the *blade reference line* which is the connection of the locus of the blade reference points, see figure 4.2 on the right.

## 4.2 Parameterization

All the above mentioned parameters both have an influence on efficiency and cavitation performance. Skew has a negligible effect on performance and mainly has its influence on cavitation behaviour. The idea behind skew is that not all blade sections pass the wake peak at the same time and in this way influences the form and shape of (mainly) the sheet cavitation.

The profile sections also have a great influence on cavitation behaviour and optimizing them is proofed to be worthwhile, see for example Zondervan, Gert Jan and Holtrop, Jan [36], Dang [10] and Kuiper and Jessup [22]. In this thesis the sections are not optimized but are fixed to a profile with a NACA 65 mean line and a NACA 66 modified thickness distribution.

### 4.2.1 Bézier curve

A Bézier curve is a spline based on Bernstein polynomials and can be used for smoothing and interpolation. A cubic Bézier curve, see figure 4.3, is used to describe the different distribution functions. The control points $P_0$ to $P_3$ control the shape of the curve and are used to indirectly change the distribution shape. With four control points a Bézier curve

offers enough freedom to describe any smooth shape distribution and limits the amount of parameters needed to change the shape of the curve to two or three.

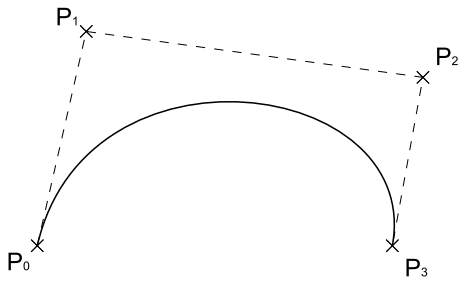



**Figure 4.3:** cubic Bézier curve with four control points

### 4.2.2 Camber and Pitch distribution

To describe the geometry of a propeller in terms of distributions for the pitch and the camber (but also the thickness and the skew), the shape of these distributions need to be (preferably separately) controlled in the following ways: shifting up and down, rotating left and right, flat or curved and location of the hump of the curve. Figure 4.4 shows these four possible transformations implemented with a Bézier curve for both the pitch and the camber. The only difference is that the camber is scaled (shifted up/down) differently, see table 4.1.

### 4.2.3 Chord distribution

The shape of the blade should be able to change from completely straight to a curved distribution. There are only two parameters that control the *shape* of the distribution. The blade area is controlled by a third parameter that is multiplied with the shape to give the final chord distribution, the scaling. Figure 4.5 gives an overview of the three possible ways to change this distribution between its extremes. To obtain the final chord length, the distribution is multiplied by the diameter and divided by the number of blades.

### 4.2.4 Thickness

Lloyds Register gives rules for a minimum thickness to comply with strength requirements, see Lloyd's Register of Shipping [25]. For a fixed pitch propeller, the minimum thickness at 0.25R and 0.6R are given by the Rules. The minimum thickness at the tip follows from practical reasons like protection against backing and is mostly taken as 0.0035 times the diameter of the propeller.

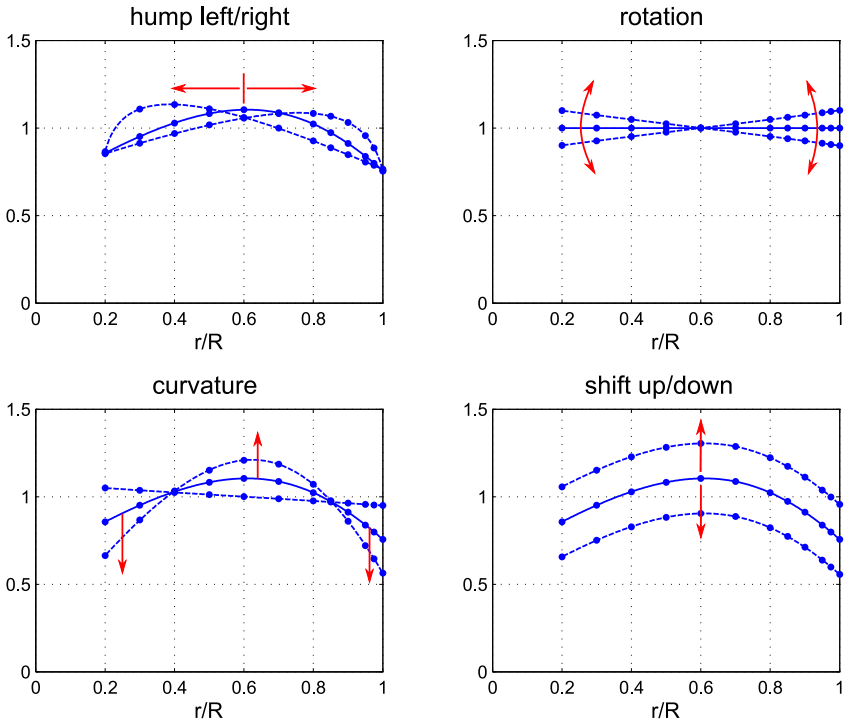



**Figure 4.4:** four possible transformations applicable to a shape distribution like with the pitch distribution and the camber distribution

The minimum thickness found for these 3 radii are in most cases a straight line. This line is taken as the minimum and added to this is a distribution which allows for a thicker tip, a thicker root or both. Secondly, it is possible to deviate from a straight line and add a curved distribution on top of it.

### 4.2.5 Skew

The skew distribution is controlled by only one parameter. The distribution is taken as a curve through 0.15 r/R and 0.7 r/R. This curve is then multiplied by a parameter that ranges between zero (no skew) and 3 (maximum skew), see figure 4.7. The skew defined according to Lloyd's Register, see the same figure, is then calculated as this is required in the determination of the minimum thickness. Propellers with more skew exhibit higher stresses and therefore require more thickness.

### 4.2.6 Overview of parameters

Table 4.1 gives an overview of all the parameters involved in describing the propeller geometry, 15 in total. Not all parameters could be divided as described in paragraph 4.2.2.

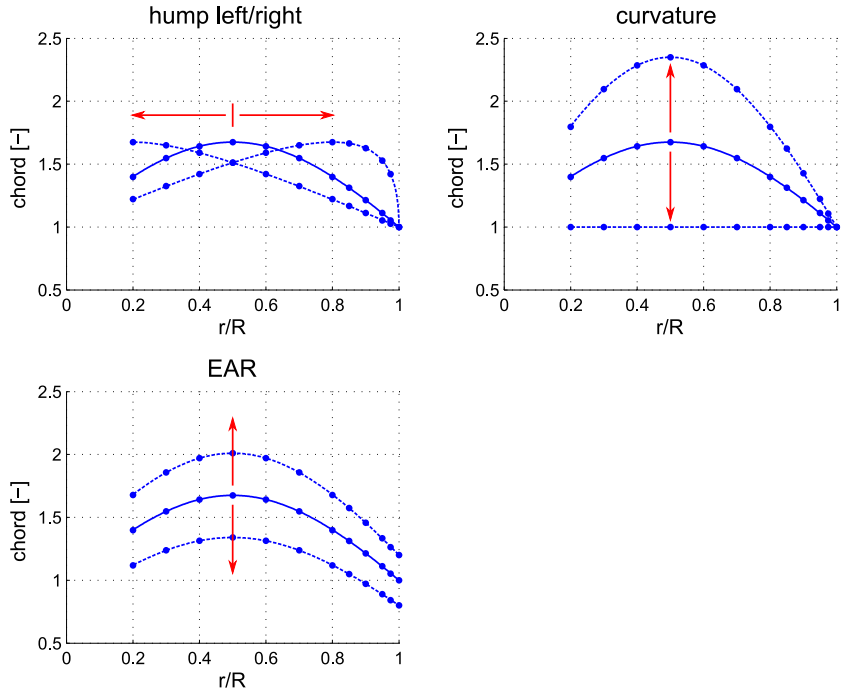



**Figure 4.5:** three different ways to change the chord distribution

The chord and thickness distributions can not be rotated and the latter one even has two parameters controlling the scaling at two different points along the radius.

**Table 4.1:** Overview of different parameters and their range that control the different distributions

| | hump left/right | rotation | curvature | shift up/down |
|---|---|---|---|---|
| pitch | [-0.3 .. 0.3] | [0 .. 0.2] | [0 .. 0.8] | [0.8 .. 1.2] |
| chord | [0 .. 0.8] | - | [1 .. 2.8] | [0.8 .. 1.2] |
| camber | [-0.3 .. 0.3] | [0 .. 0.2] | [0 .. 0.8] | [0 .. 0.06] |
| thickness | - | | [0 .. 0.4] | [0 .. 0.5] and [0 .. 3] |
| skew | - | - | - | [0 .. 3] |

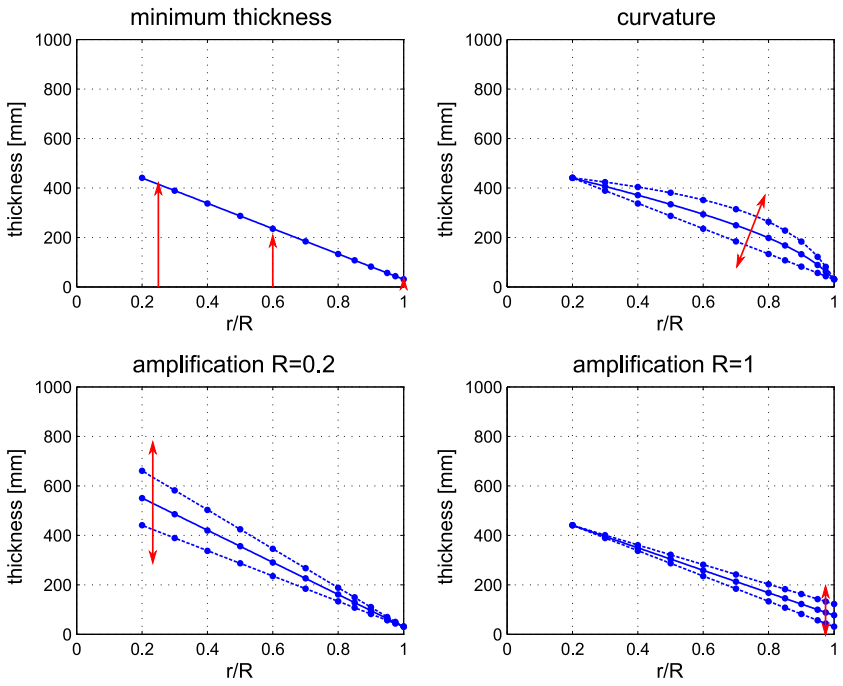



**Figure 4.6:** minimum thickness according to LR (upper/left) and three possible ways to transform the thickness distribution

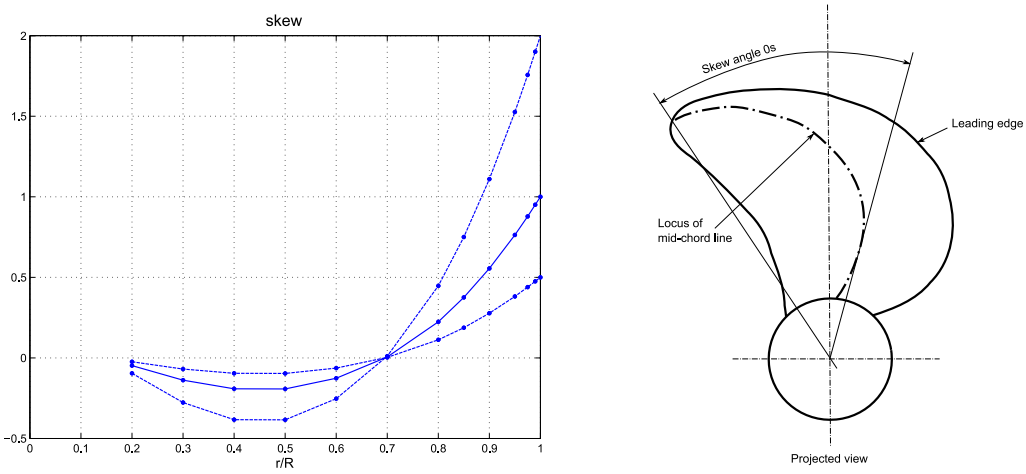



**Figure 4.7:** scaling of skew (left) and definition of skew according to LR (right)

## 4.3 Relevant propeller geometry parameters

Once the distributions are made for the different parameters describing the propeller geometry, some characteristics like nominal and virtual pitch and $A_e/A_0$ are not input parameters anymore but follow from these distributions and have to be determined afterwards. A sketch of a sectional profile is given in figure 4.8 and introduces the variables that describe chord, pitch, the nominal pitch angle and the zero lift angle.

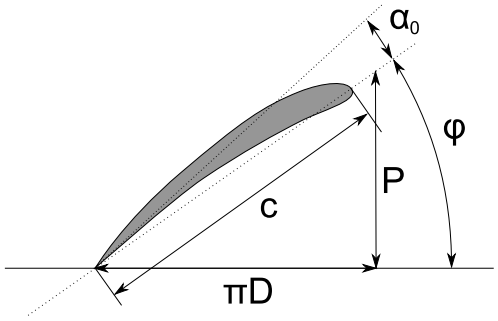



**Figure 4.8:** coordinate system on a propeller profile section

### 4.3.1 Expanded Blade Area Ratio

The area of one blade is calculated by integrating the chord over the radius. The chord is made dimensionless by dividing it by the diameter of the propeller. This integration, see formula 4.1, is done by using Simpson's rule.

$$A_e/A_0 = \frac{0.5Z \int_{0.2}^{1} \left(\frac{c}{D}\right) dr}{0.25\pi} \tag{4.1}$$

### 4.3.2 Average Nominal Pitch

The average nominal pitch is calculated by integrating the local pitch over the radius. The pitch is made dimensionless by dividing it by the diameter. Formula 4.2 is integrated using Simpson's rule.

$$P_g = \frac{\int_{0.2}^{1} \left(\frac{P}{D}\right) r dr}{\int_{0.2}^{1} r dr} \tag{4.2}$$

$$\text{with} \quad P = \pi r D \arctan(\varphi)$$

### 4.3.3 Average Virtual pitch

The virtual pitch is defined as the volumetric-average zero-lift pitch of the propeller blade. To calculate this pitch, the zero-lift angle $\alpha_0$ at every section has to be determined (see figure 4.8). If we assume "thin-profile-theory" then the general expression for the lift is given by formula 4.3, see Anderson [1] for a more elaborate deduction. From this formula the zero-lift angle follows as given in formula 4.4. After a transformation back to the Cartesian coordinate system, the integral is changed to an integration over the chord of the slope of the camber-line.

$$C_L = 2\pi \left[ \alpha + \frac{1}{\pi} \int_0^{\pi} \frac{dy}{dx} (\cos\theta_0 - 1) d\theta \right] \tag{4.3}$$

$$\alpha_{L=0} = -\frac{1}{\pi} \int_0^{\pi} \frac{dy}{dx} (\cos\theta_0 - 1) d\theta_0 \tag{4.4}$$

For the special case that the camber-line is a parabolic distribution with the maximum camber at half the chord, the integral simplifies to $\alpha_0 = -2f/c$ which means that we don't have to evaluate the integral anymore to determine the zero-lift angle, see figure 4.9.

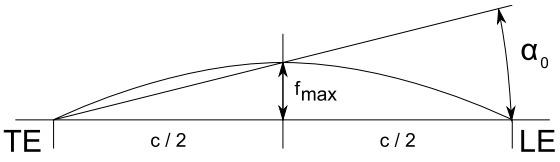



**Figure 4.9:** zero lift angle for a parabolic camber-line

This zero-lift angle is valid for the assumption of invicid flow. To account for the effect of viscosity, Burrill gave a correction factor. A polynomial expression for this factor is given

by Oossanen [27]. The polynomial expression to correct the zero-lift-angle is given in 4.5 in which $f_x$ is the maximum camber, $t$ is the maximum thickness of the profile and $c$ is the chord-length. This expression is taken from Holtrop [17].

$$\begin{aligned} K(\alpha_0) = &0.972 - 0.169\,(t/c) - 2.78\,(t/c)^2 + 21\,(f_x/c)(t/c)^2 \\ &- (f_x/c)^2\{0.32 + 278\,(t/c)^3\} - (f_x/c)^3\{28.7\,(t/c)^2 - 335\,(t/c)^3\} \end{aligned} \tag{4.5}$$

Now the virtual pitch can be determined by integration of the zero lift angles over the radius of the propeller. The integral as given in 4.6 is again evaluated by applying Simpson's rule.

$$P_{vg} = \frac{\int_{0.2}^{1} \left(\frac{P_v}{D}\right) cr^2 dr}{\int_{0.2}^{1} cr^2 dr} \tag{4.6}$$

$$\text{with} \quad P_v = \pi r D \arctan(\varphi + \alpha_0)$$

# Chapter 5

# Implementation

In this chapter, the outline of the software is discussed. It consists of the optimiser and the objective functions and are two stand alone parts, see figure 5.1 for an overview. This means that they can both be implemented and tested separately. The optimiser is tested with a number of test functions and the objective functions are tested by running a systematic variation of a certain design parameter.

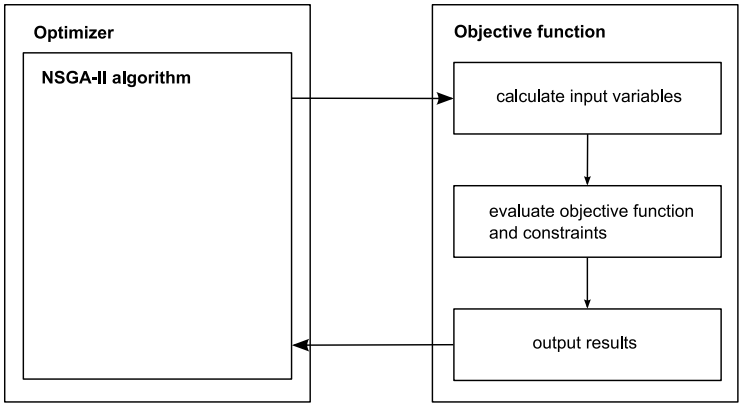



**Figure 5.1:** overview of the software architecture, left the optimiser (NSGA-II algorithm) and right the objective function

The chosen environment to implement this architecture in is Matlab. Arguments that lead to this choice are given in the definition study by Pouw [28]. There were two main important arguments. The first one is that it is preferable to have one environment to implement the architecture in so that it is easier to connect the different parts. The second argument was that familiarity with Matlab was already present.

## 5.1 Optimiser

The background of the optimization algorithm is already discussed in section 3.3.3. The implementation of an algorithm from literature is always difficult and it is preferred to have the source code available. Unfortunately this was only available in C/C++ (see http://www.iitk.ac.in/kangal/codes.shtml) but not in Matlab. The implementation of the code in Matlab is almost a direct port of the original C-code and uses the same naming of variables and functions. To check the ported code against the original code, different test functions are run. The two test functions that will be shown here are an unconstrained and an constrained problem and are both a minimisation problem.

So far the code is complete except for the handling of binary variables. The propeller design problem can be treated as a real valued problem if the only binary variable is left out of the optimization problem. This means that the number of blades can not be an optimization variable.

The next two sections discus the results of two test problems which are commonly used in the literature. The paper from Zitzler et al. [35] gives an overview of commonly used test problems. Almost all new algorithms are tested against each other using these problems. The advantage of these test problems is that the front with Pareto optimal solutions is known. The population size is taken as 80 and is lower then the default setting of 100. This is done to reduce the computation time slightly without risking the search capabilities of the algorithm. The settings for the mutation and crossover probability are directly taken from the original paper of Deb et al. [14].

### 5.1.1 Unconstrained problem - ZDT6

This test problem is given in formulae 5.1 through 5.4 and is characterized by an non-uniform spread of the solutions among its front. The problem tests for convergence to this front and the ability of the algorithm to come up with solutions that are evenly spread among this front.

$$f_1(\mathbf{x}) = 1 - \exp^{-4x_1} \sin(6\pi x_1)^6 \qquad \text{objective 1} \tag{5.1}$$

$$f_2(\mathbf{x}) = g(\mathbf{x})(1 - \left(\frac{f_1(\mathbf{x})}{g(\mathbf{x})}\right)^2) \qquad \text{objective 2} \tag{5.2}$$

$$g(\mathbf{x}) = 1 + \frac{9}{(n-1)} \sum_{i=2}^{n} x_i \qquad \text{help function} \tag{5.3}$$

$$\mathbf{x} \in [0,1]^d \qquad \text{box constraints} \tag{5.4}$$

The number of variables $n$ is similar to the number of dimensions $d$ and should at least be equal to 2 or greater than 2. The number of variables $n$ in this run was set to 10. The optimal trade off between the two objectives $f_1$ and $f_2$ is known and is plotted in the figures as the green line. To obtain this optimal solution, $x_2, \ldots, x_n$ should go to zero for all solutions which makes $g(\mathbf{x})$ equal to 1. The variable $x_1$ obtains a different value for each solution between 0 and 1.

The figures in 5.8 on page 47 show the evolution of a population on the ZDT6 test problem. When the algorithm is converging to the optimal Pareto front, the non-uniform spread among the solutions can be seen in the first 40 generations, the points are mainly clustered around $f_1 = 0.4, 0.7 \ldots 0.8$ and $0.9 \ldots 1$ and is due to the sine in the first objective function. After 40 generations the individuals are converged to the Pareto optimal front and the algorithm tries to bring in more diversity among the individuals. This can be seen in the last 20 generations when the solutions get more evenly spaced over the optimal front.

### 5.1.2 Constrained problem - CONSTR

A second test problem tests the ability to deal with constraints. This problem is characterised by two constraints which restricts the *solution space*. The formulation of this problem is given in formulae 5.5 to 5.8.

$$f_1(\mathbf{x}) = x_1 \qquad \text{objective 1} \tag{5.5}$$

$$f_2(\mathbf{x}) = \frac{1 + x_2}{x_1} \qquad \text{objective 2} \tag{5.6}$$

$$g_1(\mathbf{x}) = x_2 + 9x_1 \leq 6 \qquad \text{constraint 1} \tag{5.7}$$

$$g_2(\mathbf{x}) = -x_2 + 9x_1 \leq 1 \qquad \text{constraint 2} \tag{5.8}$$

$$\begin{aligned} x_1 &\in [0.1, 1] \\ &\qquad\qquad \text{box constraints} \\ x_2 &\in [0, 5] \end{aligned} \tag{5.9}$$

Figure 5.2 shows on the left the original solution as obtained by Deb et al. [14] and on the right the solution obtained by the Matlab implementation. The green lines represent the Pareto optimal front and the effects of the two constraint functions. The Matlab implementation is able to reproduce the same result.

To see the ability of the algorithm to satisfy the constraints, figure 5.3 shows the initial and the first generation of the algorithm. In the first generation there are a number of

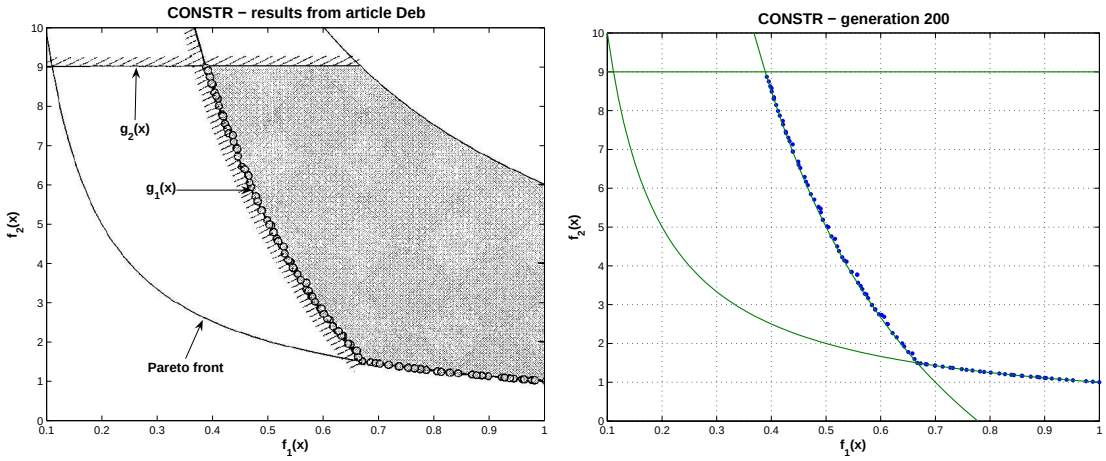



**Figure 5.2:** original solution (left) and solution of the Matlab implementation (right) of the CONSTR problem solved by the NSGA-II algorithm

solutions that violate the constraints and are therefore colored red. These can already be replaced after one generation by solutions that do satisfy the constraints. In the course of the optimization, solutions that still violate the constraints are generated but they don't survive the selection process to actually end up as an individual in the next generation.

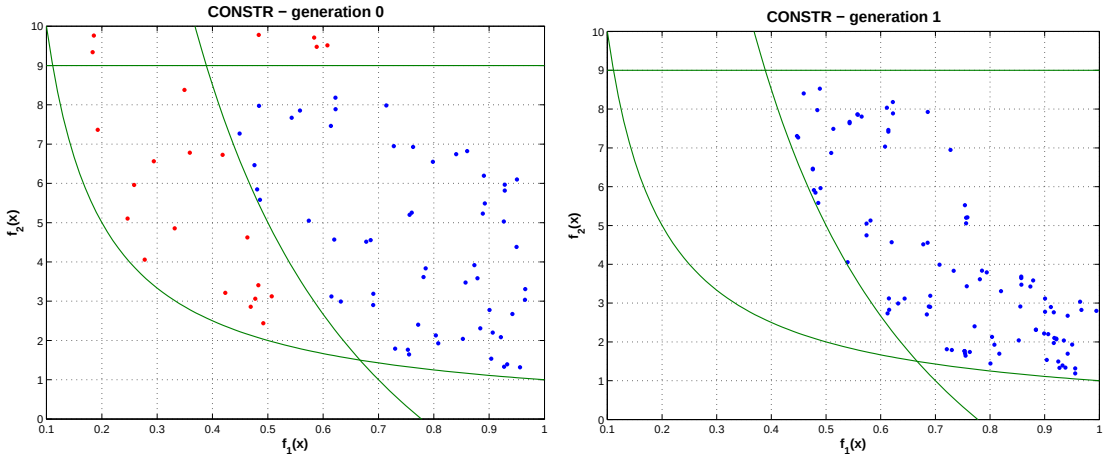



**Figure 5.3:** initial solution (left) and first solution (right) of the algorithm while discarding solutions not satisfying the two constraints

### 5.1.3 Convergence

In the case of test functions, the front of Pareto optimal solutions is known and it is easy to set up performance metrics to evaluate the algorithm as done by Deb et al. [14]. He

describes two metrics to measure (i) convergence to the true Pareto optimal front and (ii) diversity among its front. The first measure is based on the true front which is not available in a real life problem. To get a good indication if the algorithm has converged, a new metric is needed and we try to find this by looking at the age of the individuals in a solution.

If we look at figure 5.8 we can distinguish the 2 phases as described by Deb et al. [14]. The algorithm has converged at generation 50 to 60 and after that it tries to bring in diversity among its front which it should have obtained at a generation of 80. Convergence is characterized by a lot of new individuals that replace the former generations. This has its effect on the average age which should therefore be rather low. The number of new individuals on the other hand should be high. When the algorithm is trying to put in diversity among its front after 60 generations, it is harder to find new individuals that qualify. The average age should rise and the number of new individuals should drop.

**Unconstrained problem** Figure 5.4 shows on the left the number of new individuals in each generation and the average age from a run on the ZDT6 test problem. If we compare this with the results in 5.8 we see a resemblance with our observations just made. The number of new individuals starts out rather high but is continuously dropping until generation 60 where it stays around a value of 25. The average age is at the same time gradually growing until at generation 70 its average value stays around 3.8. There is a clear indication that the algorithm has converged and that at also the same time it has obtained diversity among its front, even though this is hard to distinguish in figure 5.4.

This figure 5.4 looks a little erratic but it is helpful in determining whether the algorithm has converged or not. To support this assumption, the simulation is run 10 times and the average for both measures is plotted in this figure on the right. We see that the erratic behaviour is now almost gone and that we can better see that after 70 generations the algorithm has converged.

**Constrained problem** We can now do the same for the constrained test function where we obtain the same behaviour. Figure 5.5 shows us that the algorithm is converged after 30 generations but that it takes 60 to 70 generations to bring in enough diversity. The average result of 10 simulations is shown in this figure on the right where it is clearer to see that after 30 generations there aren't hardly any new individuals and that it takes 80 generations for the average age to become stable.

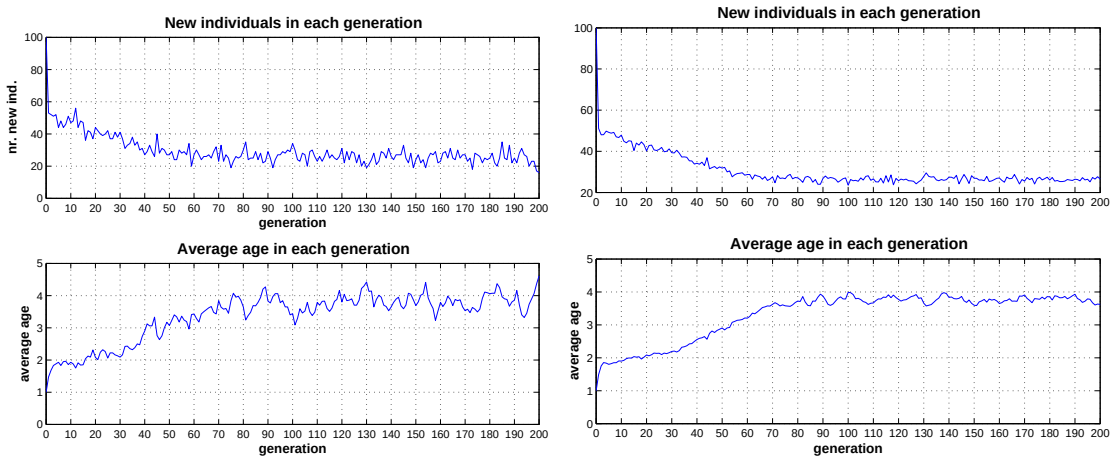



**Figure 5.4:** number of new individuals and average age in each generation for the ZDT6 test problem (left) and averaged results of 10 simulations (right)

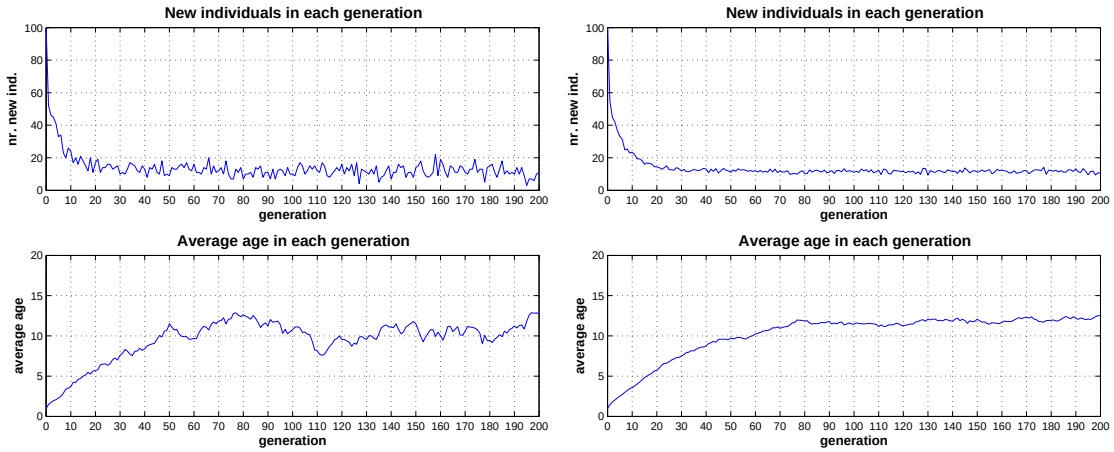



**Figure 5.5:** number of new individuals and average age in each generation for the CONSTR test problem (left) and averaged results of 10 simulations (right)

## 5.2 B-Series & Burrill

Besides test problems to test the optimisation algorithm, meaningful objective functions that can resemble a propeller design problem should also be developed, implemented and tested. To test the design case of its validity and applicability, a simple and quick evaluation of propeller performance (B-Series) and cavitation performance (Burrill diagram) is first used. Once this is working properly, it is replaced by the more advanced propeller analysis program ANPRO.

### 5.2.1 B-Series

Part of the objective function that asses propeller performance is implemented by the Wageningen B-Series data. The B-Series are available as polynomials in the form as given by equations 3.1 through 3.4 which makes them perfectly suited to be implemented in a computer code. The code is a Matlab function that takes as variables the number of blades $Z$, the expanded blade area $A_E/A_0$, the pitch ratio $P/D$ and the advance ratio $J$. It returns the thrust coefficient $K_t$, the torque coefficient $K_q$ and the propeller efficiency $\eta$.

### 5.2.2 Burrill cavitation diagram

The cavitation diagram of Burrill is only available in chart form. This chart is digitised and interpolated by an internal Matlab routine. The routine is based on an algorithm developed by Sandwell [29] and is capable to approximate the chart, see figure 5.6. Combinations of $\sigma_{0.7R}$ and $\tau_c$ that lay outside the chart and would result in a negative percentage of cavitation are truncated to zero. Values larger than 50% are truncated to 50%. Values of $\sigma_{0.7R}$ that exceed the value 1.5 are also truncated to zero.

### 5.2.3 Integration into one objective function

The B-Series polynomials and the Burrill chart are integrated into one function. The results from the B-Series is used to calculate $\sigma_{0.7R}$ and $\tau_c$ which are required to evaluate the Burrill chart. This is then returned to the optimiser. It is now possible to run a systematic series of varying the expanded blade area ration to see its influence on performance and cavitation as described in the introduction. The results are shown in table 5.1 and as expected, increasing the blade area ratio reduces cavitation and efficiency.

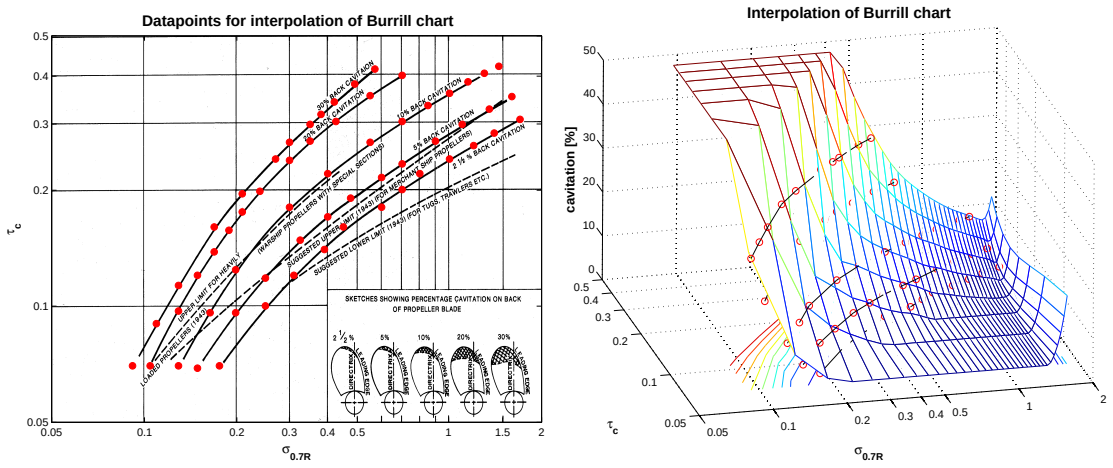



**Figure 5.6:** chosen datapoints (left) used for interpolation (right) of the Burrill cavitation diagram

**Table 5.1:** Systematic variation of EAR with B-Series

| EAR [−] | $\eta$ [−] | cavitation [%] |
|---|---|---|
| 0.8 | 0.5359 | 6.2681 |
| 0.9 | 0.5323 | 4.3213 |
| 1.0 | 0.5269 | 2.6777 |
| 1.1 | 0.5198 | 1.2987 |

## 5.3 ANPRO

The implementation of the analysis program ANPRO is more elaborated. The program requires more input data and should be run as an external program in the Matlab environment. More input data means in most cases also more output data which should be read, interpreted and send back to the objective function.

### 5.3.1 General outline

The routine consists of the following parts:

**Bookkeeping** The evaluation of the objective function starts with creating the proper directories to run the ANPRO calculations in and the creation of the files necessary starting them. These include a batch-file, a file with program options and some temporary files used by ANPRO itself. Furthermore, the file with the wake-field data is copied too.

**Creating geometry** Depending on the optimization variables, the geometry is created using a program named GEOCO. This program takes variables that describe the geometry like pitch, chord, thickness, camber in terms of a distribution over the radii. The sectional profiles are fixed to NACA 66 modified thickness distribution and a NACA 65 mean line. An output file is created that can be read by ANPRO and is copied to the proper directory.

**Running ANPRO** Once all the necessary files are generated, ANPRO is run with the proper options selected. There are 3 options to run the program: (i) performance calculation (ii) idem but without calculating the influence coefficients again and (iii) cavitation analysis. It is necessary to do at least one run with the first option to determine the influence coefficients.

**Reading output** After each run of ANPRO the output file is read and depending on which options ANPRO is set, the proper data is returned (performance data and/or cavitation data).

### 5.3.2 Interpretation of cavitation analysis

The cavitation analysis within ANPRO gives data on bubble cavitation and sheet cavitation on both suction and pressure side of the blade. This data is calculated for 60 blade angles in one revolution and is presented in one big table. The percentages of bubble cavitation are integrated over the radius of the blade. At the blade position where the largest amount of bubbles occur, this value is returned to the main function. Sheet cavitation can be evaluated in the same way as bubble cavitation but a more practical value is the maximum sheet *length* that occurs in one revolution, see figure 5.7 and this value is therefore returned instead.

### 5.3.3 Systematic variation of EAR

To test the routine a systematic variation of the blade area ratio is also performed with ANPRO. The geometry of a B-Series propeller (except for different sectional profiles) is used and the blade area ratio is systematically changed. The results on efficiency and cavitation is given in table 5.2 and the same trend is visible as with the variation with the B-Series.

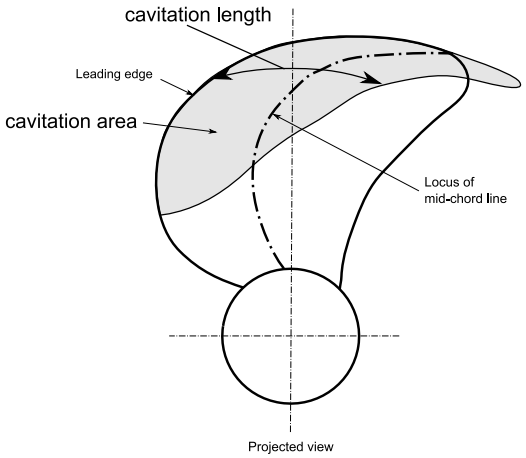



**Figure 5.7:** two ways to interpret sheet cavitation, looking to the maximum length of the sheet relative to the chord length or to the sheet area relative to the blade area

**Table 5.2:** Systematic variation of EAR with ANPRO

| | | sheet cavitation | | bubble cavitation | |
|---|---|---|---|---|---|
| EAR | $\eta$ | SS | PS | SS | PS |
| [−] | [−] | [%] | [%] | [%] | [%] |
| 0.8 | 0.6152 | 107 | 11 | 40 | 0 |
| 0.9 | 0.6089 | 98 | 7 | 20 | 0 |
| 1.0 | 0.6032 | 90 | 5 | 5 | 0 |
| 1.1 | 0.5981 | 82 | 4 | 0 | 0 |

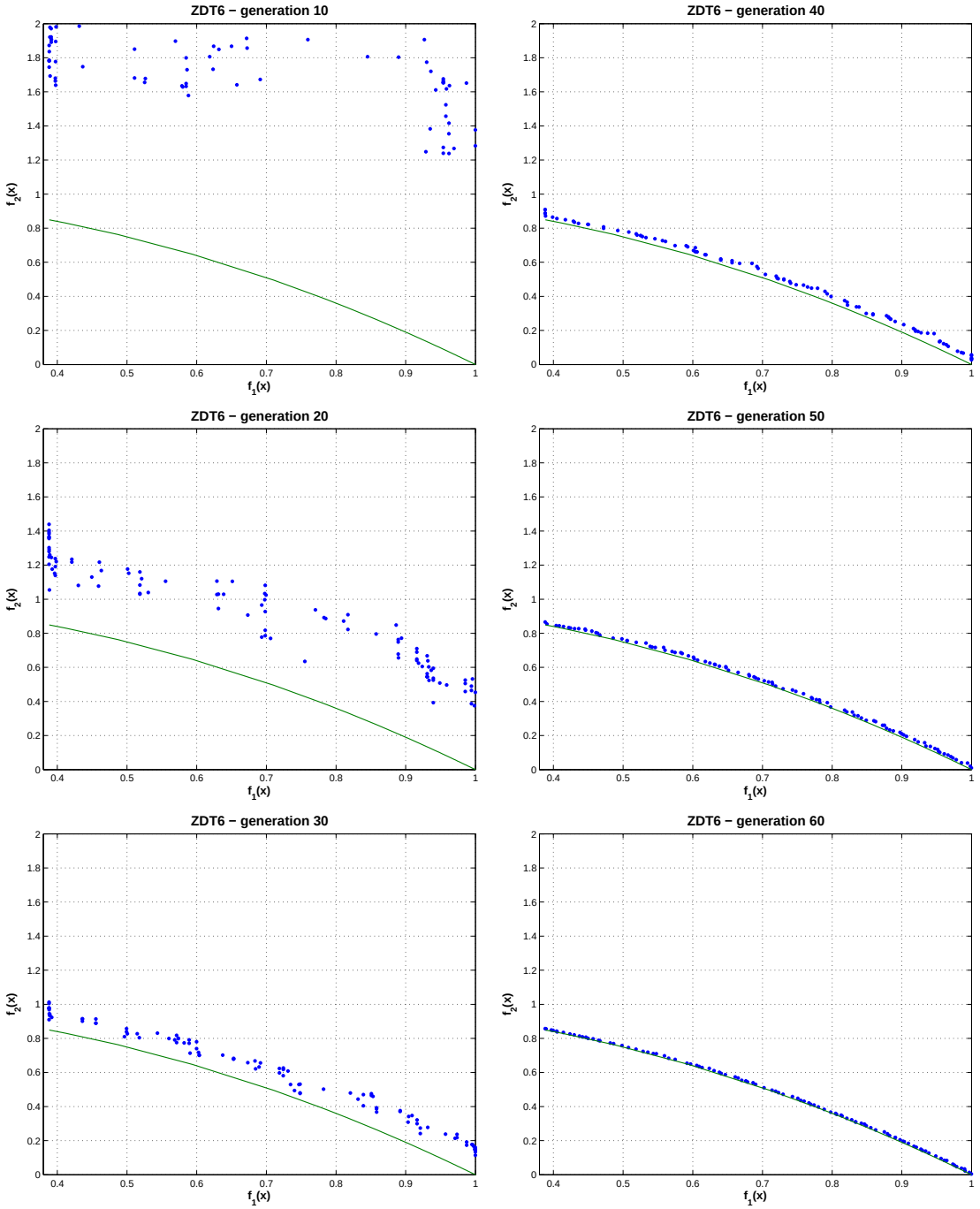



**Figure 5.8:** convergence of the optimization algorithm (top-down, left-right) on the ZDT6 test problem

# Chapter 6

# Design cases


**Figure 6.1:** the sample container vessel used in the design cases moored at the ECT container terminal in the harbour of Rotterdam

## 6.1 Container vessel design case

To apply the design procedure that was developed in the previous chapters, a design test case is setup with the data of a representative large container vessel. The data is taken from the report by van Wijngaarden et al. [33] and also provides data on full scale sea trials.

### 6.1.1 Main particulars

The vessel is a 6800 TUE container vessel (see figure 6.1) build by Hyundai Heavy Industries in South Korea. The main particulars are given in table 6.1. The ship is at the moment operated by Maersk Line. The stern of the ship is characterized by its typical barge-type shape and Hogner-type gondola. The flow field going into the propeller plane is characterised by a large peak of low velocities in the upper part of the propeller. This makes it difficult to design a propeller that exhibits low cavitation.

**Table 6.1:** Main particulars of the sample container vessel

| | | |
|---|---|---|
| **Length over all** | 299.99 | [m] |
| **Beam** | 42.84 | [m] |
| **Draft** | 13.50 | [m] |
| **Gross tonnage** | 80654 | [t] |
| **Nett tonnage** | 46660 | [t] |
| **Deadweight** | 87370 | [t] |
| **Speed** | 25.50 | [kts] |
| **Capacity** | 6802 | [TEU] |

### 6.1.2 Design criteria

The design of the propeller is done under external criteria which are taken as given in the design process. These conditions can be split up in two parts, flow-related and engine related. For the flow-related conditions we distinguish the inflow in the propeller (the wake field) and the resistance speed curve. The only geometrical aspects of the propeller that are taken as fixed are the diameter of 8.75 m, the number of blades 6 and the sectional profiles.

**Engine characteristics** The engine is designed to operate at a certain rotational speed to deliver its maximum power. The operational envelope of an engine can be visualised in a diagram where rotational speed and power output are plotted against each other. The design of the propeller should stay within this diagram but here we'll only consider the design point. In its design point the engine should operate at a rotational speed of 100 RPM and is then able to deliver a power of 55,272 kW.

**Flow field characteristics** The wake field in which the propeller operates is important to predict correctly because the cavitation performance is very sensitive for this. The wake field can be calculated by RANS solvers like *Parnassos* or measured on model scale and scaled up to full scale. Both methods have their advantages and disadvantages and here we use the wake field based on model tests. The wake field is the *effective wake field* which means the action of the propeller on the flow around the ship is accounted for, see figure 6.2.

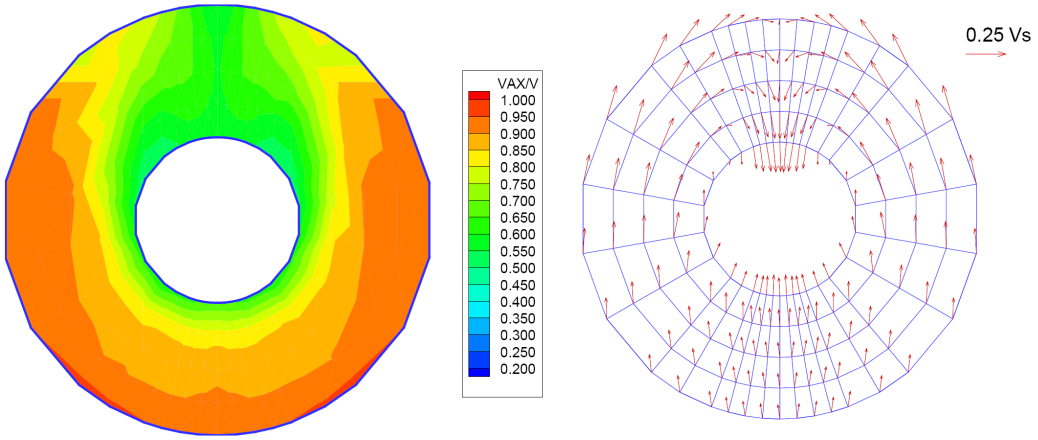



**Figure 6.2:** axial velocities (left) and transverse velocities (right) for the effective wake field which is used in both design cases

**Resistance curve** The resistance curve is taken from model test data done at MARIN. The data is scaled to full scale and a third degree curve is fitted through this data. Figure 6.3 shows the resistance curve in green and the model test data in blue.

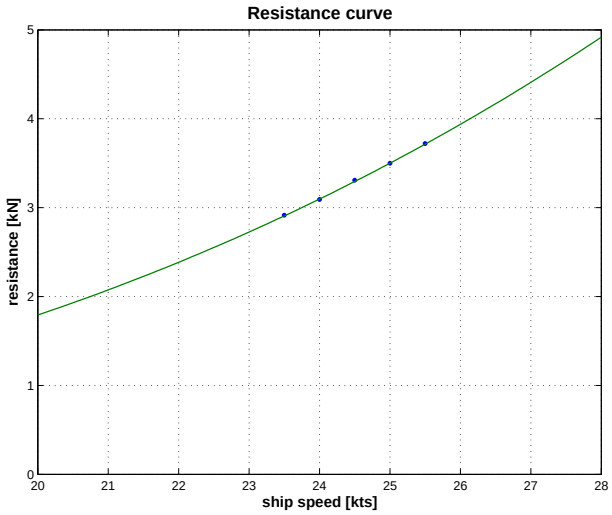



**Figure 6.3:** interpolated resistance curve based on model test data

## 6.2 Case Description

From the design point of view, there are two possible ways to approach the design, from a preliminary point of view and from a detailed point of view. In the *preliminary design* of a ship, the engine is not yet chosen and the goal is to minimise the required power of the engine to obtain a certain design speed. In the *detailed design* the engine is chosen and now the propeller should be designed to obtain at least the required design speed but with the minimum amount of cavitation.

**case 1 - preliminary design** Here the goal is to find the main particulars of the propeller that have the largest impact on the performance of a propeller. These are the diameter, the number of blades, the blade area ratio and the average pitch. The B-Series are specifically designed to assist the designer in this phase. Their use as a set of polynomials is simple and also reliable because the data is based on model tests.

**case 2 - detailed design** In the detailed design case the engine and its characteristics are given and the goal is to design a propeller that is able to obtain highest possible speed (but at least the design speed) while on the other hand exhibiting no dangerous forms of cavitation. The outline given in 2.1 is a good example of the steps that are undertaken to come to a final design. The virtual pitch is used as an aid to maintain the absorbed power by the propeller constant for different geometries and conditions.

**implementation** The simulations that are done for both cases differ from the description given above. The problem description is the same but the cases differ in the variables that are optimised. The preliminary design with an optimiser is now not limited to the main parameters but can incorporate all variables of interest. All the parameters described in chapter 4 are now subjected to change by the optimisation routine.

# Chapter 7

# Results

This chapter presents the results of the computations that are done with the newly developed optimisation procedure. The results are split in a preliminary design case and a detailed design case as described in chapter 6. For the detailed design case, the procedure with the Wageningen B-Series in combination with the Burrill cavitation chart is first applied to see if the method is working. Then ANPRO is used instead and applied to both cases. For the runs with ANPRO a sensitivity analysis is done with the pressure side sheet cavitation. This constraint is changed in steps of 5% from 0% to 20%. Furthermore data on convergence and running time are given for the simulations done with ANPRO.

## 7.1 Case 1 - Preliminary design

This section shows the results for the preliminary design case with both the B-Series with Burrill cavitation criteria and ANPRO implemented to evaluate the objectives for efficiency and cavitation performance. These two conflicting objectives are then plot against each other.

### 7.1.1 B-Series & Burrill

This implementation of the objective function is used to test the algorithm and to show if it is possible to obtain a trade-off curve in the first place. The parameters that can change are limited to the average pitch and the blade area ratio. If the data is taken from the design case then it is possible to choose a B-Series propeller with maximum efficiency that according to the Burrill diagram exhibits no cavitation. This results in no trade-off curve at all.

To avoid this, the propeller loading is gradually increased. This is done by gradually increasing the thrust that the propeller should generate. The results are shown in figure 7.1 and increasing the loading results indeed in a trade-off curve. For an increased loading of $C_t = 0.8$, the trade-off in efficiency is in the order of 2%, for cavitation this is 7%. When

the loading is increased further, the trade-off in efficiency stays in the order of 2% but the change in cavitation is increased, the curves are much steeper.

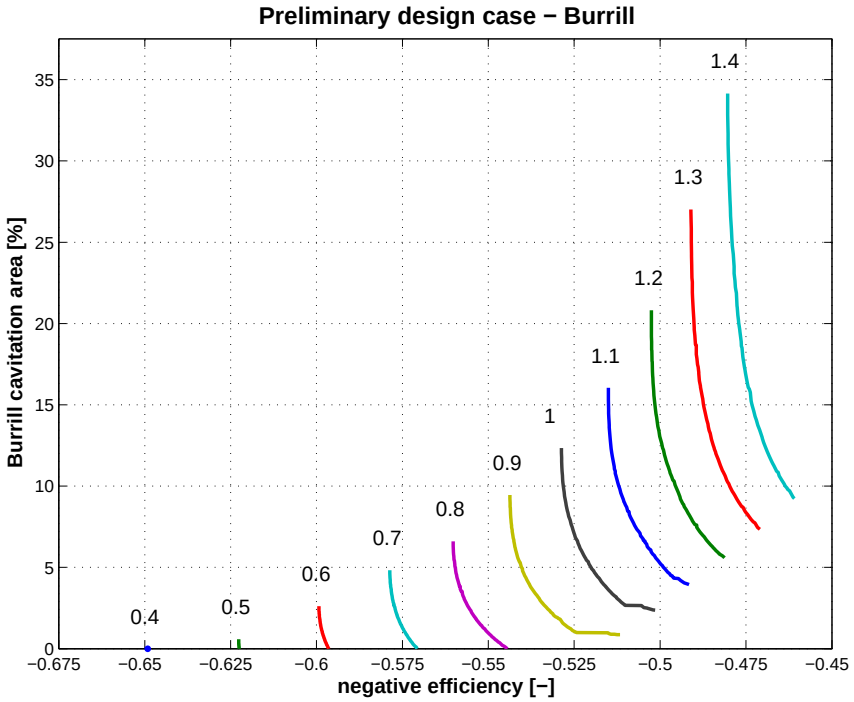



**Figure 7.1:** results for B-Series and Burrill cavitation diagram, the different plots are for increased thrust loading of the propeller

### 7.1.2 ANPRO - Cavitation length as criteria

This section shows the approximated front of Pareto optimal solutions obtained with ANPRO, visualisation of a selection of the optimal designs, a sensitivity analysis of the pressure side sheet cavitation constraint, a convergence plot and the calculation time.

**trade-off curve** The trade-off curve between efficiency and suction side sheet cavitation is shown in figure 7.2. The front consists of designs who vary in efficiency between 58.7% and 62%. The amount of sheet cavitation length varies between 64% and 95%. It can be observed that there are not so many individuals plotted, at least not 80, the number of individuals in a whole generation. This is because a lot of individuals who exhibit only small variations in the input don't result in a different outcome of the ANPRO analysis. The results from ANPRO for the sheet cavitation length is given as a percentage with zero decimals accuracy. This explains why the individuals increase in steps of minimal 1% of each other.

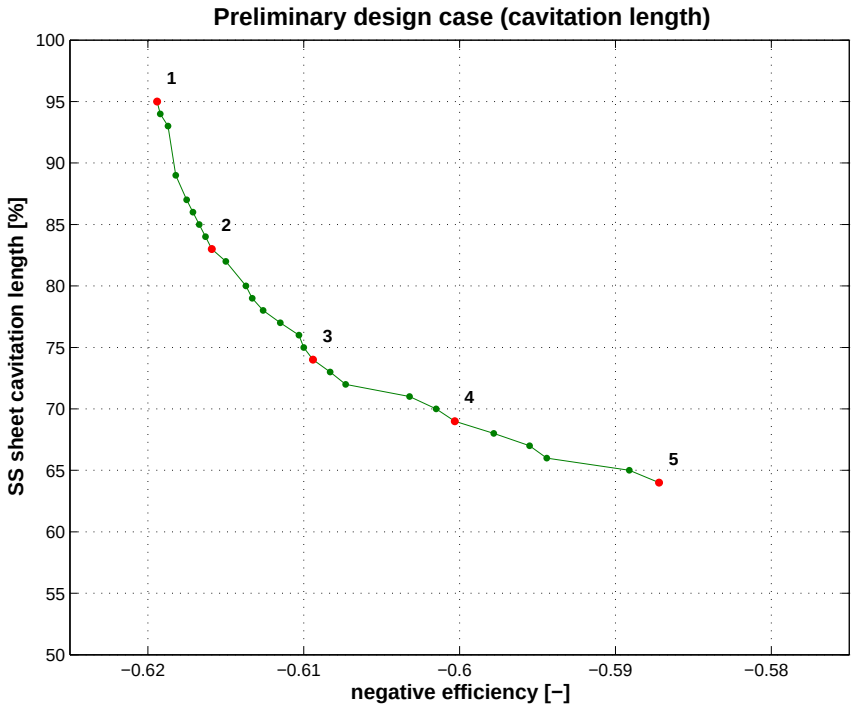



**Figure 7.2:** results of preliminary design case with 5% allowed PS sheet cavitation

**cavitation visualisation** To see how the different designs along the Pareto front differ from each other, 5 designs are picked out of figure 7.2 and plotted in figure 7.3. The plots show the upward position of the blade, this is in most cases the position that exhibits the most cavitation. The local pressure is lowest in this position and more important, the blade has to pass through the wake peak here.

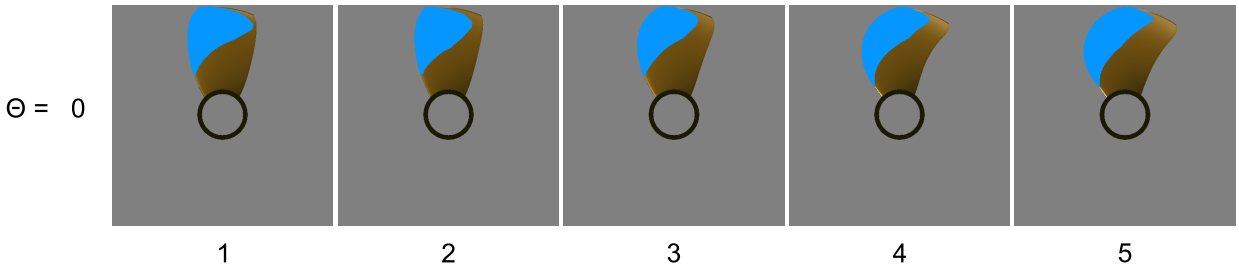



**Figure 7.3:** visualisation of 5 designs along the Pareto front

On the left we see a propeller with the highest possible efficiency but also with the largest cavitation length. The cavitation length relative to the local chord is taken as criteria. Moving to the right the cavitation reduces together with efficiency. Besides the cavitation, the shape of the blade also changes. The blade area ratio seems to increase and also skew

is introduced. The appendix on page 79 gives an overview of the above designs for different blade angles, the growth and decay of the cavity can be observed when the blade passes through the weak peak.

**variation allowed PS sheet cavitation** Next, the constraint on the pressure side sheet cavitation is varied to see its influence on the trade-off, the results are given in figure 7.4. Here we see 5 different lines and the different colors represent the different constraints. The green line is the same line as in figure 7.2. It can be seen here that all the other results also increase cavitation with steps of one percent.

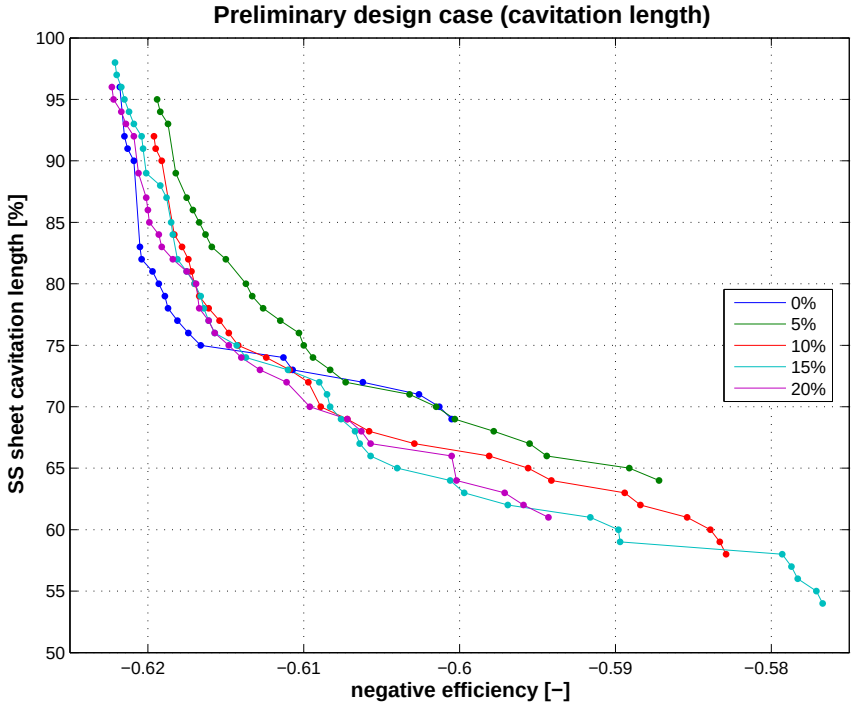



**Figure 7.4:** results of preliminary design case while systematically varying the constraint on PS sheet cavitation

Relaxing the constraint on the pressure side sheet cavitation would suggest that there is more room for other variables to come to a valid design. This would suggest an improvement of the front of optimal design solutions. In other words, the blue line with the most strict constraint of 0% should be on the top-right and the fronts should gradually improve into the direction of the down-left corner. This is happening but not in a very consistent way. The 0%-line is deviating the most from this behaviour by crossing all the other lines.

The same reasoning can be applied to the length of the front. Relaxation of the constraint would imply that more extreme solutions would enter the front. This can be observed in almost all the lines except for the 20% line. The lines extend more to the right of the figure

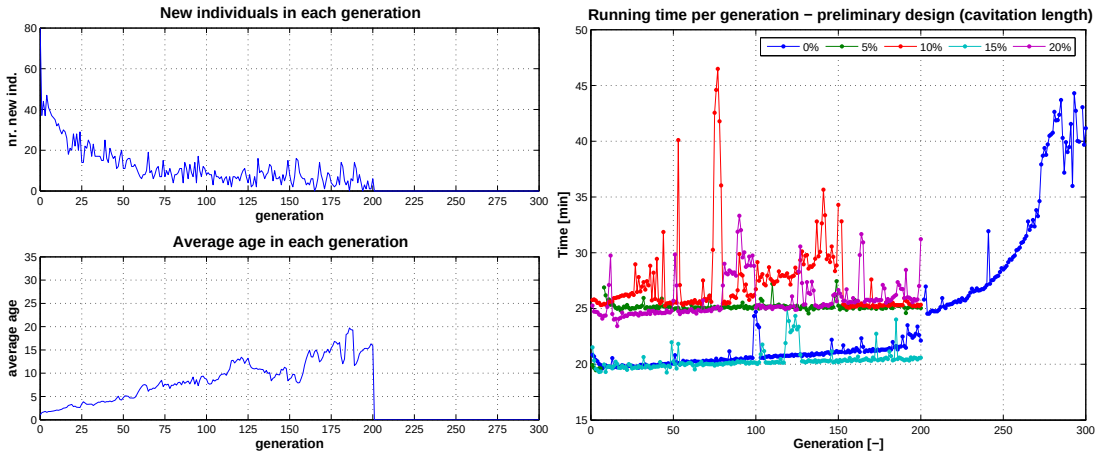
where solutions are found that exhibit lower efficiency with a shorter cavitating sheet on the blade.

**convergence** To see if the algorithm has converged we look at the number of new individuals and the average age as plotted in figure 7.5(a). Here the axis is set for 300 generations but the first simulations were run for only 200 generations. As discussed in chapter 5 we try to determine convergence of the simulation by looking at the behaviour of the number of new individuals that enter a generation and the average age of a generation.

The number of new individuals is gradually decreasing as we have seen before on the constraint test problem. This suggest the algorithm is converging but not *if* it has converged. The plot is too erratic between 100 and 200 generations. If the algorithm has converged, the average age should approach a limit value. This is difficult to see from figure but it looks like it did not. At this point it was decided to let the algorithm run for 300 generations.

In the appendix on page 76 the average is given over the 5 simulations. It can be seen that the average age is still increasing and didn't get to a limiting value before 300 generations. This suggests that even 300 generations is not enough to converge.

**running time** A more practical result from the simulations is the average calculation time required to make one run. This is plotted in 7.5(b) where the running time for one complete generation varies between 20 and 45 minutes. Not all simulations were run on the same computer. It is clearly visible that there is an average of 20 minutes and one of 25 minutes for evaluating a complete generation. The simulation with 0% pressure side sheet cavitation deviated from the trend as describe in previous paragraphs and it was therefore decided to let it run for an extra 100 generations. The simulation was also migrated to another PC where it clearly took longer to finish than on the original PC.




(a) convergence results showing number of new individuals and the average age in each generation

(b) calculation time in minutes to analyse one generation of 80 individuals

**Figure 7.5:** Convergence (left) and calculation time per generation (right)

### 7.1.3 ANPRO - Cavitation area as criteria

This section is the same as the previous one except that it is now based on a different interpretation of the sheet cavitation data. Instead of cavitation length, cavitation area is now a measure for the second objective function.

**trade-off curve** The first thing that can be observed from the trade-off curve as plotted in figure 7.6 is that the y-axis is shifted down. The amount of sheet cavitation is now in the range of 13% to 41%. The plot now also contains more points as the resolution is increased with which the percentage of cavitation is calculated. This also results in a smoother curve. The range off efficiency that is covered, is larger in this case. Both implementations don't go above 62% but we see that it can get lower than the 58.7%, namely to 57.9%.

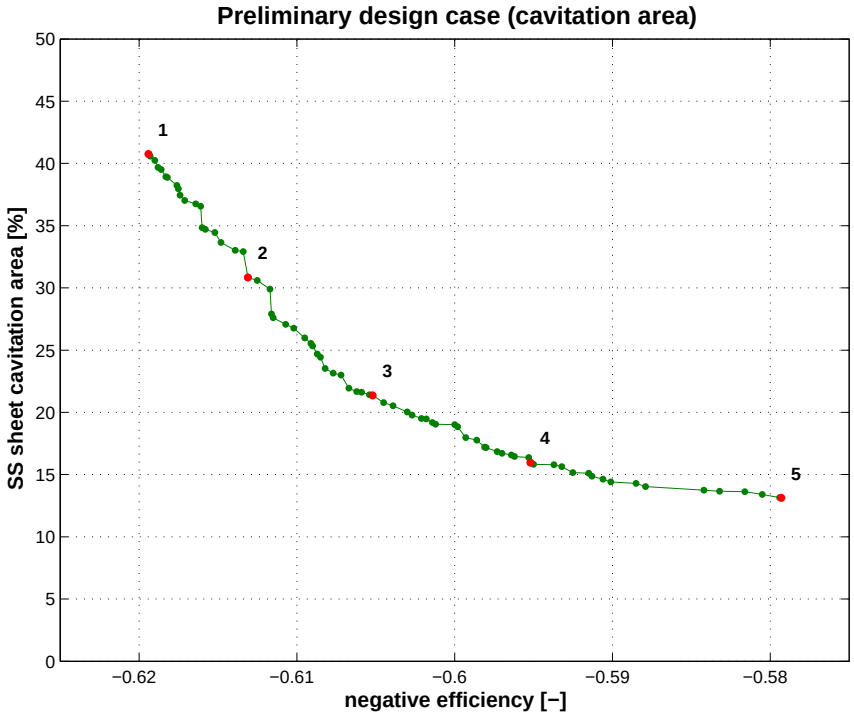



**Figure 7.6:** results of preliminary design case but now with sheet cavitation *area* as a criteria for both SS and PS

**cavitation visualisation** Figure 7.7 shows 5 different designs along the Pareto front as depicted in figure 7.6. On the left is again the propeller with the highest efficiency and on the right the lowest efficiency. Cavitation changes in the same manner from left to right. The changes in blade shape follow the same trend as the case with cavitation length. High efficiency propellers mainly have straight blades with a low blade area ratio. Skew appears for lower efficiency designs.

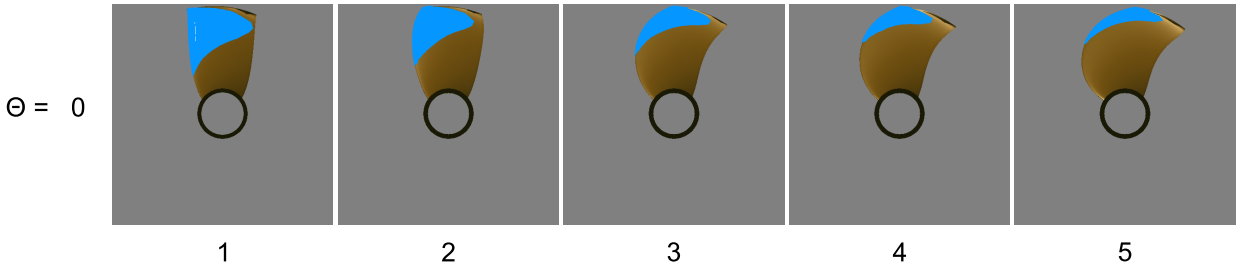



**Figure 7.7:** visualisation of a range of designs along the Pareto front

An overview of different blade angles is given in the appendix at page 81. Here it is possible to see how the sheet cavitation on the suction side grows and disappears when the blade passes the wake peak. The maximum area of the sheet occurs when the blade passes the top position. There is a difference in growth and disappearance of the sheet. The higher efficiency designs exhibit the largest sheet but also for a shorter period of time, mainly in the top position. The lower efficiency designs have less cavitation but cavitate on average more during one rotation.

**variation allowed PS sheet cavitation** A sensitivity analysis for the pressure side sheet cavitation constraint is also done. The result is shown in figure 7.8. The gradual increase of the front for a relaxation of the constraint is here also observed except for the 15% curve which is breaking with the trend here. The length of the front is also increased for a relaxation of the constraint. The constraint of 0% sheet cavitation at the pressure side limit the blue line to 59.7% of efficiency. The 10%, 15% and 20% curves have their extreme more and more to the right. Now the 5% curve is breaking with the trend.

**convergence** Here we look again at the number of new individuals and the average age to get a clue if the algorithm has converged or not. Now all the simulations are run for 300 generations and figure 7.9(a) shows the result for the run with 5% allowed PS sheet cavitation. The number of new individuals gradually decreases and appears to stay around a value of 5 after 150 generations.

The average age shows the same trend. After 150 generations it appears to have achieved a limiting value of 15 but keeps increasing until all the 300 generations are finished. All of the other simulations show the same behaviour. If we take a look at the averaged values of all the 5 simulations in figure B.1(f), we see that the average age is indeed increasing until 300 generations have passed.

**running time** Figure 7.9(b) shows the results of the required calculation time. Now the simulations are run on only two different workstations which are almost equal to each other in terms of speed. In most cases, the average of 25 minutes is obtained to evaluate one

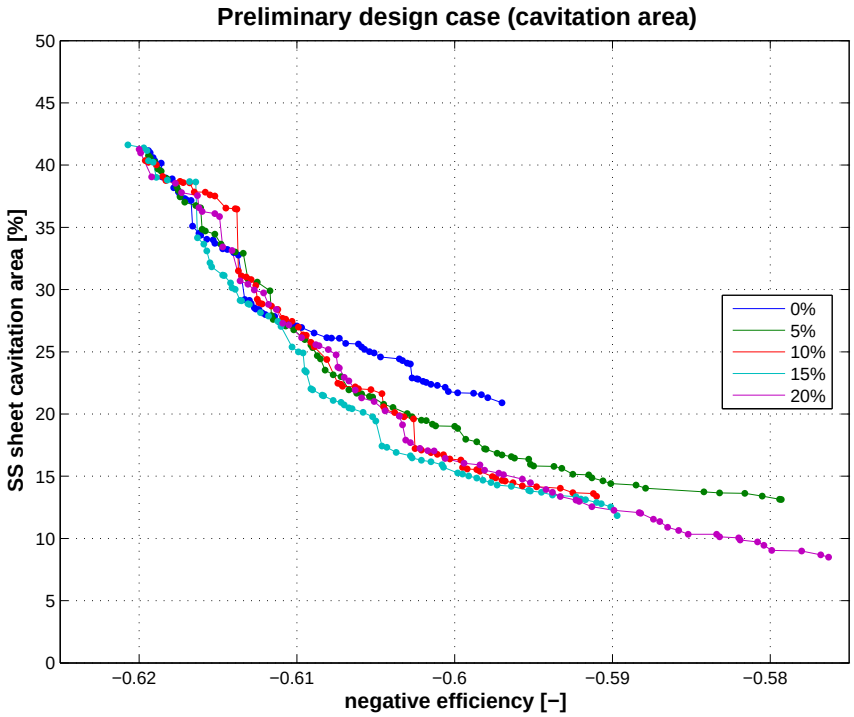
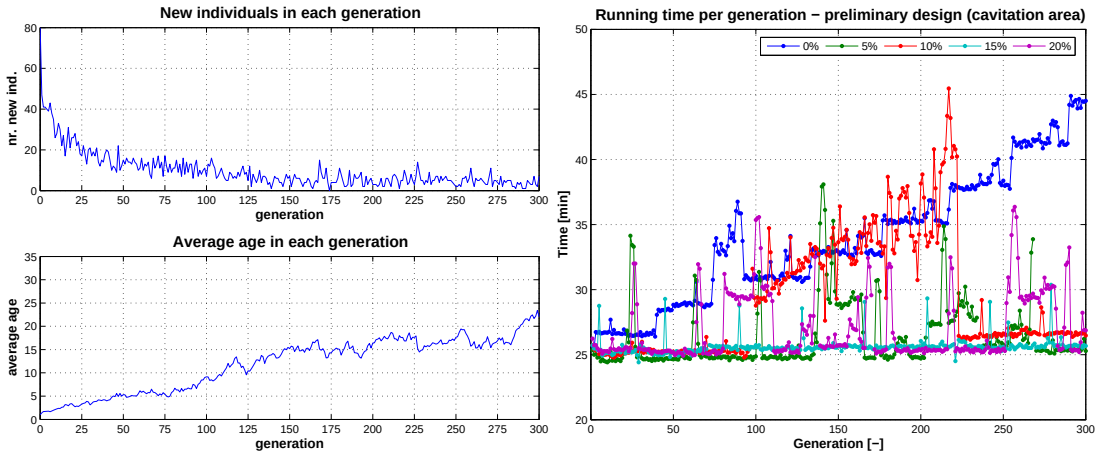



**Figure 7.8:** results of preliminary design case but now with sheet cavitation *area* as a criteria for both SS and PS

generation. Occasionally there are some spikes that are caused by a network update or user intervention at the workstation. The blue and red line however show an increasing evaluation time during a complete simulation. In most cases Windows Explorer is the cause because it requires more and more resources when the amount of folders is increased during the simulation (each ANPRO calculation is put in its own folder). When closed, the evaluation time drops back again to the average value which is what happens with the red line.




(a) convergence results showing number of new individuals in each generation and the average age for the run with 5% allowed PS sheet cavitation

(b) overview of the calculation time in minutes to analyse one generation of 80 individuals for all the simulations

**Figure 7.9:** Convergence (left) and calculation time per generation (right)

## 7.2 Case 2 - Detailed design

This section shows the results for the detailed design case with only the implementation of ANPRO to evaluate the objectives for efficiency and cavitation performance. The two conflicting objectives that are plot against each other are ship speed and cavitation.

### 7.2.1 ANPRO - Cavitation length as criteria

This section shows for the detailed design case the approximated front of Pareto optimal solutions, a sensitivity analysis of the constraint on pressure side sheet cavitation, convergence plot and calculation time.

**trade-off curve** The trade-off curve is plotted in figure 7.10 and shows the maximum obtainable speed versus the amount of sheet cavitation length. It is possible to obtain at least the minimum design speed of 24.5 knots with a sheet length of 35% of the chord length. The curve is gradually increasing and almost like a straight line, this in contradiction to the previous case. This is caused by the change in inflow velocity, its influence on the cavitation behaviour is that it suppresses it.

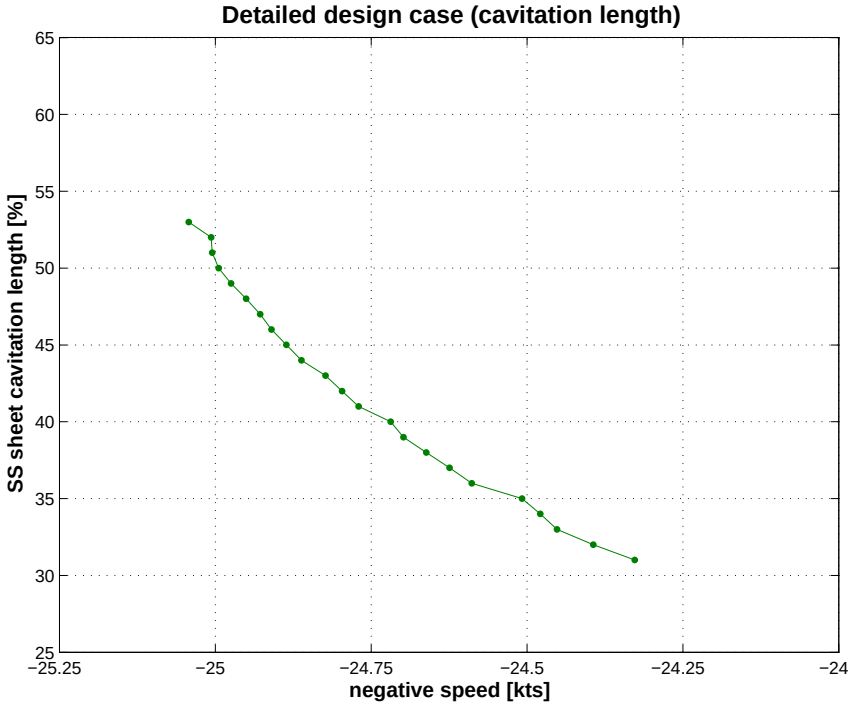



**Figure 7.10:** results of detailed design case with 5% allowed PS sheet cavitation

**variation allowed PS sheet cavitation** The variation of the constraint on the allowed pressure side sheet cavitation is show in figure 7.11. The different variations are close together and it is hard to tell them apart. There is also not a clear ordering for an relaxation of the constraint. The 0% crosses the 5% curve several times and the same goes for the others. It is clear though that for a relaxation of the constraint, higher speeds can be obtained with evidently more cavitation.

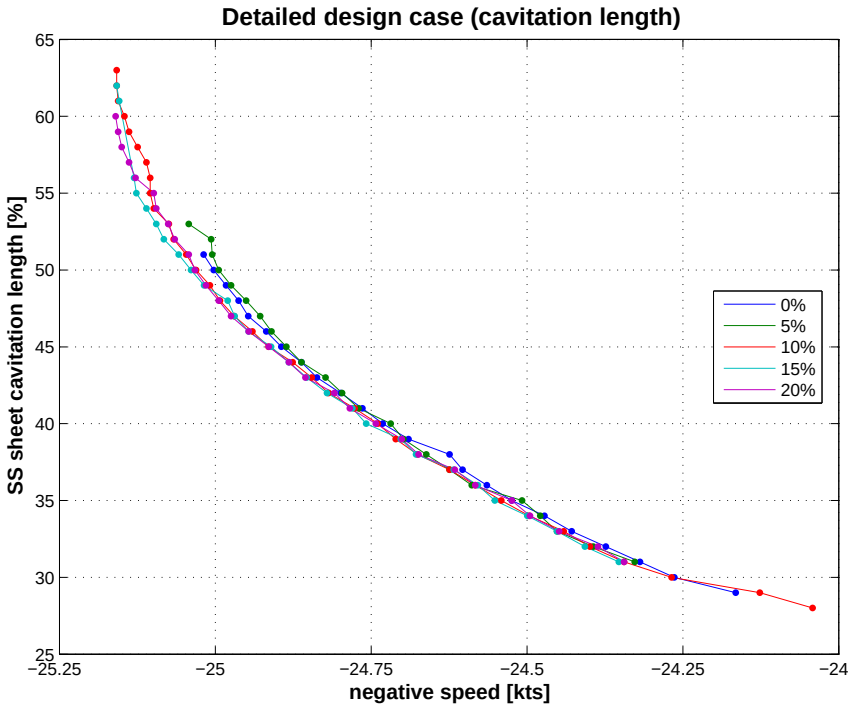



**Figure 7.11:** results of detailed design case and systematically varying the constraint on PS sheet cavitation

The increase in efficiency looks higher on first sight but this is not the case. An increase in speed of 1 knot is equivalent to 4%. Translated to required power (which is a third order curve) this means an increase of 12%. Figure 7.12 shows the real increase in efficiency which is in the order of 2%.

The explanation for this behaviour is related to the use of the virtual pitch as an aid to the designer to keep the maximum absorbed power constant. The virtual pitch should only be used under the condition that the rotational rate and the velocity are kept unchanged. A change in blade area ratio is accounted for but the change in velocity is significant. This results in much higher absorbed power ratings and not in an increase in efficiency.

**convergence** A plot of the average age in each generation is shown in figure 7.13(a). The simulations were run for 200 generations except for the 5% case. This was the first

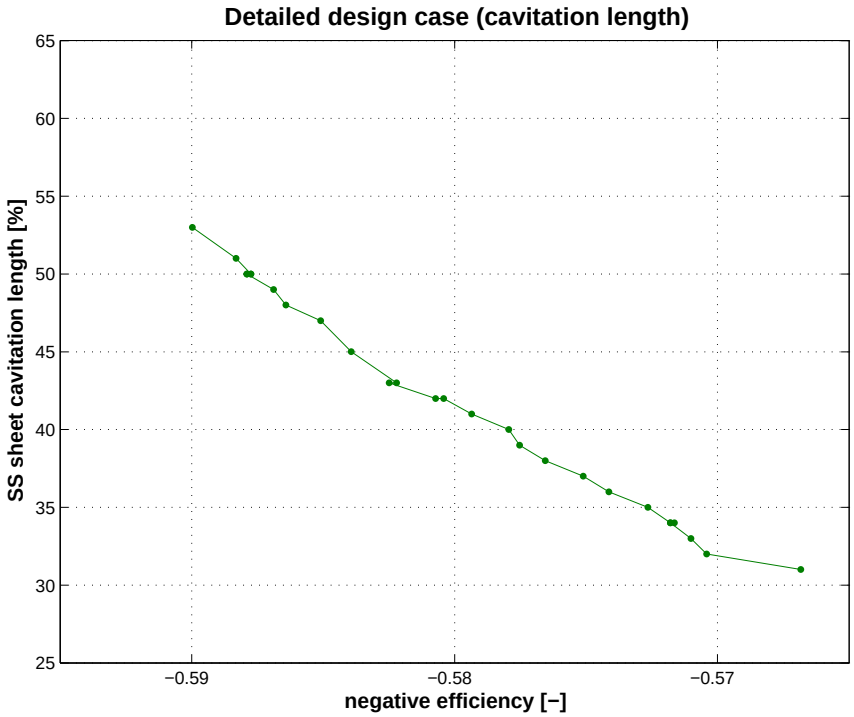
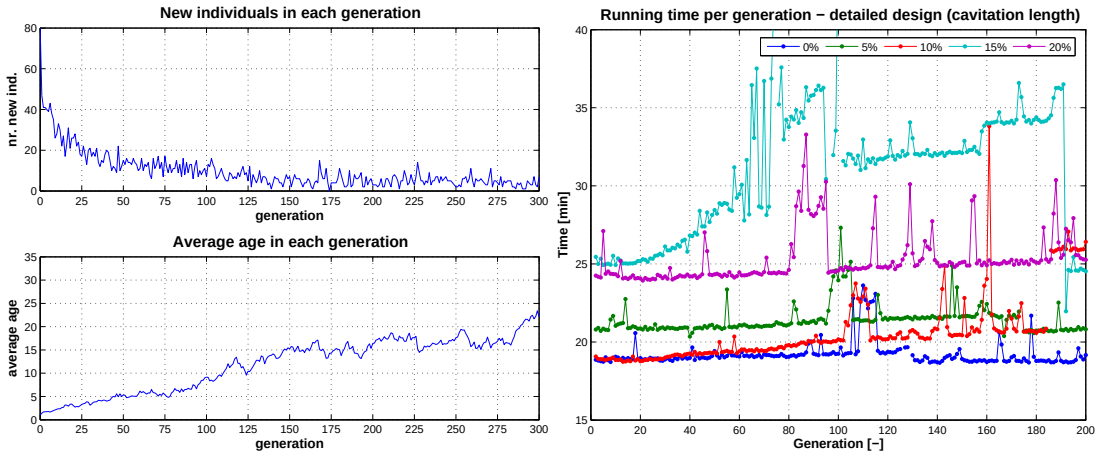



**Figure 7.12:** results of detailed design case for 5% allowed PS sheet cavitation but now with efficiency plotted against the amount of SS sheet cavitation

simulation that run for 300 generations to see if the average age keeps on a constant level. This is indeed happening until after 275 generations it suddenly jumps up. An inspection of the trade-off curve showed that the algorithm was able to find an improvement of the Pareto front. When this happens, several old individuals are replaced by new ones.

**running time** Figure 7.13(b) shows the running time for an evaluation of one generation. Here it is obvious to see that some of the simulations are run on different computers. The 15% simulation shows some strange behaviour. It gradually increases in evaluation time and here again it is Windows Explorer that was not closed while running the simulations. Only during the last 10 generations it drops to its original evaluation time of 25 minutes.




(a) convergence results showing number of new individuals in each generation and the average age

(b) overview of the calculation time in minutes to analyse one generation of 80 individuals for all the simulations

**Figure 7.13:** Convergence (left) and calculation time per generation (right)

# Chapter 8

# Closure

This chapter finalises the research with conclusions, recommendations for a future implementation and an outlook of applications in other areas besides propeller design.

## 8.1 Conclusions

The conclusions are separated in three categories; related to the algorithm, the outcome of the simulations and the outcome of the sensitivity analysis on the pressure side sheet cavitation constraint.

**algorithm** The test problems showed a successful implementation of the optimisation algorithm by reproducing two well known test problems from literature. The average age and the number of new individuals in each generation to indicate if a run is converged proofed to be a helpful approach. This measure showed no complete convergence for the propeller test cases. This is believed to be caused by the more complex nature of the objective function which is mainly caused by the implementation of cavitation as a constraint. Compared with the ZDT6 test problem, these cavitation constraints are not just limiting a certain area of the feasible space but show a solution of small islands scattered over the feasible space. One could say there is not a simple path out of these areas that can guide the algorithm to a feasible part of the solution space. The inability of the algorithm to converge after 200 generations was the reason to extend the number of generations to 300 but even this was not enough.

**test cases** The construction of an approximation of the front of Pareto optimal individuals is successful even though the algorithm is not able to converge completely it gives a good impression of the trade-off. The results of the simulations for the different cases is summarised in table 8.1. The table shows the trade-off between the two conflicting objectives cavitation and efficiency.

**Table 8.1:** Results overview of trade-off with 5% PS allowed sheet cavitation

| | | Objective 1 - Performance | | | Objective 2 - Cavitation | | |
|---|---|---|---|---|---|---|---|
| Case 1: preliminary design | result 1 | efficiency : | 3.2 | [%] | length : | 30 | [%] |
| | result 2 | efficiency : | 4 | [%] | area : | 27 | [%] |
| Case 2: detailed design | result 3 | speed : | 0.75 | [knots] | length : | 25 | [%] |
| | | efficiency : | 2.3 | [%] | length : | 25 | [%] |

The data in the table is consistent, both efficiency and cavitation are of the same order of magnitude for all the simulations. This is surprising because the simulations were not completely converged. The interpretation of the cavitation data as an objective for the algorithm is very important. Cavitation area as a measure instead of cavitation length makes the trade-off curve shift downwards which is caused by lower calculated cavitation percentages. It also results in different propeller designs where the cavitation behaviour is different. The visualisation's show less cavitation during a passage through the wake-peak for the designs based on cavitation area. The cavitation also occurs for a shorter period of time.

The running time is in general in the order of three days. This depends on the amount of generations (300) that the optimiser is allowed to run and the chosen population size (80 individuals). The evaluation of one design with ANPRO takes 20 seconds. Most of the time (15 seconds) is spent determining the influence coefficients of a design, cavitation analysis takes 5 seconds. Calculation time is influenced by external processes that are running on the workstation.

The choice of parameters describing the propeller geometry offers enough freedom to come up with designs that are feasible according to the constraints set in the optimisation and also allow for enough variation. The choice to describe the geometry by means of distributions based on Bézier curves guarantees smoothness in the geometry while reducing the number of variables describing this geometry. To a certain extent it also limits the possible geometries that can be described but not so much that variation among designs disappears.

The application of the virtual pitch as an aid in the detailed design process proofed unsuccessful. The virtual pitch is applicable for the assumption that rotation rate and inflow velocity stay constant which is not the case when the velocity is an objective of the optimisation.

**sensitivity analysis** The inability of the algorithm to converge within 300 generations show its effect in the outcome of the variation of the pressure side sheet cavitation constraint. The expected effect of loosening the constraint is visible but not very consistent. When more cavitation is allowed on the pressure side the algorithm is able to find better designs that dominate the original designs in both objectives. The algorithm also extends the front of Pareto optimal solutions, showing more extreme designs. However, the fronts are not very consistent when compared to each other. For the most strict constraint the algorithm

sometimes comes up with designs that are better than designs for a more relaxed constraint.

## 8.2 Recommendations

The implementation in Matlab showed that the optimiser can add useful information to the designer to asses cavitation performance of a design and to make a better decision regarding the trade-off between cavitation and efficiency. A further implementation should consider dedicated software like *modeFrontier* to make the implementation more robust and transparent to other users. This would also allow for an easier connection with other programs or even an integrated environment like *Quaestor* that can provide an objective evaluation.

To reduce computation time of one simulation, the application of parallel computation on multiple workstations could be considered. Most of the time is spend evaluating the objective function and as these are stand alone computations, they can easily be distributed over other processors.

The inability of the algorithm to converge can be solved by trying to implement different forms and combinations of crossover, mutation and selection mechanisms. The increased complexity of the simulations makes this a so called *rotated problem* which means that the variables are not simply related to the objective functions anymore. Some alternative crossover and mutation algorithms suited to these kind of problems can be found in Iorio and Li [18].

The results are influenced by the interpretation of the cavitation data. The lack of resolution in the cavitation length criteria (only measurable in whole percentages) should be removed. Furthermore the cavitation area and length are both relative to the blade area and the chord length while these two are part of the (changing) geometry. It would be better to relate this to a fixed parameter like the propeller disc area assuming that the diameter is fixed.

## 8.3 Outlook

In general designers want to apply more and more complex models to evaluate or to optimise their designs. This multiobjective optimisation tool gives the possibility to do so because it allows for an easy connection with the software/programs that evaluate a design. This 'black box' approach allows for the implementation of existing programs without the need to fiddle around in the source code. This approach also makes implementation of distributed simulations over multiple workstations fairly simple.

There are a number of other applications that are interesting to investigate which are too difficult or too complex. Cavitation inception speed could be optimised together with the objective of maximising efficiency and the extent of cavitation on the blades. Is there

a trade-off between increasing the inception speed and the amount of cavitation on the blades? If the inception speed could be increased, does this result in more excessive cavitation once it occurs? Hull form optimisation is another field where this algorithm could be applied. A trade-off could be found for two different speeds. A patrol craft spends most of its time at cruising speed but should be able to obtain a top speed as high as possible with the same hull form.

In the field of optimisation algorithms, research is ongoing to improve the efficiency of an algorithm. Engineering problems mostly involve costly objective function evaluations and techniques are developed to reduce the amount of necessary evaluations while obtaining the same results. The application of surrogate models that approximate the outcome of an objective function evaluation are used to guide the algorithm to converge.

# Bibliography


[1] John D. Anderson. *Fundamentals of Aerodynamics, second edition.* McGraw-Hill, Inc, New York, 1991. [34]

[2] Daniel Ashlock. *Evolutionary Computation for Modeling and Optimization.* Springer, 233 Spring Street, New York, NY 10013, USA, 2006. ISBN: 978-0-387-22196-0. [20]

[3] Frans Assenberg, W. van Gent, Gert Kuiper, and B. Luttmer. *ANPRO Lifting surface propeller analysis program.* Maritime Research Institute Netherlands, P.P. Box 28, 6700 AA, Wageningen, The Netherlands, version 00 edition, 2000. Reference and user's manual. [14]

[4] Ernesto Benini. Multiobjective design optimization of b-screw series propeller using evolutionary algorithms. *Marine Technology*, 40(4):229–238, October 2003. [2]

[5] Volker Bertram. *Practical Ship Hydromechanics.* Butterworth-Heinemann, August 14 2000. [3]

[6] L.C. Burrill and A. Emerson. Propeller cavitation: further tests on 16 in. propeller models in the king's college cavitation tunnel. *Trans. NECIES*, 79:295–320, 1963. [18]

[7] Mario Caponnetto. A new propeller design method for fast planing boat applications. In *Fast Transportation at Sea (FAST)*, pages 174–185, 1997. [2]

[8] John S. Carlton. *Marine Propellers and Propulsion.* Butterworth Heinemann, 1994. [16]

[9] David W. Corne, Nick R. Jerram, Joshua D. Knowles, and Martin J. Oates. PESA-II: Region-based selection in evolutionary multiobjective optimization. In Lee Spector, Erik D. Goodman, Annie Wu, W. B. Langdon, Hans-Michael Voigt, Mitsuo Gen, Sandip Sen, Marco Dorigo, Shahram Pezeshk, Max H. Garzon, and Edmund Burke, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 283–290, San Francisco, California, USA, 7-11 2001. Morgan Kaufmann. [21]

[10] J. Dang. Improving cavitation performance with new blade sections for marine propellers. *International Shipbuilding Progress*, 51(4):353–376, April 2004. [28]

[11] Kalyanmoy Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, June 2001. ISBN-10: 047187339X ISBN-13: 978-0471873396. [10]

[12] Kalyanmoy Deb and Ram Bhushan Agrawal. Simulated binary crossover for continuous search space. *Complex Systems*, 9:115–148, 1995. [24]

[13] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable test problems for evolutionary multi-objective optimization. Technical Report 112, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 2001. [-]

[14] Kalyanmoy Deb, Amrit Pratab, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions On Evolutionary Computation*, 6(2):182–197, April 2002. [21, 38, 39, 41]

[15] R.W.L. Gawn. Effect of pitch and blade width on propeller performance. *Transactions of the Royal Institution of Naval Architects*, 95:157–93, 1953. [1, 13]

[16] J.A. Geurst. Linearized theory of two-dimensional cavity flows. Master's thesis, Delft University of Technology, 1966. [19]

[17] J. Holtrop. An introduction to propeller design. In *Developments in the Design of Propulsors and Propulsion Systems*, volume 34 of *Wegemt School*. Wegemt School, 2000. [35]

[18] Antony W. Iorio and Xiaodong Li. Rotated test problems for assessing the performance of multi-objective optimization algorithms. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 683–690, New York, NY, USA, 2006. ACM. ISBN 1-59593-186-4. doi: http://doi.acm.org/10.1145/1143997.1144118. ISBN: 1-59593-186-4. [69]

[19] ITTC. Propeller models terminology and nomenclature for propeller geometry, 1999. International Towing Tank Conference. [27]

[20] W.H. Auf'm Keller. Enige aspecten bij het ontwerpen van scheepsschroeven. *Schip en Werf*, 24(33):658–663, December 1966. in Dutch, 33ste jaargang, nummer 24. [1, 18]

[21] Gert Kuiper. *The Wageningen Propeller Series*. Number Publication 92-001. MARIN, Haagsteeg 2, Wageningen, May 1992. Published on the occasion of its 60th anniversary. [1, 2, 14]

[22] Gert Kuiper and Stuart D. Jessup. A propeller design method for unsteady conditions. *SNAME Transactions*, 101:247–273, September 1993. [28]

[23] Daniel Kunkle. A summary and comparison of moea algorithms. unpublished, May 2005. [21]

[24] H. W. Lerbs. Moderately loaded propellers with a finite number of blades and an arbitrary distribution of circulation. *SNAME Transactions*, 60:–, 1952. [4]

[25] Lloyd's Register of Shipping. *Rules and Regulations for the Classification of Ships*, volume 5, chapter Chapter 7 - Propellers, pages 1–5. Lloyd's Register of Shipping, 1999. [29]

[26] R.N. Newton and H.P. Rader. Performance data of propellers for high speed craft. *Transactions of the Royal Institution of Naval Architects*, 103:93–118, 1961. [13]

[27] P. van Oossanen. Calculation of performance and cavitation characteristics of propeller including effects of non-uniform flow and viscosity. Technical Report 457, NSMB, Publ. No. 457, 1974. [35]

[28] C.P. Pouw. Multi-objective design optimization of marine propellers. Definition study, MARIN, March 2007. [37]

[29] D.T. Sandwell. Biharmonic spline interpolation of geos-3 and seasat altimeter data. *Geophysical Research Letters*, 14(2):139–142, Februari 1987. [43]

[30] Jyh-bin Suen and Jen-shiang Kouh. Genetic algorithms for optimal series propeller design. *Marine Technology III*, 42:237–246, 1999. [1]

[31] L. Troost. Open water test series with modern propeller forms. *Transactions of the North East Coast Inst. of Engineers and Shipbuilders*, (33):321, 1938. [13]

[32] W.P.A. van Lammeren, J.D. van Manen, and M.W.C. Oosterveld. The wageningen b-screw series. *Trans. SNAME*, 77:19–69, 1969. [14]

[33] H.C.J. van Wijngaarden, J. Bosschers, G.J.D. Zondervan, A.R. Starke, and J. Holtrop. Development of hybrid procedures for the prediction of propeller cavitation induced excitation and erosion. draft report 15116-16-CP, MARIN, June 2003. A 'Nederland Maritiem Land'-project. [49]

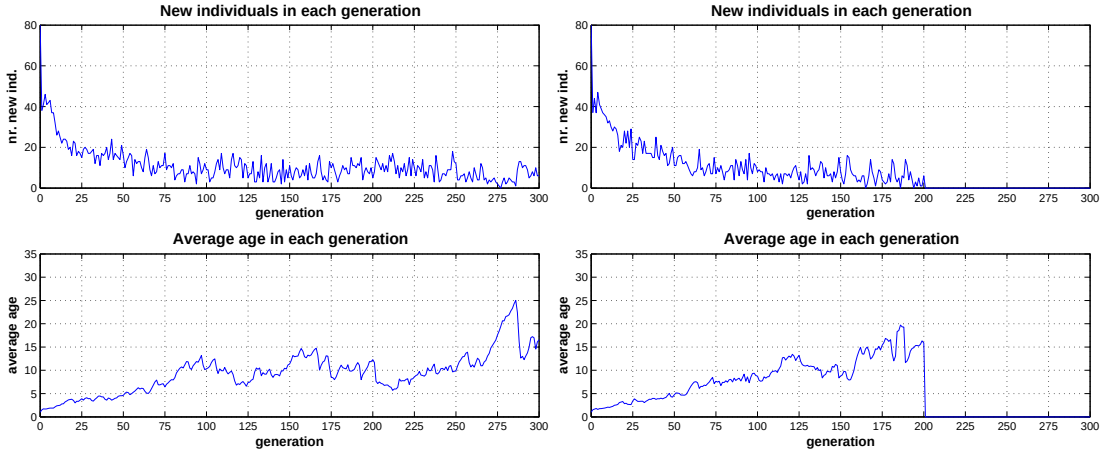
[34] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Zurich, Switzerland, 2001. [21]

[35] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of multiobjective evolutionary algorithms: Emperical results. *Evolutionary Computation*, 8(2):173–195, June 2000. [38]

[36] Zondervan, Gert Jan and Holtrop, Jan. Application of advanced sectional profiles in the design of propulsors and ship appendages. In *Propellers & Shafting Proceedings, Virginia Beach, USA*, September 2000. [28]

# Appendix A

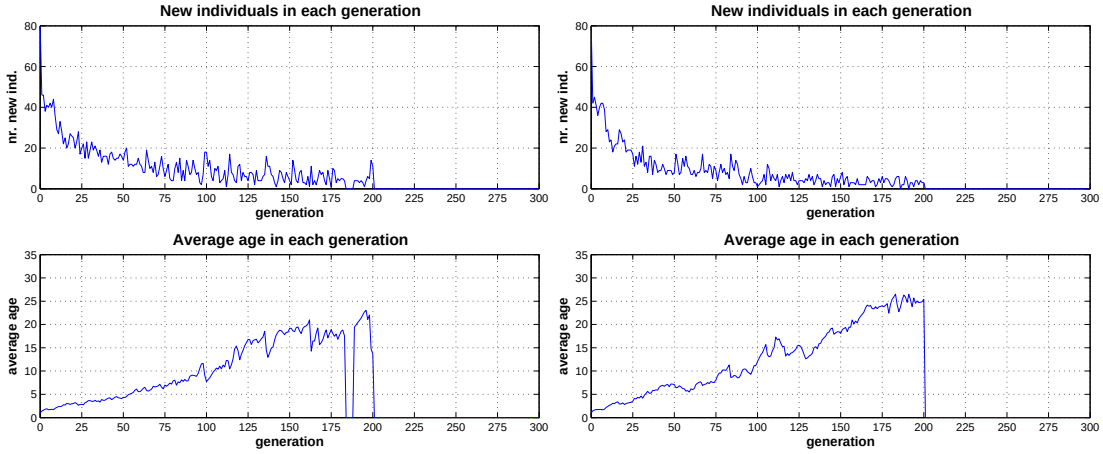
# Convergence visualisation - cavitation length




(a) 0% allowed PS sheet cavitation



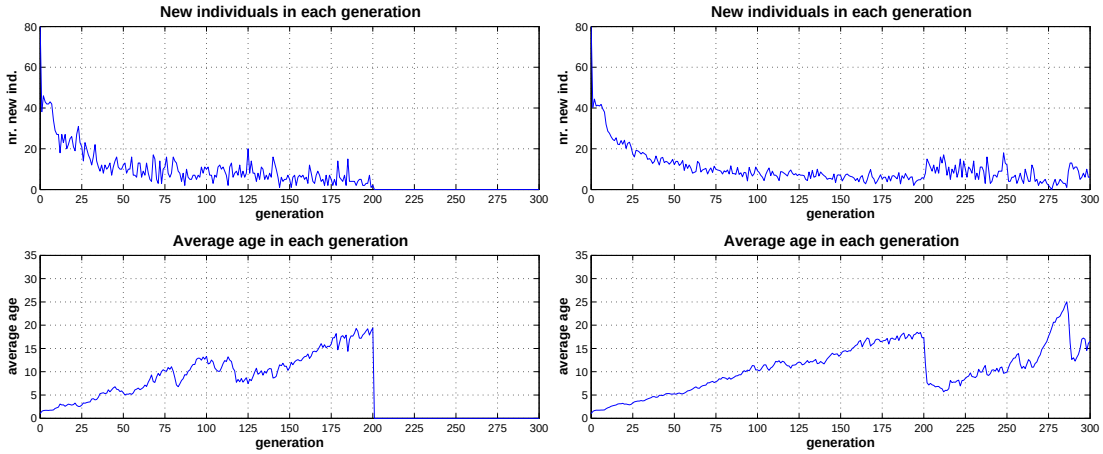

(b) 5% allowed PS sheet cavitation




(c) 10% allowed PS sheet cavitation




(d) 15% allowed PS sheet cavitation




(e) 20% allowed PS sheet cavitation



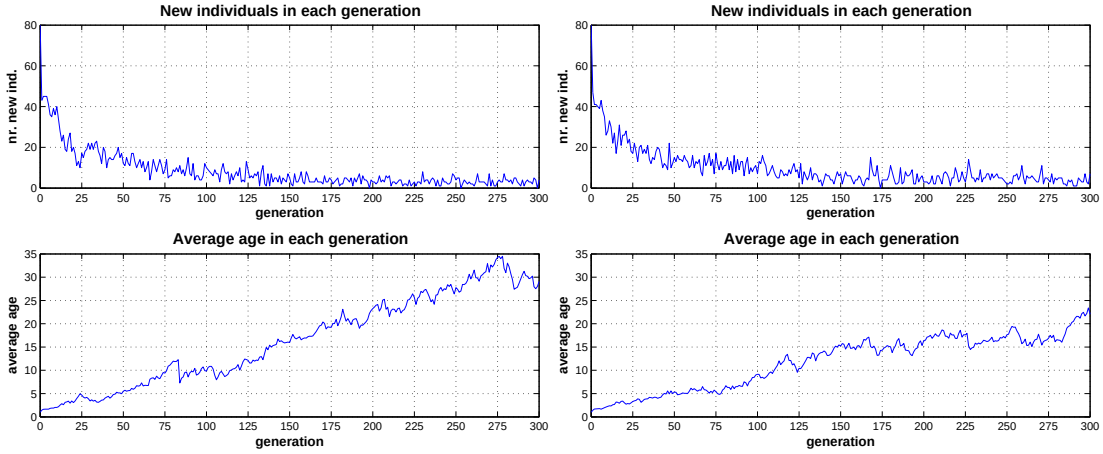

(f) averaged results



**Figure A.1:** overview for cavitation length as criteria

# Appendix B

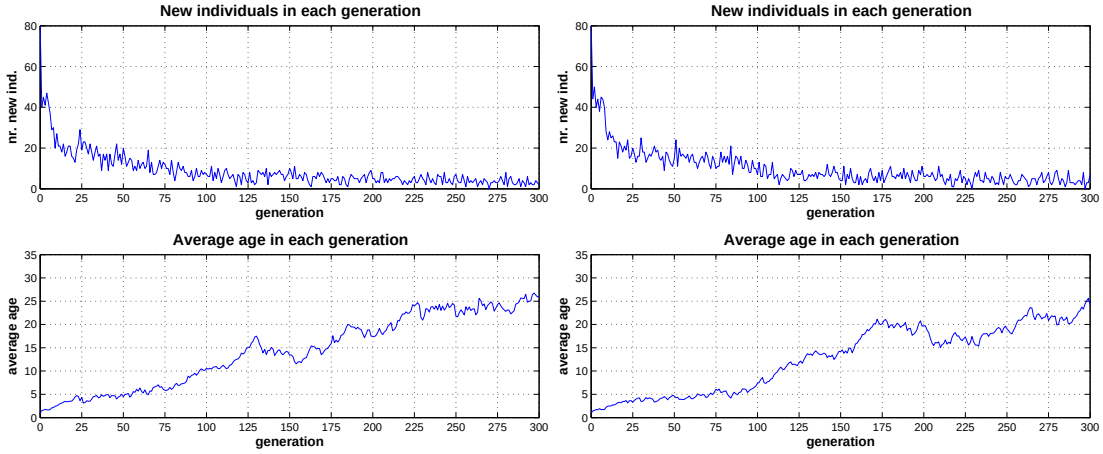
# Convergence visualisation - cavitation area




(a) 0% allowed PS sheet cavitation



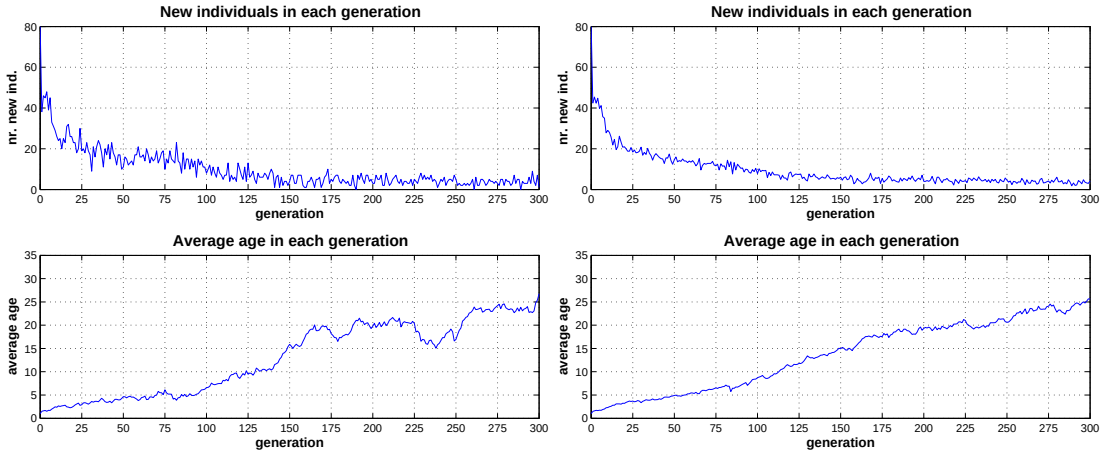

(b) 5% allowed PS sheet cavitation




(c) 10% allowed PS sheet cavitation




(d) 15% allowed PS sheet cavitation




(e) 20% allowed PS sheet cavitation




(f) averaged results



**Figure B.1:** overview for cavitation area as criteria

# Appendix C

# Propeller design visualisation - cavitation length

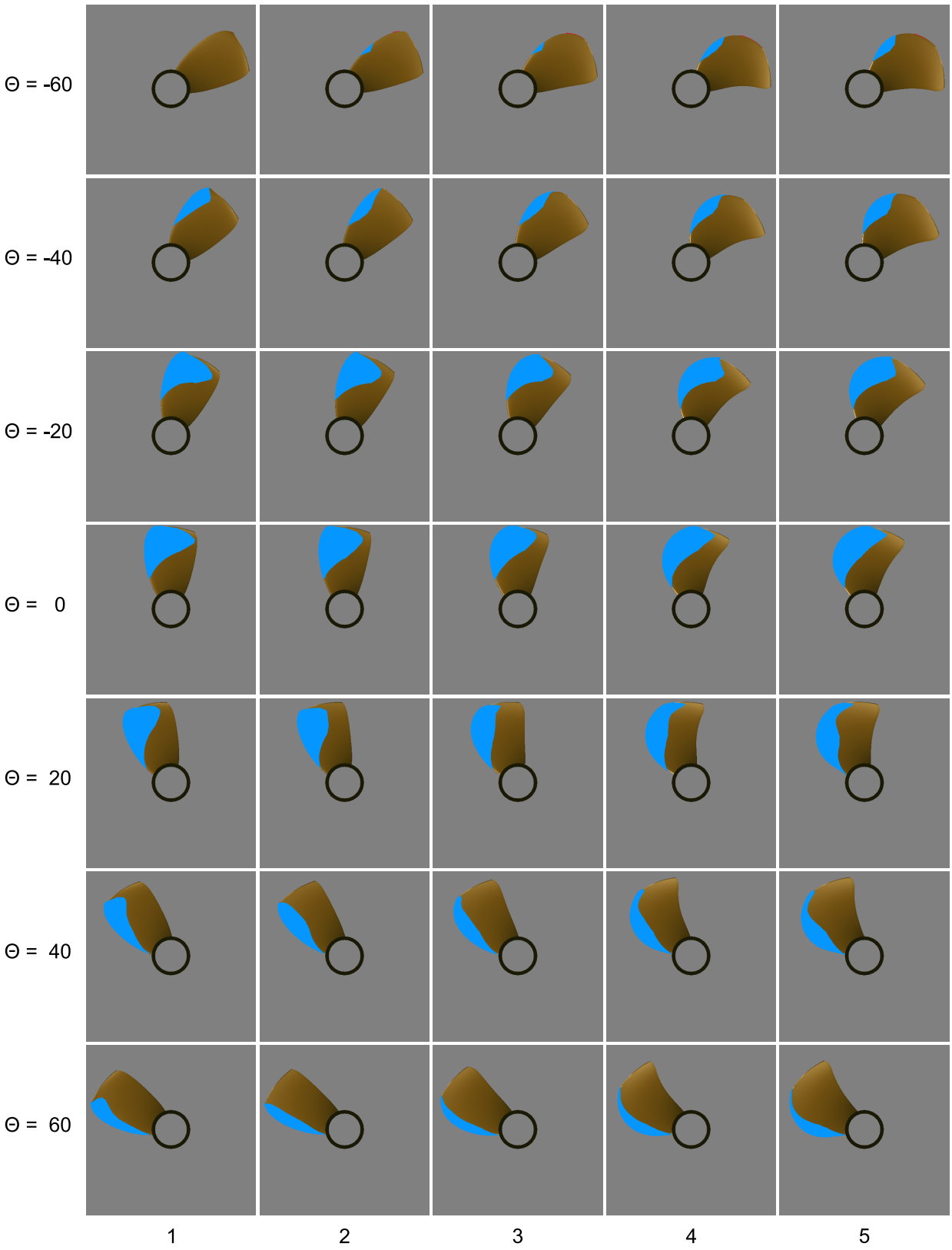



**Figure C.1:** Overview of designs along the Pareto front

# Appendix D

# Propeller design visualisation - cavitation area

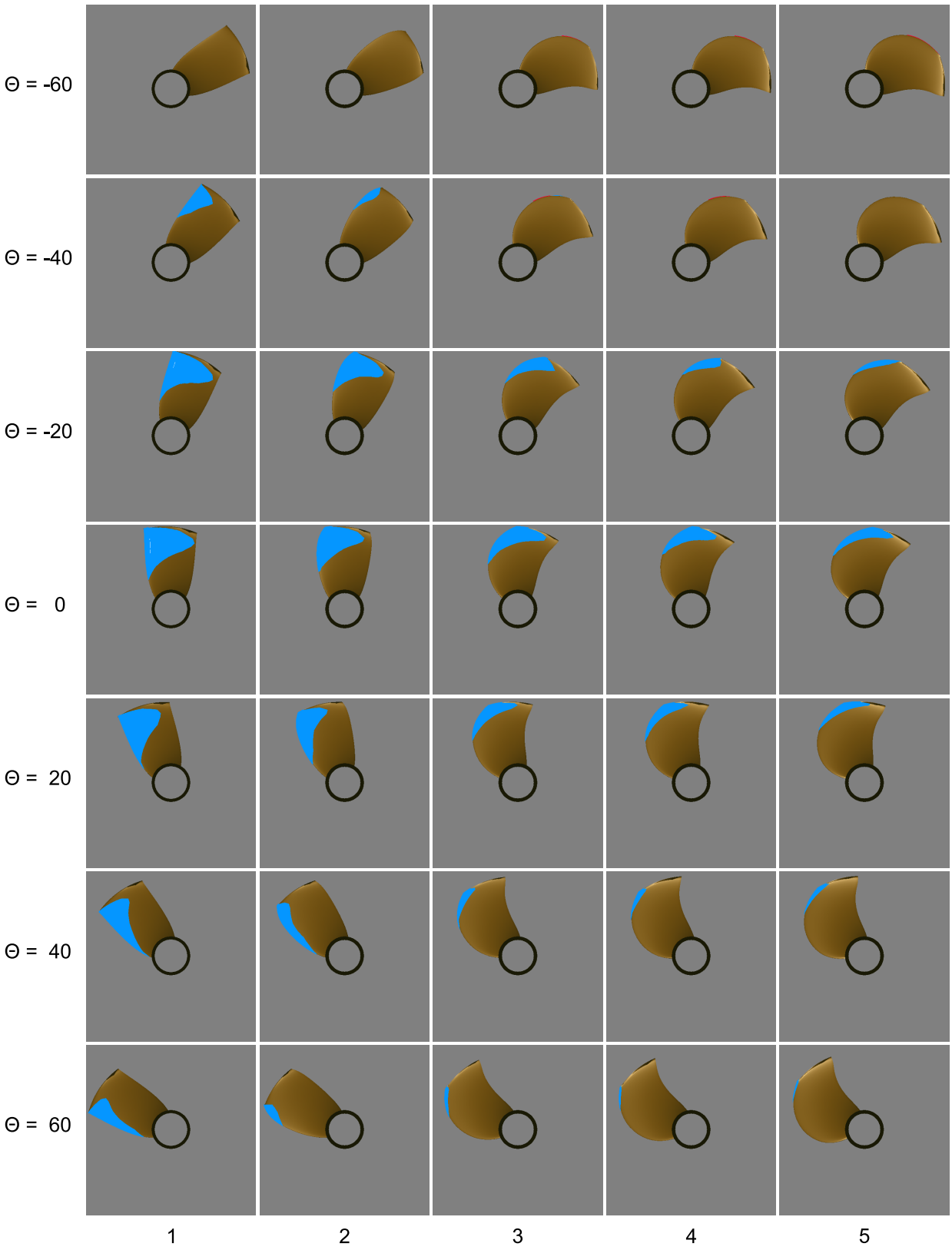



**Figure D.1:** Overview of designs along the Pareto front