

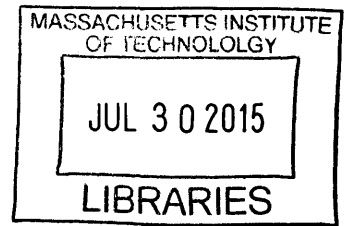
**A Parametric Modelling Tool for High Speed  
Displacement Monohulls**

by

Mert Timur

B.S., Turkish Naval Academy (2010)

**ARCHIVES**



Submitted to the Department of Mechanical Engineering  
in partial fulfillment of the requirements for the degree of

Master of Science in Naval Architecture and Marine Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

© Massachusetts Institute of Technology 2015. All rights reserved.


  
**Signature redacted**

Author .....

Department of Mechanical Engineering  
May 13, 2015

**Signature redacted**

Certified by .....

  
Stefano Brizzolara  
Research Scientist and Assistant Director for Research at MIT Sea  
Grant  
Thesis Supervisor

  
**Signature redacted**

Accepted by .....

  
David E. Hardt  
Chairman, Department Committee on Graduate Students



# **A Parametric Modelling Tool for High Speed Displacement Monohulls**

by

Mert Timur

Submitted to the Department of Mechanical Engineering  
on May 13, 2015, in partial fulfillment of the  
requirements for the degree of  
Master of Science in Naval Architecture and Marine Engineering

## **Abstract**

In ship design projects, it is of utmost importance to investigate a wide range of options during the concept design phase in order to determine which one best suits to the requirements. Although, keeping the concept design phase shorter in order to be competitive in the market is as important. The chances for a shipyard to win a contract would surely increase with a proposed design whose performances are demonstrated through a systematic evaluation of alternative solutions. However, the number of the design alternatives is inversely proportional to the time span of concept design for each alternative. The detailed evaluations at this stage can only be performed with CFD (Computational Fluid Dynamics) and FE (Finite Element) tools, and both require a complete representation of the ship hull geometry. So, only having a faster hull form generation tool would enable the designer to evaluate more options.

It is possible to achieve rapid geometry generation through fully parametric modeling. Fully parametric hull modeling is the practice of creating the entire hull shape definition only from form parameters, without the need for offset data or predefined lines plan. In this thesis a fully parametric modeling tool, PHull, is developed using Java programming language for rapid geometry generation of high speed displacement monohulls, in order to be used in hydrodynamic optimization process. The results from the validation cases, FFG-7 and ATHENA Model 5365, are presented.

Thesis Supervisor: Stefano Brizzolara

Title: Research Scientist and Assistant Director for Research at MIT Sea Grant



## Acknowledgments

First of all I would like to thank the Turkish Naval Forces and the Turkish taxpayer, who gave me the opportunity to attend MIT and financially supported my studies.

I am indebted to my advisor and supervisor, Prof. Stefano Brizzolara, for his teaching, guidance, and support throughout my thesis. His passion and excitement about Naval Architecture is always a great influence. I'm also grateful to Luca Bonfiglio and Giuliano Vernengo for their contribution.

I would like to express my deepest gratitudes to RADM Nurhan Kahyaoglu, Turkish Navy, Retired; CDR Noyan KILINC, Turkish Navy, Retired; CDR Bahadir Aytac, Turkish Navy; and CDR Zafer Elcin, Turkish Navy, for helping me to develop my background in naval architecture and teaching me to think like an engineer.

Next, I would like to thank to CAPT Mark Thomas, USN, Retired; CDR Jerod Ketcham, USN; and CDR Weston Gray, USN; for all the valuable knowledge and experience they shared during my education here at MIT.

Last but not least, I would like to thank my parents Nazmi and Gulnur, my sister Duygu, and my fiancée Selin. I wouldn't be where I am without their devoted support and patience.

"Akıl ve mantığın çözümlenemeyeceği mesele yoktur."

"No problem exists that cannot be solved with mind and logical reasoning."

**Mustafa Kemal ATATÜRK**



# Contents

<b>1</b>	<b>Introduction</b>	<b>17</b>
1.1	Motivation . . . . .	17
1.2	Goals . . . . .	18
1.3	State of the Art in Parametric Modeling Tools for Ship Hulls . . . . .	19
1.4	Organization of the Thesis . . . . .	21
<b>2</b>	<b>Introduction to B-Splines, NURBS (Non-Uniform Rational B-Spline), and iGeo Computational Design Library</b>	<b>23</b>
2.1	Bézier Curves and Surfaces . . . . .	24
2.1.1	Bernstein Polynomials . . . . .	24
2.1.2	Definition and Properties of Bézier Curves . . . . .	25
2.2	B-Spline Curves and Surfaces . . . . .	26
2.2.1	Definition and Properties of B-Spline Curves . . . . .	27
2.3	NURBS (Non-Uniform Rational B-Spline) Curves and Surfaces . . . . .	29
2.4	B-Splines and NURBS Representations in Hull Design . . . . .	32
2.5	iGeo Computational Design Library . . . . .	33
<b>3</b>	<b>PHull, The Parametric Modeling Tool</b>	<b>35</b>
3.1	How Does PHull Work? . . . . .	36
3.2	Inputs for PHull . . . . .	38
3.3	Control Curves . . . . .	40
3.3.1	Underwater Profile Curve . . . . .	40
3.3.2	Design Waterline Curve (DWL) . . . . .	43

3.3.3	Edge of the Weather Deck Curve . . . . .	47
3.3.4	Deadrise Angle Curve . . . . .	52
3.3.5	Flare Angle at DWL Curve . . . . .	55
3.3.6	Sectional Area Curve (SAC) . . . . .	55
3.3.7	Flare Angle at the Weather Deck Curve . . . . .	60
3.3.8	Width of the Keel Curve . . . . .	60
3.4	Representing the Hull Form Using the Control Curves . . . . .	62
3.4.1	Cross Section Curves . . . . .	64
3.4.2	Surface Generation . . . . .	70
3.5	Validation of PHull with FFG-7 and ATHENA Hull Forms . . . . .	73
3.5.1	Comparison of the Profile Curves . . . . .	74
3.5.2	Comparison of the DWL Curves . . . . .	76
3.5.3	Comparison of the Edge of the Weather Deck Curves . . . . .	77
3.5.4	Comparison of the Cross Section Curves . . . . .	78
<b>4</b>	<b>Parametric Hull Form Modifications with PHull</b>	<b>85</b>
4.1	Simple Examples of Parametric Hull Form Modification with PHull on FFG-7 . . . . .	86
4.2	A Simple Approach for the Systematic Hull Form Generation with Shape Variation Using PHull . . . . .	89
4.3	Parametric Modifications for Hydrodynamic Optimization Using PHull	95
<b>5</b>	<b>Integration with CFD and Automatic Optimization Procedure</b>	<b>97</b>
5.1	Integration with the Panel Method . . . . .	97
5.2	Integration with the Optimization Algorithm . . . . .	99
5.3	Hydrodynamic Optimization of FFG-7 Hull . . . . .	101
<b>6</b>	<b>Conclusions and Recommendations</b>	<b>109</b>
6.1	Conclusions . . . . .	109
6.2	Recommendations for Future Work . . . . .	110



<b>A User Guide for PHull</b>	<b>111</b>
A.1 Preparation of the Input File . . . . .	111
A.2 Representation of an Existing Hull Using the Interpolated and Approx- imated Control Curves Together . . . . .	117
A.3 Reproducing the Hull Form Using Only Approximated Control Curves	118



# List of Figures

2-1	3 <sup>rd</sup> Degree Cubic Bézier Curves . . . . .	26
2-2	Cubic B-spline Curve . . . . .	27
2-3	A NURBS Curve . . . . .	30
2-4	Influence of Weights . . . . .	31
3-1	An Underwater Profile Curve Example . . . . .	40
3-2	Underwater Stem Curve . . . . .	41
3-3	Underwater Stem Curve with Control Vertices . . . . .	42
3-4	Keel Curve . . . . .	42
3-5	Keel Curve with Control Vertices . . . . .	43
3-6	DWL Curve from Forward . . . . .	44
3-7	DWL Curve from Top . . . . .	45
3-8	Forward Part of the DWL Curve . . . . .	45
3-9	Aft Part of the DWL Curve . . . . .	47
3-10	Edge of the Weather Deck Curve for FFG 7 (Generated by PHull) . .	49
3-11	Projection of Edge of the Weather Deck Curve (Thick Blue Line) on XZ Plane . . . . .	49
3-12	Which Part Does Each Control Curve Control on a Cross Section? . .	52
3-13	Deadrise Control Curve with Interpolation Method . . . . .	53
3-14	Deadrise Control Curve with Approximation Method . . . . .	54
3-15	Flare Angle Control Curve with Interpolation Method . . . . .	56
3-16	Flare Angle Control Curve with Approximation Method . . . . .	56
3-17	Sectional Area Curve with Interpolation Method . . . . .	57

3-18 SAC with Approximation in Two Pieces . . . . .	58
3-19 Flare Angle at the Weather Deck Curve with Interpolation Method . . . . .	61
3-20 Flare Angle at the Weather Deck Curve with Approximation Method . . . . .	61
3-21 Keel Width Curve with Interpolation Method . . . . .	63
3-22 Keel Width Curve with Approximation Method . . . . .	63
3-23 Underwater and Above Water Parts of a Cross Section Curve . . . . .	65
3-24 Underwater Cross Section Curve . . . . .	66
3-25 Intermediate Control Point as the Intersection of Two Lines . . . . .	67
3-26 Intermediate Control Point as Obtained from the Cubic Spline Fit . . . . .	68
3-27 Comparison of the Cross Section Forms for Bow and Amidships . . . . .	68
3-28 Above Water Cross Section Curve . . . . .	69
3-29 Transition Region between Two Parts of the Cross Section Curve . . . . .	70
3-30 Stem and Main Body Surfaces . . . . .	72
3-31 Transom Surfaces . . . . .	72
3-32 ATHENA Hull . . . . .	74
3-33 FFG-7 Hull . . . . .	75
3-34 Profile Curve Comparison for ATHENA Model 5365 . . . . .	75
3-35 Profile Curve Comparison for FFG-7 . . . . .	76
3-36 DWL Curve Comparison for ATHENA Model 5365 . . . . .	76
3-37 DWL Curve Comparison for FFG-7 . . . . .	77
3-38 Edge of the Weather Deck Curve Comparison for ATHENA Model 5365 . . . . .	77
3-39 Edge of the Weather Deck Curve Comparison for FFG-7 . . . . .	78
3-40 Cross Section Curve Comparisons for ATHENA Model 5365. Sectional curves are colored in red (original) and blue (Phull). . . . .	79
3-41 Cross Section Curve Comparisons for Forward Part of FFG-7. Sec- tional curves are colored in red (original) and blue (Phull). . . . .	80
3-42 Cross Section Curve Comparisons for Aft Part of FFG-7. Sectional curves are colored in red (original) and blue (Phull). . . . .	81
3-43 Surfaces for the ATHENA Model 5365 as Generated by PHull . . . . .	82

3-44	Comparison of the Original FFG-7 Model (top) and the FFG-7 Model Obtained with PHull . . . . .	82
4-1	DWL Entrance Angle Parameter Modification . . . . .	87
4-2	Keel Rise Point Parameter Modification . . . . .	87
4-3	Longitudinal Position of the Section with the Maximum Beam Param- eter Modification . . . . .	88
4-4	Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 1 . . . . .	91
4-5	Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 2 . . . . .	92
4-6	Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 3 . . . . .	93
4-7	Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 4 . . . . .	93
5-1	Wave Pattern Plot of FFG-7 Hull Created with the Output of the Panel Method . . . . .	98
5-2	General Outline of the Automatic Optimization Process with PHull .	101
5-3	Plot of the Objective Function Values of the Hulls That Were Evalu- ated in the Optimization . . . . .	105
5-4	Comparison of the Sectional Area Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue) (note that forward part is to the left) . . . . .	106
5-5	Comparison of the Deadrise Angle Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue) . . . . .	106
5-6	Comparison of the DWL Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue) . . . . .	107
5-7	Comparison of the Profile Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue) . . . . .	107

5-8 Comparison of the Cross Section Curves of the Original FFG-7 Model  
(red) and the Optimum Hull (blue) . . . . . 108

# List of Tables

3.1	.csv Input File for PHull . . . . .	39
3.2	Parameters for the Profile Curve . . . . .	44
3.3	Parameters for the DWL Curve . . . . .	48
3.4	Parameters for the Edge of the Weather Deck Curve . . . . .	51
3.5	Input for Deadrise Angle Curve Interpolation . . . . .	52
3.6	Input for Deadrise Angle Curve with Approximation Method . . . . .	54
3.7	Input for Flare Angle Curve with Approximation Method . . . . .	57
3.8	Input for SAC with Approximation in Two Pieces . . . . .	59
3.9	Input for SAC with Approximation as a Single Piece . . . . .	59
3.10	Input for Flare Angle at the Weather Deck Curve with Approximation Method . . . . .	62
3.11	Input for Keel Width Curve with Approximation Method . . . . .	64
4.1	Parameters that Are Used in the Systematic Hull Form Generation . . . . .	90
4.2	Parameters that Are Used for Increasing Displacement in the Forward Part . . . . .	91
4.3	Parameters that Are Used for Decreasing Displacement in the Forward Part . . . . .	91
4.4	Parameters that Are Used for Increasing Displacement in the Aft Part . . . . .	92
4.5	Parameters that Are Used for Decreasing Displacement in the Aft Part . . . . .	92
4.6	Parameters that Are Used for Increasing Displacement on Both Parts . . . . .	94
4.7	Parameters That Are Used for Decreasing Displacement on Both Parts . . . . .	94

5.1	Panel Method Outputs and Total Resistance Calculation Results for Integration Test Case . . . . .	99
5.2	Comparison of the Set of Free Parameters for the Original and the Optimum Hulls . . . . .	104
A.1	Complete Set of Parameters for PHull . . . . .	113
A.2	Parameters of Blue Section with Explanations . . . . .	114
A.3	Parameters of Yellow Section . . . . .	115
A.4	Parameters of Orange Section with Explanations . . . . .	116
A.5	Parameters of Green Section . . . . .	117



# Chapter 1

## Introduction

### 1.1 Motivation

In ship design projects, it is of utmost importance to investigate a wide range of options during the concept design phase in order to determine which one best suits to the requirements. Although, keeping the concept design phase shorter in order to be competitive in the market is as important. The chances for a shipyard to win a contract would surely increase with a proposed design whose performances are demonstrated through a systematic evaluation of alternative solutions. However, the number of the design alternatives is inversely proportional to the time required to develop the concept design of each alternative. The detailed evaluations at this stage can only be performed with CFD (Computational Fluid Dynamics) and FE (Finite Element Methods) tools, and both require a complete representation of the ship hull geometry [1]. So, only having a faster hull form generation tool would enable the designer to evaluate more options.

It is possible to achieve rapid geometric modeling through fully parametric design and the studies within the last few decades has shown that hull forms can be generated and modified efficiently with this approach [1] [2] [3] [9] [10]. Along with the rapid hull form generation, fully parametric modeling approach also has the potential to integrate CASHD (Computer Aided Ship Hull Design) and CFD in order to be used in the parametric hydrodynamic optimization process, which would otherwise work

apart from each other with no direct feedback [2].

## 1.2 Goals

In parametric hull form optimization hull is controlled by a set of parameters, and the optimization algorithm works on these parameters to achieve better results according to its objective function. The objective function is the attribute we want to minimize or maximize. Minimizing the total ship resistance for a certain hull speed is a good example, here the objective function is the total ship resistance. If the requirement is to minimize the resistance for a range of speeds, then the problem will be to minimize a multi-valued objective function. The fully parametric modeling tool developed in this thesis aims to provide the appropriate geometric representations of navy ships hull forms to be used in automatic optimization procedures. Using these parametric representations the multi-objective global minimization algorithms can work their way to the optimal combination of parameters.

The goal of this thesis is to develop an efficient fully parametric modeling tool for rapid geometric generation of high speed displacement monohulls, that will be used in hydrodynamic optimization process. Of course, the output of the parametric modeling tool has to meet some requirements.

- The resulting hull form must have fair curves that would result in fair surfaces.
- The format of the output must be appropriate for the intended CFD tools.
- The way that the hull forms being represented in the modeling tool must give optimization algorithm the chance to modify the hull form easily.

Even though the primary concern is to create the parametric modeling tool, CFD analysis and hydrodynamic optimization of the hull forms generated with the tool are also the focus of this thesis. The hull representations from the modeling tool have been evaluated with a low order Panel Method that only requires a number of cross section curves and the stem curve. However, other CFD tools are currently under

development and in use at the MIT iShip lab require the 3D surface representation of the hull. In fact, the tool also has the ability to generate the hull surfaces that can be analyzed in RANS (Reynolds Averaged Navier-Stokes) models. Panel method was chosen for this work mostly because of the time constraints. For the hydrodynamic optimization, a global convergence, single objective, evolutionary algorithm was used.

### 1.3 State of the Art in Parametric Modeling Tools for Ship Hulls

The purpose of this section is to give a very brief summary about the history of hull form representation methods, and then to talk about the recent applications in the field.

Methods for ship hull form design and representation have a long history. It is possible to see the usage of the simple geometric shapes such as ellipses and circles in ancient and medieval ships. Even though the mathematical representation was not of interest by then, the introduction of free form hull shapes to naval architecture is around 1700. Later the use of a family of parabolas for defining the ship curves was suggested by Chapman around 1760. Then D. W. Taylor laid the groundwork for the mathematical ship lines by using mathematical functions to define hull shapes in the beginning of 20<sup>th</sup> century, he used 5<sup>th</sup> degree polynomials to present sectional area curve and the waterlines. He also used form parameters to generate systematic variations of the existing hull forms, an approach which would be widely used later in the field [3]. Types of parameters he used were:

- Ending positions of the curves
- Slopes at the end of the curves
- Differential form parameters (curvature)
- Integral form parameters (area under the curve) [2] [3].

The advent of computers in Naval Architecture was another milestone that would result in a new research field known as the Computer Aided Ship Hull Design (CASHD) [2]. Another important event, that is quite relevant to this work, was the introduction of the B-spline and Non-Uniform Rational B-spline (NURBS) curves into Naval Architecture programs in the eighties and nineties, that would provide the ability to modify curves locally. By means of these new technologies, numerous methods and techniques were introduced for hull representation, hull design and hydrodynamic hull optimization in the last few decades. Some of these methods and some applications are briefly introduced in the following part in order to show the state of the art in parametric modeling tools for ship hulls.

The method developed by Harries et al. [1] and Harries [2], which is part of the commercial FRIENDSHIP-Framework software today, uses planar B-spline curves with intrinsic fairness to represent ship curves. This method generates longitudinal basic curves first, then the data for cross sections are derived from these basic curves, finally the hull surfaces are created through interpolation of these sections. The method that was introduced by Fuller et al [8] and recently reappraised by Yang and Bowers [9] in order to achieve hull form generation is also based on the longitudinal curves, but in this method these curves are defined as polynomials. Even though these two methods are similar on the basis, there are some other significantly different methods as well.

For example Yang et al. [10] introduced a combined local and global hull form modification approach, that is based on both the radial basis function interpolation and the sectional area curve. For global modifications sectional area curve can be modified by changing the slope at the end points. In order to achieve more subtle local modifications, using radial basis function interpolation is suggested by this method. The NURBS surface fitting method through genetic algorithms for ship hulls was suggested by Le and Kim [12].

The possibility of using an automated optimization procedure with the parametric generation of the geometries was proven to be effective by Brizzolara [4]. Furthermore Brizzolara and Vernengo developed a completely automatic parametric optimization

procedure for SWATH hull form using the FRIENDSHIP-Framework [5] [7]. The study of Vernengo et al. [6] showed the advantages of using the parametric modeling for the case of a frigate.

## 1.4 Organization of the Thesis

This thesis develops an alternative parametric modeling tool, that can be used for shape representation and ab initio design. It is intended to be used at early stage design in order to provide the designer a 3D model that he/she can evaluate to make accurate decisions.

The main two curve representation methods that are used by the parametric modeling tool, B-spline and NURBS representations, are discussed in Chapter 2 together with the the iGeo Computational Design Library, which provides the opportunity to use them in Java programming language. In Chapter 3 the parametric modeling tool itself, PHull, is introduced. The validation cases for PHull, FFG-7 and R/V ATHENA Model 5365, are also presented in this chapter. Consequently, Chapter 4 shows the hull modification abilities of PHull through some examples. Later in Chapter 5 the integration of the parametric modeling tool with CFD tool and the Optimization Algorithm is discussed with the FFG-7 example case. Finally, Chapter 6 includes the conclusions and recommendations for further studies.



## Chapter 2

# Introduction to B-Splines, NURBS (Non-Uniform Rational B-Spline), and iGeo Computational Design Library

For curve generation there are two different set of techniques, namely the interpolation and the approximation techniques. The interpolation techniques require the resulting curve to pass through all data points. An example for this type is the cubic spline. When the shape of the curve is known through a sufficiently high number of points (high sampling rate), then very good results can be obtained with the interpolation methods. Hydrofoils and aerofoils can be given as examples. But these methods do not suit well *ab initio* design problems, since an offset of the curve is not known. Ship hull design and car design are examples for the *ab initio* design problems. A curve generation technique which is well suited for this type was proposed by Pierre Bézier. In this technique the curve is defined by its defining polygon [3], and the curve approximates this polygon. Further developments on the Bézier representation resulted in B-spline and NURBS representations, which are even more advantageous for the *ab initio* design problems.

In this chapter Bézier, B-Spline, and NURBS representation techniques will be investigated to a certain level of detail, since these are the techniques that are used by the parametric modeling tool by means of iGeo Computational Design Library. Later the reason for choosing these shape representation techniques for hull generation will be discussed. And lastly iGeo Computational Design Library will be introduced to the reader.

There are many introductory and application oriented books about Bézier, B-Spline, and NURBS representation techniques. In this thesis the books of Rogers and Adams [13], Nowacki et al. [3], and Patrikalakis et al. [14] were used as main references.

## 2.1 Bézier Curves and Surfaces

A Bézier curve is a parametric curve that is determined by its defining polygon. These curves do not have many applications in naval architecture today, but they laid the groundwork for the B-splines and the NURBS, so they worth mentioning.

### 2.1.1 Bernstein Polynomials

Bernstein polynomials are the mathematical basis for Bézier curves. The definition of the Bernstein basis function is as follows:

$$B_{i,n}(t) = \frac{n!}{i!(n-i)!} (1-t)^{n-i} t^i, \quad i = 0, \dots, n. \quad (2.1)$$

Using an  $n^{th}$  degree Bernstein polynomial a function  $f(t)$  can be approximated in the following pattern:

$$B_n(t) = \sum_{i=0}^n B_{i,n}(t) f\left(\frac{i}{n}\right), \quad t \in [0, 1] \quad (2.2)$$

The basic properties of Bernstein basis functions and polynomials as they are given in Nowacki et al.[5 ] are presented below.



## Properties of Bernstein Basis Functions

- Positivity
- Cauchy's Relation (partition of unity)
- Recursion
- Symmetry

## Properties of Bernstein Polynomials

- Convergence
- Boundedness
- Linearity
- Variation Diminishing Property

Here, the accurate approximation of the arbitrary function  $f(t)$  is assured by convergence and boundedness properties. Linearity property assures that a linear function's Bernstein approximation is the linear function itself. And finally variation diminishing property assures the smoothing effect of Bernstein approximation [5].

### 2.1.2 Definition and Properties of Bézier Curves

Mathematical basis for Bézier representation corresponds to Bernstein basis. This means that the properties of Bernstein basis is inherent in the Bézier curves. A parametric  $n^{th}$  degree Bézier curve's mathematical definition is as follows:

$$P(t) = \sum_{i=0}^n V_i B_{i,n}(t), \quad t \in [0, 1] \quad (2.3)$$

where  $P(t)$  is the position on the curve as a function of parameter  $t$ ,  $V_i$  represents the control vertices, and  $B_{i,n}(t)$  represents the Bernstein basis. In Figure 2-1 two different examples of cubic Bézier curves with different set of control vertices are shown.

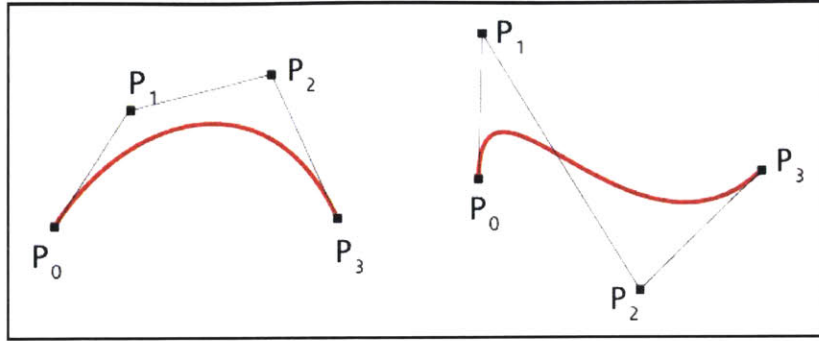


Figure 2-1: 3<sup>rd</sup> Degree Cubic Bézier Curves

For the case of Bézier curves, the number of control vertices is always one greater than the degree of the defining Bernstein basis function. For example from Figure 2-1 it can be seen that a 3<sup>rd</sup> degree Bézier curve has four control vertices. Again from Figure 2-1 it is easy to see that the curves follow their defining polygon's shape, and the end points of the curves are coincident with their first and last control vertices.

Unfortunately Bézier curves have two drawbacks for CASHD, both about their flexibility. First one is their global nature. This means that it's not possible to modify the curves locally, and every change that has been made would affect the curve globally. But in CASHD having the ability to control the curve locally is very important.

Secondly, the only way to increase or decrease a Bézier curve's degree is to respectively increase or decrease the number of control vertices it has. For example, in order to define a fifth degree Bézier curve six control vertices have to be used, which might not be a feasible option. So the order  $k$ , which is one greater than the degree, of the Bézier curves are fixed with the number of their control vertices [13].

## 2.2 B-Spline Curves and Surfaces

B-spline curves carries all the positive aspects of the Bézier curves without their drawbacks that were just mentioned. This makes B-spline representation very useful in CASHD applications. The B-spline representation was developed from Bézier representation by replacing the Bernstein basis by a polynomial spline basis [3]. On

special occasions B-spline basis can reduce to Bernstein basis. Figure 2-2 displays an example for the B-spline curves.

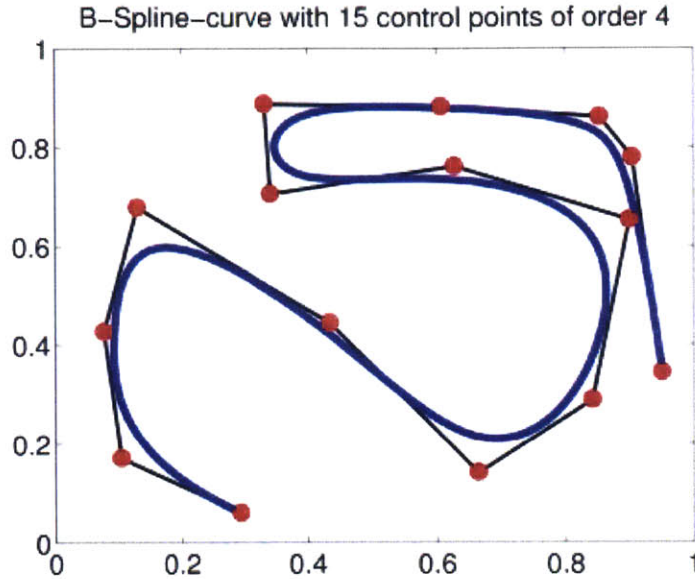


Figure 2-2: Cubic B-spline Curve

The 15 points you can see on Figure 2-2 are the control vertices, and if you connect these control vertices with straight lines, that would give you the defining polygon of the B-spline curve.

### 2.2.1 Definition and Properties of B-Spline Curves

B-spline curve's mathematical definition can be given as:

$$P(t) = \sum_{i=1}^{n+1} V_i N_{i,k}(t), \quad t_{min} \leq t \leq t_{max}, \quad 2 \leq k \leq n+1 \quad (2.4)$$

Here  $P(t)$  gives, just as in the Bézier curve's definition, the position on the curve with respect to parameter  $t$ ,  $V_i$  is the position vector for  $n+1$  control vertices,  $N_{i,k}(t)$  represents the B-spline basis function, and  $k$  is the order of this function. Order of a basis function is always one greater than its degree.

The general definition of a B-spline basis function according to Cox-deBoor recur-

sion formula is:

$$N_{i,1}(t) = \begin{cases} 1, & \text{if } x_i \leq t \leq x_{i+1} \\ 0, & \text{otherwise} \end{cases} \quad (2.5)$$

$$N_{i,k}(t) = \frac{(t - x_i)N_{i,k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t)N_{i+1,k-1}(t)}{x_{i+k} - x_{i+1}} \quad (2.6)$$

and the following expression is the vector of knots:

$$T = (t_0, t_1, \dots, t_m), \quad t_{i+1} \geq t_i \quad (2.7)$$

Knots are the junction points of the piecewise polynomials that form the B-spline curve, and the classification of a B-spline curve is determined by its vector of knots.

The reason why B-spline curves are so useful in the CASHD applications is self evident from their properties. Some of these properties, and their comparisons with Bézier curves where applicable, are presented here [13]:

- In B-spline curves a vertex is associated with its own basis function, thus can only affect the shape of the curve for a certain range of parameter value. As a result, non-global property is obtained, which means that local modifications on the representations can be achieved by using B-spline curves.
- Using B-spline basis also brings the ability to change the order (so the degree as well) of the curve without changing the number of control vertices. For example the curve in Figure 2-2 is a third degree B-spline curve with fifteen control points, which is not possible for a Bézier curve.
- Function  $P(t)$ , which gives the position along the curve as a function of parameter  $t$ , is continuous, and so are its derivatives until the  $(k - 2)^{th}$  order. Again giving an example from Figure 2-2, the curve here is third degree, which means that it's fourth order ( $k = 4$ ). So the derivatives of this curve would be continuous including the second,  $(k - 2)^{th}$ , derivative.
- A B-spline curve's order cannot be greater than the number of control vertices

it has. For Bézier curve's the order of the curve is always equal to number of control vertices. When this is the case, B-spline curve reduces to Bézier curve.

- A  $k^{th}$  order and  $p^{th}$  degree, where  $k = p + 1$ , B-spline curve with  $N = n + 1$  control vertices has  $S = N - k + 1$  segments. Since the knots are the junction points of these segments, the total number of knots  $q$  would be one less than the number of spans, so  $q = N - k$ . Each of the control vertices can only affect the shape of the curve for  $k$  spans. So with a number of excess control vertices local changes of the shape can be achieved [3].
- A B-spline curve's subsequent control vertices can be coincident. Multiple coincident vertices have significant effects on the form of the curve. For example having  $k$  coincident vertices would cause the curve to pass through this point. Having  $(k - 1)$  coincident vertices can be presented as another example. In this case curve would have a linear segment [3].
- Unlike Bézier curves, B-spline curves don't have to converge with the defining polygon at the end points. However, this can be achieved by having order  $k$  number of control vertices at the end points of the control polygon
- Number of oscillations of a B-spline curve about a straight line cannot be greater than its defining polygon's.
- The curve follows the shape of its defining polygon, and stays within its defining polygon's convex hull.
- Basis functions of B-spline curves are non-negative.

## 2.3 NURBS (Non-Uniform Rational B-Spline) Curves and Surfaces

Rational B-splines were derived from B-splines and first introduced by Versprille [3]. Basically a rational B-spline curve is the projection of a non-rational 4D B-spline

curve into 3D physical space. In the essence a NURBS curve is the ratio of two B-spline curves, so the rational term in NURBS refers to the ratio of the polynomial bases.

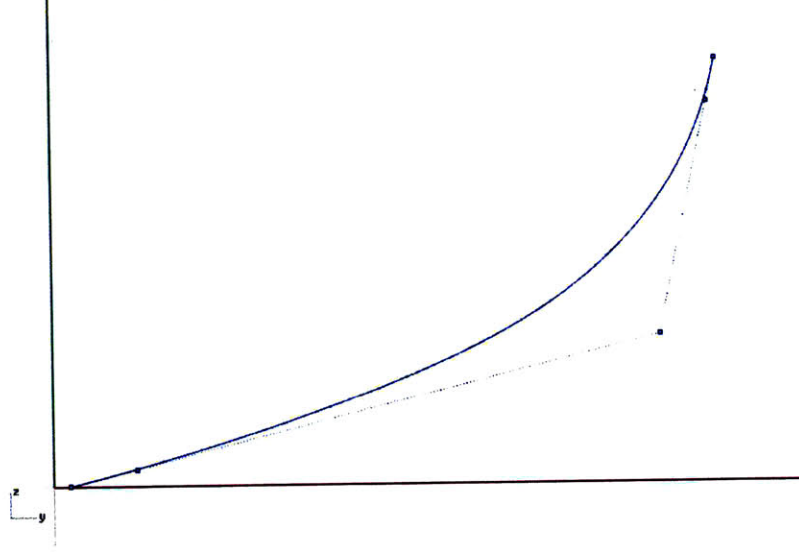


Figure 2-3: A NURBS Curve

$$P(t) = \sum_{i=1}^{n+1} V_i^h N_{i,k}(t) \quad (2.8)$$

In the equation 2.8,  $V_i^h$  represents the non-rational 4D B-spline curve's 4D homogenous defining polygon vertices, whereas  $N_{i,k}(t)$  is the same basis function as in equation 2.6. A rational B-spline curve can be derived from the nonrational curve by projecting back into 3D space and dividing with homogeneous coordinates.

$$P(t) = \frac{\sum_{i=1}^{n+1} V_i h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} = \sum_{i=1}^{n+1} V_i R_{i,k}(t) \quad (2.9)$$

$V_i$ 's in equation 2.9 are the polygon vertices of the rational B-spline curve and equation 2.10 shows the rational B-spline basis functions [13].

$$R_{i,k}(t) = \frac{h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} \quad (2.10)$$

In equation 2.10  $h_i$  terms are the weight parameters, and each weight parameter

(or just weight) corresponds to a distinct control point [3]. The influence of the weights on the curves and surfaces is a unique feature of the NURBS representation. When the weight parameters associated with each of the control vertices are all equal to 1, rational B-spline curve would reduce to its nonrational counterpart. This is the case where we only have a B-spline curve. If a control point's weight parameter is 0, this control point would have no effect on the curve. Oppositely if the weight parameter is infinite for one of the control points, the rational curve would interpolate through this point.

In short, weights can alter the section of the curve and by doing that they provide another opportunity for the local control of curves and surfaces. So that the curve is pulled towards the control point  $V_i$  when the weight is increased, and moves away from the control point  $V_i$  when the weight is decreased. An example is displayed in Figure 2-4 to provide the type of influence the weight has over the shape of the curve. All three curves in Figure 2-4 are 3<sup>rd</sup> degree rational B-spline curves with 5 control vertices. The control vertices are also depicted in the images as the 4 black and 1 white points. The frame on the left has the weight of all its control points as 1, so that it can be considered as the baseline. In the middle frame the weight parameter associated with the white control vertex is reduced to 0.25, as a result you can see that the curve is moved away from the intermediate control point, and almost formed a straight segment. In the final frame the weight parameter of the white control vertex is increased to 2, and this caused the curve to pass closer to this point.

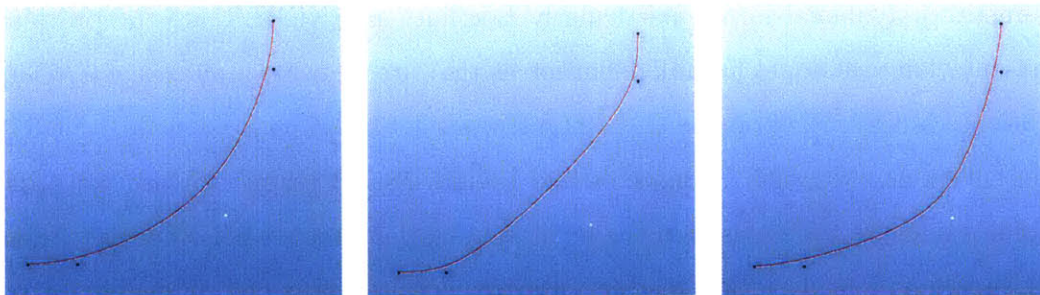


Figure 2-4: Influence of Weights

Rational B-spline representation is a generalization of non-rational B-spline rep-

resentation, and because of that they almost carry out all the characteristics of their non-rational counterparts. Some of the notable characteristics are as follows [3], [13] :

- A rational B-spline curve of order  $k$  is  $C^{k-2}$  continuous everywhere.
- Just like their nonrational counterparts, rational B-spline curves generally follow the shape of their defining polygon, and they lie within convex hull of the  $k$  defining control vertices.
- For any parameter value  $t$ , the sum of all the rational B-spline basis functions is one.

$$\sum_{i=1}^{n+1} R_{i,k}(t) = 1 \quad (2.11)$$

- A change in one of the control point's position or weight parameter will only affect the curve in  $k + 1$  knot spans.

## 2.4 B-Splines and NURBS Representations in Hull Design

Although there are some approaches that use other curve representation techniques, like Yang and Bowers' [9] method which defines the curves as polynomials, parametric polynomial forms, particularly B-splines and NURBS, are the prevalent representation techniques for hull definition as they are for other branches of CAD applications. Most of the popular CASHD softwares are based on B-spline and NURBS representations. Maxsurf, Rhinoceros and FRIENDSHIP Framework are just a few examples that are widely used in the field. Among these tools FRIENDSHIP Framework should be considered in a distinct category with its ability of building parametric models, however in the end all these 3 tools use B-spline and NURBS representation techniques.

B-spline and NURBS representations have many outstanding features that makes



them very popular in naval architecture. The most important ones of these features are listed here [15]:

- With vertex control, it is possible to manipulate these representations.
- Their polynomial degrees can be raised in order to obtain the desired order of continuity.
- It is possible to obtain greater variety of shapes with these representations, than it is for cubic splines.
- They allow for local shape modifications.
- They are invariant under coordinate system transformation.
- Conversion between these representations are feasible.

Looking all the features listed above it can be seen that, with the right set of data, requirements, and constraints, these representations are great for modeling the hull forms of any complexity [15].

## 2.5 iGeo Computational Design Library

iGeo Computational Design Library is an open source 3D modeling software library in Java that includes vector math operations, NURBS curve and surface representations, and 3D model input/output libraries [16]. iGeo comes with an interface for Processing development environment [17], piGeon. Processing's coding environment is a compact one which makes coding a lot easier with its OpenGL integration that provides instant visual feedback. In the early stages of the thesis Processing sketches, which means computer programs for Processing, were extensively used for developmental purposes. Later for modular programming purposes all the iGeo and Processing libraries were imported into Eclipse development environment [18], and the development of the parametric modeling tool was carried out in this programming environment.

Parametric modeling tool generates all the curves and surfaces through iGeo. With its modular structure and good documentation the library is easy to read, and a comprehensive understanding of the library was really helpful during the development of the tool. Another advantage of iGeo is its output format. 3D model outputs can be obtained in widely used .3dm format from iGeo.

One can wonder why Java programming language was preferred despite the fact it is an interpreted language and hence it runs slower than C or C++. There were two reasons for that. First Java is platform independent. In fact throughout the development phase both Windows and Unix platforms were used without any problems. Second reason was the time limitation. This tool is part of a master's degree thesis, and Java programming language is more intuitive to the student than C or C++ languages. So the decision was made in favor of the Java library. However, when the validation cases (FFG-7 and Athena hull forms) were run, it was noted that program only took around 100 seconds to generate the complete 3D model of the hull surfaces, and less than 15 seconds for the generation of the cross sections. This computational times are compatible with the purpose of being integrated into an automatic hull form optimization procedure.

## Chapter 3

# PHull, The Parametric Modeling Tool

In the first chapter a short background was given about the parametric modeling tools, and their capabilities. In the second chapter the mathematical representation techniques that are used by the parametric modeling tool that is developed for this thesis were explained, and the iGeo Computational Design Library was briefly introduced. So now is the time to introduce the parametric modeling tool itself, PHull.

The parametric modeling tool was named as PHull, and it stands for Parametric Hull. One of the first name ideas for the tool was Pusula which literally means "compass" in Turkish. As discussed in the first chapter, parametric modeling tools help designers to make accurate decisions at early phases of the design, so they are like compasses that show the designer the right direction in the early stages of design. PHull's main goal is exactly this. PHull does that with fully parametric rapid hull form generation, which is essential for hydrodynamic optimization. Coupling PHull with a CFD tool and an optimization algorithm, numerous design alternatives can be evaluated at the early design stage, which will significantly increase the chances for the new design to be a successful one.

In this chapter the way that PHull generates the hullform will be discussed first. Then two cases that were used to validate the tool, FFG 7 and ATHENA hulls, will be presented.

### 3.1 How Does PHull Work?

In the development stages of PHull some different methods were used for the representation of the curves that will eventually form the hull. Sometimes it was necessary to devise more than one method to define some of the curves. It was necessary to do this mainly because of the difference between the shape representation and the ab initio design. For instance using interpolation techniques will yield a more accurate result for hull representation, since it's possible to obtain required data from the existing model. However, interpolation techniques don't provide a flexible representation method that is necessary for ab initio design and hydrodynamic optimization. Considering this more than one functions were attached to PHull for the representation of some of the curves, each generates the curve with a different technique. These functions will be described in the relevant parts of this chapter.

Basically PHull generates a hull model in 4 steps:

1. Accepts input in comma separated value (.csv) format
2. Generates control curves from the input data
3. Defines cross section curves from control curves data
4. Creates hull surfaces using the cross section curves (only if it's required)

The ultimate goal of PHull is to generate either the cross section curves or the hull surfaces depending on the CFD tool that is being employed. For a panel method, generation of the cross section curves can be enough, however for a RANS method generation of the high quality and perfectly closed hull surfaces will be required. PHull has the ability to provide the required outputs for both cases. Note that the generation of the cross section curves is a prerequisite for the surface generation in PHull, since PHull creates the hull surfaces with the lofting method over the cross section curves. So, even if it is not the final product, generation of the cross section curves is an essential step when PHull is coupled with a RANS method CFD tool.

In order to generate the cross section curves, longitudinal basic curves called control curves are being used. Control curves provide the essential parametric data

that will be required for the definition of the cross section curves. The types of parameters that are used by PHull are very similar to the ones that were introduced by D. W. Taylor and they are listed below:

- Ending positions of the curves
- Slopes at the end of the curves
- Integral form parameters (area under the curve)

The complete set of control curves that are used in PHull are listed below:

- Underwater Profile Curve
- Design Waterline (DWL) Curve
- Edge of Weather Deck Curve
- Deadrise Angle Curve
- Flare Angle at DWL Curve
- Flare Angle at Weather Deck Curve
- Sectional Area Curve (SAC)
- Width of the Keel Curve

Underwater profile curve, DWL curve, edge of the weather deck curve, and width of the keel curve provide position data for the cross section curves. Similarly deadrise angle curve, flare angle at DWL curve, and flare angle at weather deck curve provide slope data. Finally sectional area curve, self evident from its name, provides area data. Detailed information about these control curves and PHull in general will be provided in the following sections of this chapter.

## 3.2 Inputs for PHull

In the end of Chapter 2 the iGeo Computational Design Library, and its interface for Processing development environment, piGeon, were introduced. PHull implements piGeon in order to benefit from the perks of Processing.

Processing provides the chance to use the data from comma separated value (.csv) files, and this is how PHull gets its inputs. The .csv extension may not be unfamiliar to some, but a spreadsheet can be saved as a .csv file from Microsoft Excel. An example input table is displayed in Table 3.1.

The complete manual for the input file is presented in Appendix A, but to give a quick overview, data related to the principal dimensions, hull form coefficients, and the important points, e.g. stem rise point and keel rise point, are defined in the blue section of the spreadsheet. In the yellow section position, slope, and area data about the cross sections are defined that are used for the generation of the control curves by interpolation. Whereas the orange section contains the data for the generation of the control curves by approximation. Finally in the green section some parameters are defined for the systematic generation by PHull, which will be discussed in Chapter 4. It is always required by PHull to fill in the blue section, however for the orange or the yellow sections filling either of them is enough. And the choice between these two depends on the intention of the user.

Data from the blue section is called by PHull for creating:

- Underwater Profile Curve
- Design Waterline (DWL) Curve
- Edge of the Weather Deck Curve

and the data from the yellow or the orange section is used to generate

- Deadrise Angle Curve
- Flare Angle at DWL Curve
- Flare Angle at the Weather Deck Curve

LBP	124.04	DeadriseAngle1	47.917	DeadriseAngle11	10.596	DeadriseAngle21	1.312
L/B	9.107195	DeadriseAngle2	38.21	DeadriseAngle12	11.788		
T	4.38	DeadriseAngle3	28.512	DeadriseAngle13	13.898		
Top Stem Angle	46.12	DeadriseAngle4	21.332	DeadriseAngle14	17.942		
Bottom Stem Angle	1.5	DeadriseAngle5	15.298	DeadriseAngle15	18.993		
Stem Corner Point Weight	0.85	DeadriseAngle6	13.602	DeadriseAngle16	17.481		
Stem Radius/T	1.783105	DeadriseAngle7	12.998	DeadriseAngle17	15.093		
Keel Rise Point	0.645679	DeadriseAngle8	12.381	DeadriseAngle18	11.166		
Transom Depth	0.23	DeadriseAngle9	11.447	DeadriseAngle19	7.297		
Keel Angle @ Transom	6.21	DeadriseAngle10	10.7	DeadriseAngle20	3.642		
Transom Angle	50.94	FlareAngle1	13.175	FlareAngle11	9.366	FlareAngle21	71.945
Beam@Transom/Maxbeam	0.502203	FlareAngle2	15.65	FlareAngle12	9.592		
Max Beam X Location	0.554418	FlareAngle3	15.695	FlareAngle13	11.475		
DWL Fwd Angle	11.02	FlareAngle4	16.38	FlareAngle14	16.908		
DWL Aft Angle	11.78	FlareAngle5	17.766	FlareAngle15	21.949		
Cwp Aft	0.814	FlareAngle6	21.003	FlareAngle16	23.63		
Cwp Fwd	0.613	FlareAngle7	20.884	FlareAngle17	24.19		
DWL Fwd WLPtShftX	-3.25	FlareAngle8	17.638	FlareAngle18	28.575		
DWL Fwd WLPtShftY	-0.19	FlareAngle9	14.329	FlareAngle19	35.622		
DWL Aft WLPtShftX	3	FlareAngle10	11.7	FlareAngle20	45.182		
DWL Aft WLPtShftY	-0.1	SectionalArea1	3.298	SectionalArea11	22.192	SectionalArea21	0.6053
xProfileEndToTransom	0.15	SectionalArea2	5.172	SectionalArea12	22.01		
fbAft	5.147	SectionalArea3	7.01	SectionalArea13	21.224		
fbFwd	8.37	SectionalArea4	8.898	SectionalArea14	19.822		
fbLow	4.505	SectionalArea5	11.752	SectionalArea15	17.484		
fbFwdXPos	-8.537	SectionalArea6	14.212	SectionalArea16	14.532		
fbLowXPos	88.784	SectionalArea7	16.466	SectionalArea17	11.413		
fbAftAng	1.623	SectionalArea8	18.51	SectionalArea18	8.312		
fbFwdAng	5.7	SectionalArea9	20.253	SectionalArea19	5.437		
weatherDeckFwdAng	16.231	SectionalArea10	21.507	SectionalArea20	2.82		
weatherDeckAftAng	6.478	KeelWidth1	0	KeelWidth11	0.176	KeelWidth21	0.176
fwdDisplacement	1615	KeelWidth2	0.148	KeelWidth12	0.176		
areaAtTransom	1.16	KeelWidth3	0.164	KeelWidth13	0.176		
fwdLcb	40.29	KeelWidth4	0.176	KeelWidth14	0.176		
sacEntranceAngFwd	35.964	KeelWidth5	0.176	KeelWidth15	0.176		
sacEntranceAngStem	34.767	KeelWidth6	0.176	KeelWidth16	0.176		
areaAtMaxBeam	44.389	KeelWidth7	0.176	KeelWidth17	0.176		
tolLCB	0.1	KeelWidth8	0.176	KeelWidth18	0.176		
tolDisp	1	KeelWidth9	0.176	KeelWidth19	0.176		
maxSecAreaXPos	0.5	KeelWidth10	0.176	KeelWidth20	0.176		
aftDisplacement	1060	WDFlareAng1	22.309	WDFlareAng11	3.025	WDFlareAng21	13.025
aftLcb	83.522	WDFlareAng2	24.913	WDFlareAng12	1.937		
Displacement (for design)	3295.73	WDFlareAng3	25.688	WDFlareAng13	1.018		
LCB (for design)	62.237	WDFlareAng4	25.948	WDFlareAng14	1.278		
1st CrossSec drAng	47.917	WDFlareAng5	26.007	WDFlareAng15	2.734		
Last CrossSec drAng	1.312	WDFlareAng6	24.823	WDFlareAng16	5.597		
drAngCrv fwdEntAng	13.138	WDFlareAng7	21.295	WDFlareAng17	9.794		
drAngCrv aftEntAng	67.111	WDFlareAng8	16.099	WDFlareAng18	14.254		
drAng at 0.45	10.47	WDFlareAng9	10.488	WDFlareAng19	17.326		
drAng at 0.7	19.083	WDFlareAng10	5.703	WDFlareAng20	15.23		
1st CrossSec flAng	13.175	# of Outputs	1				
Last CrossSec flAng	71.945	Increase Fwd Fullness	0				
flAngCrv fwdEntAng	43.83	Decrease Fwd Fullness	0				
flAngCrv aftEntAng	6.835	Increase Aft Fullness	0				
flAng at 0.275	21.915	Decrease Aft Fullness	1				
flAng at 0.525	9.117	Increase Fullness	0				
1st CrossSec wdAng	26.101	Decrease Fullness	0				
Last CrossSec wdAng	17.324						
wdAngCrv fwdEntAng	90						
wdAngCrv aftEntAng	90						
wdAng at 0.625	1						
wdAng at 0.7	0						
1st CrossSec kw	0.176						
Last CrossSec kw	5.91						
kwCrv fwdEntAng	90						
kwCrv aftEntAng	90						
kw straightLinePt xPos	0.15						
xMultiplierFwd	1						
xMultiplierAft	1						
IntermediateDrAngPos1	0.45						
IntermediateDrAngPos2	0.7						
IntermediateFlAngPos1	0.275						
IntermediateFlAngPos2	0.525						
IntermediateWdAngPos	0.625						
weightPYPos@0.4LBP	5.333						
weightPYPos@0.4SLBP	5.865						
weightPYPosDenominatorForStemRise	24						
weightPZPos@0.4LBP	1.08						
weightPZPos@0.4SLBP	1.108						
weightPZPosDenominatorForStemRise	8						
transomHalfbeamMultiplier	1.57						
maxHalfbeamMultiplier	1.04						

Table 3.1: .csv Input File for PHull

- Sectional Area Curve (SAC)
- Width of the Keel Curve

### 3.3 Control Curves

Following from section 3.1, the second step is to generate the control curves. The control curves are the main building block of PHull and used to generate the underwater and the above water parts of the cross section curves. They are the longitudinal curves that run from the stem rise point to the aft perpendicular with the exception of the sectional area curve, which runs from the forward perpendicular to aft perpendicular. They contain all the position, slope, and area data that is necessary for the generation of the transversal sections. The eight control curves that PHull uses are introduced next.

#### 3.3.1 Underwater Profile Curve

Underwater profile curve is the first control curve that PHull generates. It's a 2D curve in X-Z plane and provides the position data for the bottom end points of the transversal sections. In Figure 3-1 it can be seen as the red curve.

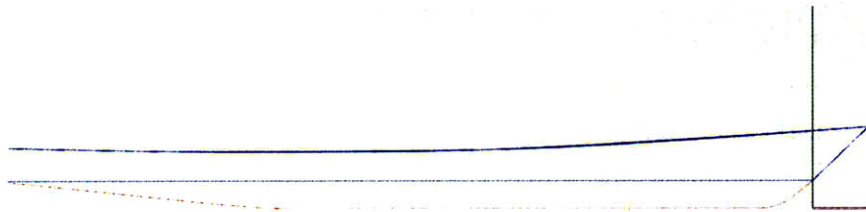


Figure 3-1: An Underwater Profile Curve Example

PHull generates the underwater profile curve in two pieces, namely the underwater stem curve and the keel curve. Underwater stem curve is a NURBS curve, that is



created with 5 control points, two of which are the end points, other two are the tangency constraint points, and the final one is the intermediate, or the weight, point.

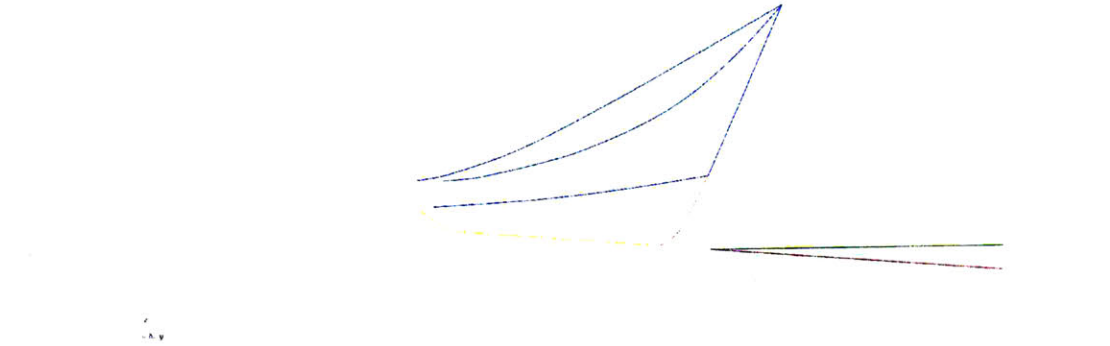


Figure 3-2: Underwater Stem Curve

A stem curve, the part in red, generated by PHull can be seen in Figure 3-2. According to the coordinate system that PHull uses intersection of the design waterline and the stem curve is at  $(0, 0, DRAFT)$ , which is also considered as the longitudinal position of the FP (forward perpendicular).

At this point the the coordinate system and the sign convention that PHull uses need to be introduced. Longitudinally any point aft of FP, or the intersection of the design waterline and the stem curve, has a negative x coordinate, transversally port side of the centerline is positive, and vertically any point above baseline is positive, where the vertical position of baseline is 0. These conventions will apply throughout this thesis.

Figure 3-3 shows an underwater stem curve with its control vertices, the first end point, top end point, is located at  $(0, 0, DRAFT)$ , and the location of the first tangency constraint point is calculated by using `TOP_STEM_ANGLE` parameter. The second end point is located at  $(-D\_STEM\_RISE, 0, 0)$ , which is exactly at the stem rise point, similarly the location of the second tangency constraint point is calculated with the `BOTTOM_STEM_ANGLE` parameter. Each end point and its respective tangency constraint points sit on a line, and the intersection of these two lines gives the location of the intermediate point, which will also be referred as the

weight point throughout this thesis. The weight point here controls the fullness of the stem curve, by means of the `STEM_CORNER_WEIGHT` parameter.

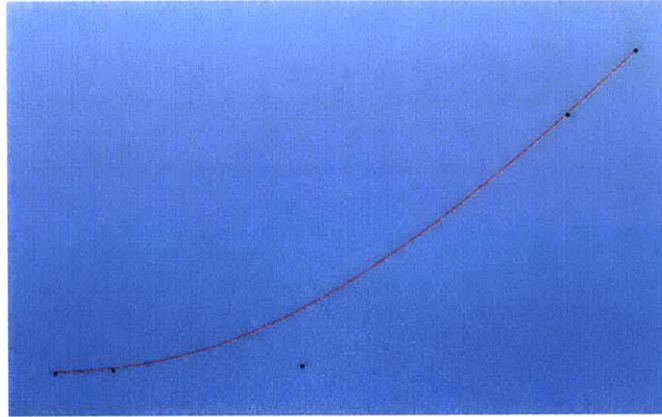


Figure 3-3: Underwater Stem Curve with Control Vertices

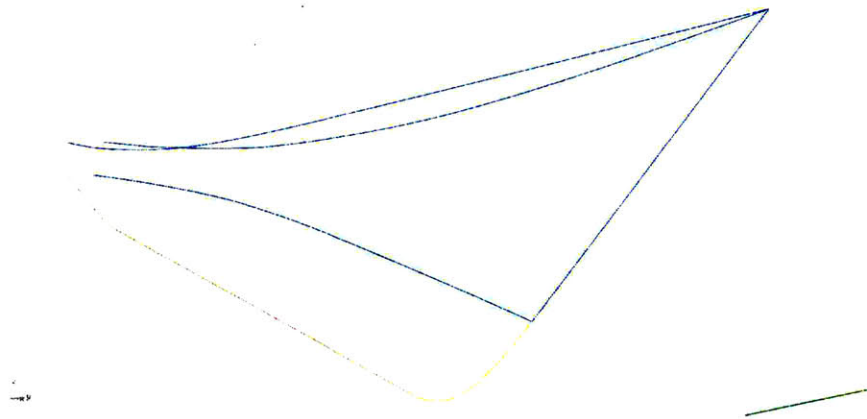


Figure 3-4: Keel Curve

A keel curve (red curve) on a 3D hull model is displayed in Figure 3-4, which has a straight part in the forward and a curved part in the aft. Furthermore Figure 3-5 depicts this keel curve with its 8 control vertices. The first vertex for the keel curve is the same with the last vertex of the underwater stem curve. Second vertex is located halfway between the first vertex and  $(KEEL\_RISE\_POINT, 0, 0)$ . The purpose of this second vertex is to make sure that the first part of the keel curve would run straight. Fourth vertex is exactly at  $(KEEL\_RISE\_POINT, 0, 0)$ , this is the place

that the transition from the straight part to the curved part takes place. Third and fifth vertices are located very close to fourth point and they assure that the keel curve would rise from the baseline exactly at the `KEEL_RISE_POINT`, these points are automatically assigned by PHull. These three points are so close to each other that it's not possible to distinguish them from the figure.

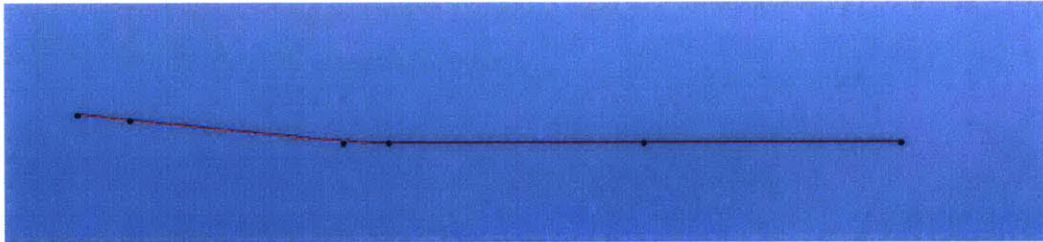


Figure 3-5: Keel Curve with Control Vertices

Sixth and seventh points are the tangency constraint points for the curved part of the keel curve. Finally the last point is the end point of the keel curve, which is located at  $(-LBP, 0, DRAFT - TRANSOM\_DEPTH)$ .

This is the right place to talk about an assumption made by PHull about the transom part of the ship. PHull assumes that the transom part is straight and vertical, and models the hull according to this assumption. So it's not possible to model a hull with an inclined transom with PHull for now.

A complete list of parameters that are used for the definition of the underwater profile curve is shown in Table 3.2.

### 3.3.2 Design Waterline Curve (DWL)

Second control curve that is generated by PHull is the DWL curve. This curve provides both the top end point data for underwater part of the cross section curve, and the bottom end point data for the above water part of the cross section curve. A DWL curve generated by PHull is presented in Figure 3-6 and Figure 3-7 from 2 different views.

PHull creates the design water line curve in two steps. In the first step aft and forward parts of the DWL are generated separately with 5 control vertices for each

<b>Input Parameters for Profile Curve</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	T	m	draft
<b>3</b>	Top Stem Angle	deg	determines tangency constraint for the intersection of DWL and Profile Curve
<b>4</b>	Bottom Stem Angle	deg	determines tangency constraint for the stem rise point
<b>5</b>	StemRadius/T	nd	determines the position of stem rise point
<b>6</b>	Stem Corner Point Weight	nd	control the fullness of the underwater stem curve
<b>7</b>	Transom Depth	m	determines vertical position of the profile curve's end point at stern
<b>8</b>	Keel Rise Point	nd	ratio of the longitudinal position where keel rises from baseline to lbp
<b>9</b>	Keel Angle @ Transom	deg	determine the tangency constraint at the end point

Table 3.2: Parameters for the Profile Curve

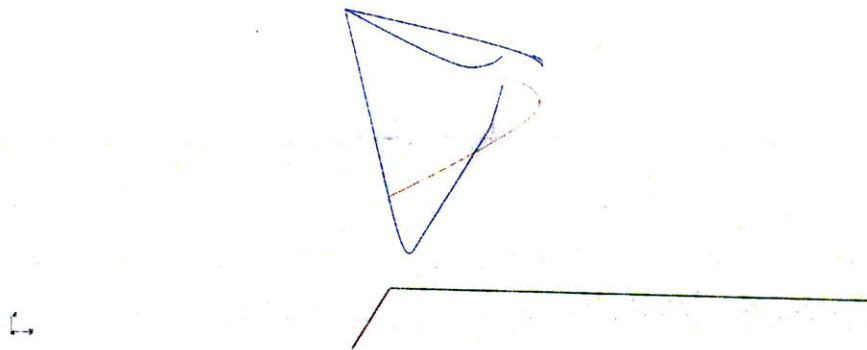


Figure 3-6: DWL Curve from Forward

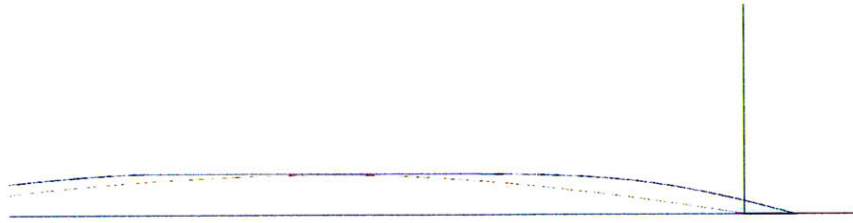


Figure 3-7: DWL Curve from Top

curve, as the beam with the maximum width is the border between them. In the second step all 9 control vertices (1 vertex is common for both curves) are used to define one DWL curve. The reason for this might not be clear right now, but it will be clear towards the end of this part.

The way the two pieces of the DWL curve are built is similar to the way the underwater stem curve was obtained. First of all they are both NURBS curves, secondly control vertices of each curve consist of two end points, two tangency constraint points, and the weight point, just as the stem curve. Figure 3-8 and Figure 3-9 shows the forward and aft parts of the DWL curve with their control vertices. It's not easy to see the tangency constraint vertices from the figures since they are very close to the end points.

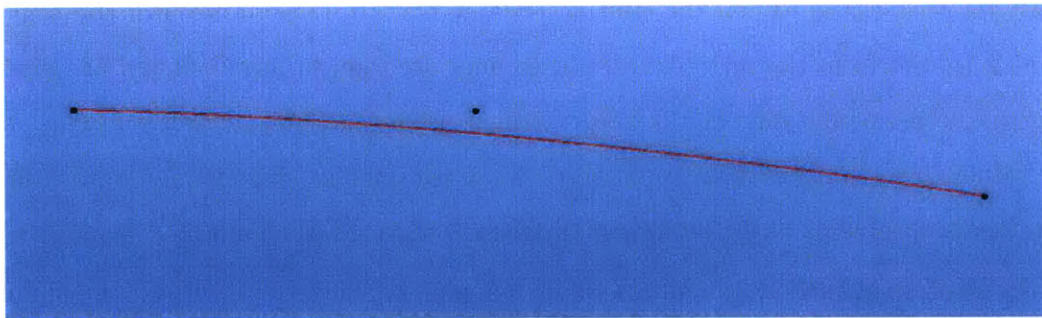


Figure 3-8: Forward Part of the DWL Curve

For the forward part first end point is the same as the top end point of the underwater stem curve,  $(0, 0, DRAFT)$ . The tangency constraint point for the first end point is determined by the `DWL_FWD_ANGLE` parameter. The coordinates of the second end point, which is at the same longitudinal position with the maximum beam, is  $(-X\_MAX\_BEAM * LBP, HALFBEAM, DRAFT)$ , and 0 degree tangency constraint is used for this end point in order to assure the continuity between the two parts of the DWL curve. Finally the position of the weight point is found as the intersection of two end point - tangency constraint point lines, just as it was done in the underwater stem curve case, but the process of curve generation is not over for DWL curve, since there is one more constraint that has to be satisfied, the forward waterplane coefficient,  $C_{wp_{fwd}}$ , constraint.

This constraint is identified with the `CWP_FWD` parameter, and should be given as an input by the user. Then PHull calculates the forward waterplane area as the given area using this coefficient. After the position of the weight point is found as the intersection of two lines, area of the generated waterplane is calculated with the trapezoidal method. Then PHull checks if this calculated area value is within a certain tolerance of the given area value. If not, and in most cases it's not for the first try, the weight parameter of the weight point is changed iteratively until the calculated area falls somewhere within the tolerance range of the given area.

There are two more parameters for the forward part of the DWL curve, `DWL_FWD_WT_PT_SHIFT_X` and `DWL_FWD_WT_PT_SHIFT_Y`. These parameters can be used to shift the location of the weight point to achieve better accuracy for shape representation cases. When the shifting parameters are used, position of the weight point is moved from the intersection of two lines according to the value of the input, and the area calculations would be performed after this shifting process. However the values should be assigned carefully for these parameters, since large modifications could cause problems with the form and fairness of the DWL curve.

As mentioned earlier, generation of the aft part of the DWL curve is very similar to the forward part. First of all two curves share the end point at the maximum beam location,  $(-X\_MAX\_BEAM * LBP, HALFBEAM, DRAFT)$ . Again, 0 degree

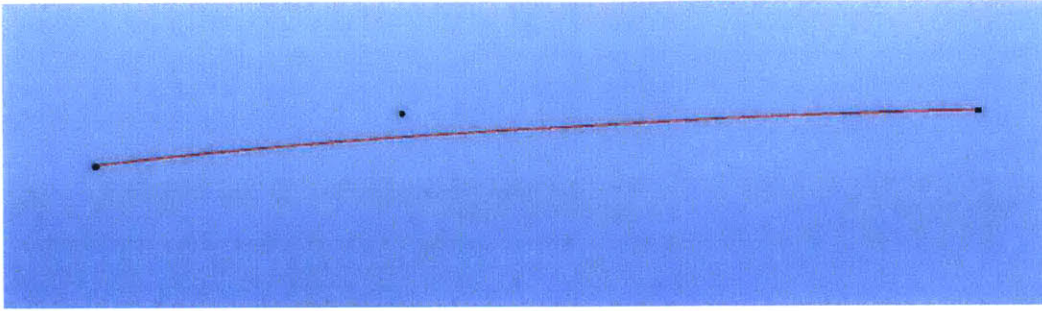


Figure 3-9: Aft Part of the DWL Curve

tangency constraint is used for this point to assure continuity between the two pieces of the DWL curve. Second end point's location is  $(-LBP, HALFBEAM\_TRANSOM, DRAFT)$ , which is at the same position with AP (aft perpendicular), and its tangency constraint point is determined with the `DWL_AFT_ANGLE` parameter. Area constraint is applied with the `CWP_AFT` parameter, and the same iterative approach is used to obtain the curve with the right area. Finally there are two parameters, just as in the forward part, that enables the user to control the position of the weight point, `DWL_AFT_WT_PT_SHIFT_X` and `DWL_AFT_WT_PT_SHIFT_Y`.

The reason for building the DWL curve in two separate pieces should be explained now. During the development of PHull the most accurate DWL curve outputs were obtained when two waterplane coefficients, forward and aft, were used. So this is the approach that PHull uses now. As the final step the DWL curve is generated as a single curve by using the already known 9 control vertices of the forward and aft parts.

A complete list of parameters that are used for the definition of the DWL curve is shown in Table 3.3.

### 3.3.3 Edge of the Weather Deck Curve

In the beginning of this part an important fact about PHull has to be emphasized, as mentioned in the introduction part, the ultimate goal of PHull is to produce a parametrically defined 3D CAD model in order to be used in hydrodynamic analysis

<b>Input Parameters for DWL Curve</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	L/B	nd	length to max beam ratio
<b>3</b>	Beam@Transom/ MaxBeam	nd	ratio of the beam at transom and the maximum beam
<b>4</b>	Max Beam X Location	nd	ratio of the longitudinal location of max beam and lbp
<b>5</b>	Dwl Fwd Angle	deg	determines tangency constraint for the forwardmost point of DWL curve
<b>6</b>	Dwl Aft Angle	deg	determines tangency constraint for the aftmost point of DWL curve
<b>7</b>	Cwp Fwd	nd	waterplane area coefficient for aft part
<b>8</b>	Cwp Aft	nd	waterplane area coefficient for fwd part
<b>9</b>	DWLFwdWtPtShiftX	m	changes the longitudinal position of the weight point on forward part of DWL
<b>10</b>	DWLFwdWtPtShiftY	m	changes the transversal position of the weight point on forward part of DWL
<b>11</b>	DWLAftWtPtShiftX	m	changes the longitudinal position of the weight point on aft part of DWL
<b>12</b>	DWLAftWtPtShiftY	m	changes the transversal position of the weight point on aft part of DWL

Table 3.3: Parameters for the DWL Curve



and optimization. This means that the generation of the above water part is not as important as the generation of the underwater part, since it doesn't have a significant impact on the hydrodynamic performance of the hull form. So, PHull only represent the above water part of the hull roughly for the model to be used in RANS (Reynolds Averaged Navier Stokes) analysis.

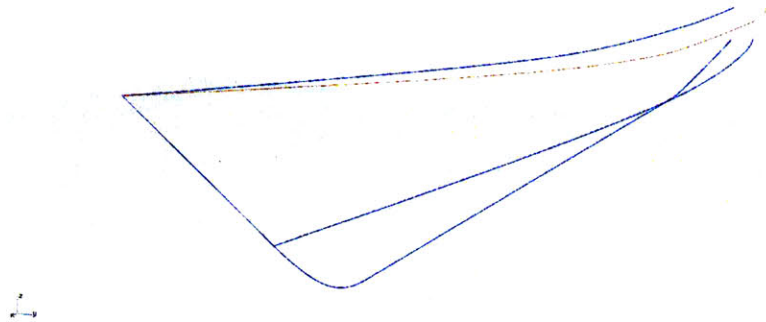


Figure 3-10: Edge of the Weather Deck Curve for FFG 7 (Generated by PHull)

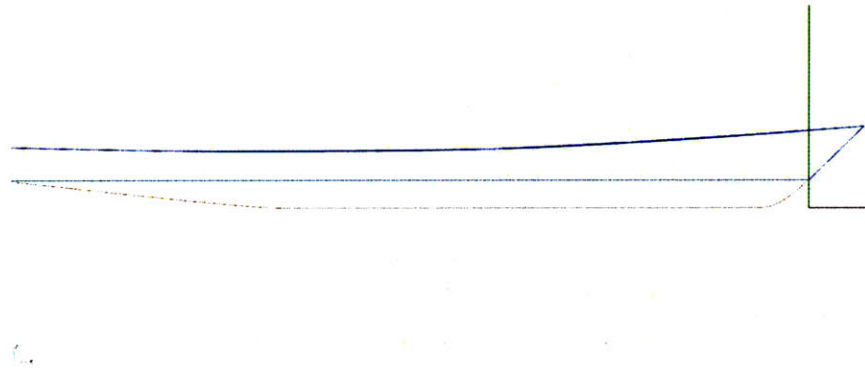


Figure 3-11: Projection of Edge of the Weather Deck Curve (Thick Blue Line) on XZ Plane

Edge of the weather deck curve, the red curve in Figure 3-10, provides data for the top end point positions of the above waterline parts of the cross section curves and is the only 3D control curve that PHull generates. This fact makes the generation of this curve a little bit more difficult. The generation of the edge of the weather deck curve is performed in two steps by PHull. Firstly a 2D projection of the curve is

generated in XZ plane, the thick blue curve in Figure 3-11, with the parameters for the end positions, tangency constraints for these end positions in XZ plane, and an additional position data for the point with the smallest height value on the curve. The purpose of this 2D curve is to provide the z coordinate data for the control vertices in the second step, at which the edge of the weather deck curve itself is being built.

For the second step 11 control points that are not sitting on the same plane are used to represent the edge of the weather deck curve accurately. The positions of these control points on XY plane are determined using the end point positions data, tangency constraint data for these end points in XY plane, and the position data for the section with the maximum beam. Additionally, similar to the approach followed by Nestoras [19], the width of the weather deck curve is assumed to be constant from 40% to 80% of LBP. To provide this straight segment of the weather deck some additional control points are defined automatically by PHull, and that's the reason 11 control points in total are needed for the generation of the edge of the weather deck curve.

A complete list of parameters that are used for the definition of this curve is presented in Table 3.4.

If we classify the control curves into 2 groups, the ones that we investigated so far, the profile curve, design water line curve, and the edge of the weather deck curve, would be in one group. Because all these curves are directly part of the hull, and PHull uses only one way to generate these curves, the approximation technique. But the control curves starting with the deadrise angle curve that we will start investigating now form the second group. These curves actually are not part of the hull, and they only include some necessary data for the generation of the cross sections. Furthermore PHull has two different methods for the representation of these curves, one with the interpolation technique and the other one with the approximation technique. As mentioned earlier interpolation technique is mainly for the shape representation purposes, whereas the approximation technique is for ab initio design. The control curves of the second group will be discussed after this point.

<b>Input Parameters for Edge of Weather Deck Curve</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	L/B	nd	length to max beam ratio
<b>3</b>	Beam@Transom/ MaxBeam	nd	ratio of the beam at transom and the maximum beam
<b>4</b>	Max Beam X Location	nd	ratio of the longitudinal location of max beam and lbp
<b>5</b>	Weather Deck Fwd Angle	deg	determines tangency constraint for the forwardmost point of the edge of the weather deck curve (in XY plane)
<b>6</b>	Weather Deck Aft Angle	deg	determines tangency constraint for the aftmost point of the edge of the weather deck curve (in XY plane)
<b>7</b>	fbFwdAng	deg	determines tangency constraint for the forwardmost point of the edge of the weather deck curve (in XZ plane)
<b>8</b>	fbAftAng	deg	determines tangency constraint for the aftmost point of the edge of the weather deck curve (in XZ plane)
<b>9</b>	fbFwdXPos	m	longitudinal position data for the forwardmost point of the ship
<b>10</b>	fbLowXPos	m	longitudinal position data for the point with the smallest height in the edge of the weather deck curve
<b>11</b>	fbFwd	m	Freeboard value for the forwardmost point of the edge of the weather deck curve
<b>12</b>	fbAft	m	Freeboard value for the aftmost point of the edge of the weather deck curve
<b>13</b>	fbLow	m	Freeboard value for the point with the smallest height in the edge of the weather deck curve

Table 3.4: Parameters for the Edge of the Weather Deck Curve

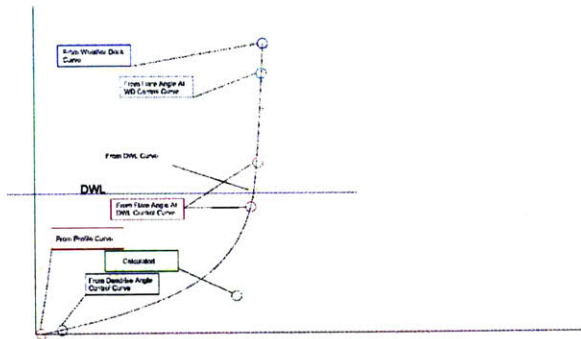


Figure 3-12: Which Part Does Each Control Curve Control on a Cross Section?

### 3.3.4 Deadrise Angle Curve

The first curve that is being generated by PHull from the second group is the deadrise angle curve. This curve carries the data about the tangency constraints for the bottom end points of the underwater cross section curves. This means that it controls the position of the control point within the black circle in Figure 3-12, which literally determines the deadrise angle for the cross section. PHull generates this curve from the stem rise point to the aft perpendicular.

<b>DeadriseAngle1</b>	47.917	<b>DeadriseAngle11</b>	10.596	<b>DeadriseAngle21</b>	1.312
<b>DeadriseAngle2</b>	38.21	<b>DeadriseAngle12</b>	11.788		
<b>DeadriseAngle3</b>	28.512	<b>DeadriseAngle13</b>	13.898		
<b>DeadriseAngle4</b>	21.332	<b>DeadriseAngle14</b>	17.942		
<b>DeadriseAngle5</b>	15.298	<b>DeadriseAngle15</b>	18.993		
<b>DeadriseAngle6</b>	13.602	<b>DeadriseAngle16</b>	17.481		
<b>DeadriseAngle7</b>	12.998	<b>DeadriseAngle17</b>	15.093		
<b>DeadriseAngle8</b>	12.381	<b>DeadriseAngle18</b>	11.166		
<b>DeadriseAngle9</b>	11.447	<b>DeadriseAngle19</b>	7.297		
<b>DeadriseAngle10</b>	10.7	<b>DeadriseAngle20</b>	3.642		

Table 3.5: Input for Deadrise Angle Curve Interpolation

The fact that PHull can generate this curve with two different ways was discussed previously. A deadrise angle curve generated with the interpolation technique is presented in Figure 3-13, which is the recommended approach for shape representation. For this representation of the curve deadrise angle data for 21 cross sections have to be provided in the input file like in Table 3.5. When the tool starts working with this input, it'll call the SPLINEFIT toolbox of MATLAB that provides the ability

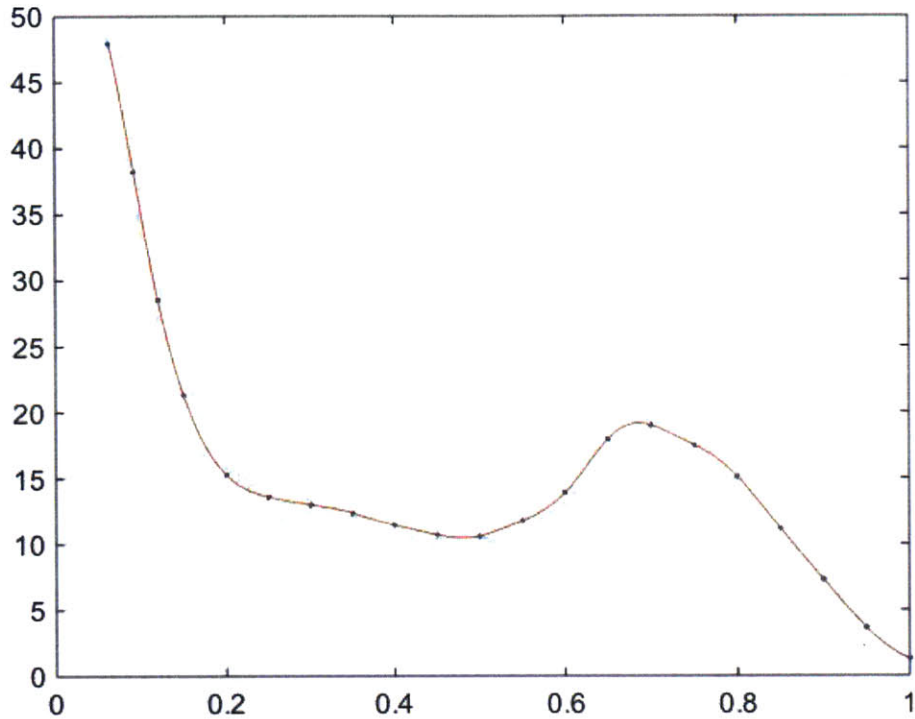


Figure 3-13: Deadrise Control Curve with Interpolation Method

to interpolate through these 21 points with a reverse B-spline algorithm. Using the reverse B-spline algorithm brings more flexibility than using a cubic spline interpolation. It is also possible to see these 21 points from Figure 3-13 as the black dots on the red curve.

The SPLINEFIT toolbox is based on B-splines and it generates the curve with the interpolation of the data points. It can be considered as a reverse algorithm for B-spline curves, since normally they are approximating curves. This MATLAB toolbox can be called from Java programming environment with the help of Builder JA toolbox. This toolbox gives user the opportunity to create Java classes from MATLAB code.

The second method PHull uses for the generation of the deadrise angle curve is the approximation method. In this method the curves are being created with B-spline

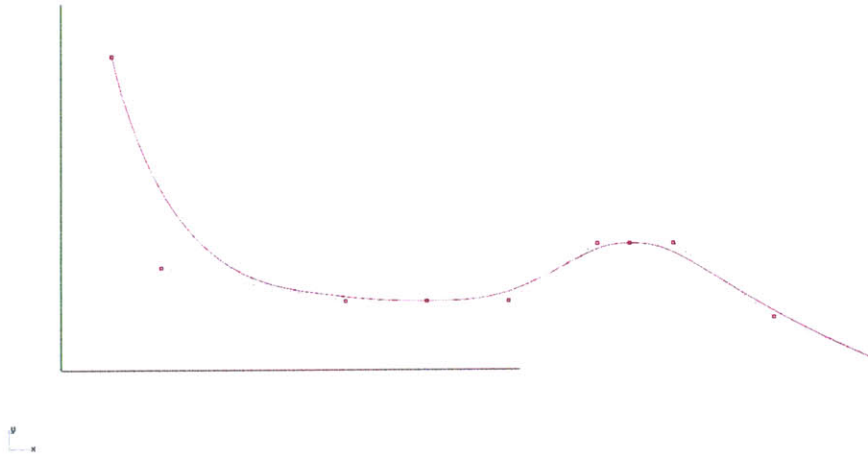


Figure 3-14: Deadrise Control Curve with Approximation Method

<b>Input Parameters for Deadrise Control Curve Approximation</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	T	m	draft
<b>3</b>	StemRadius/T	nd	determines the position of stem rise point
<b>4</b>	1st CrossSecDRAng	deg	deadrise angle for the cross section at the stem rise point
<b>5</b>	Last CrossSecDRAng	deg	deadrise angle for the cross section at AP
<b>6</b>	2nd CrossSecDRAng X Position	m	longitudinal position data for the second cross section
<b>7</b>	2nd CrossSecDRAng	deg	deadrise angle for the second cross section
<b>8</b>	3rd CrossSecDRAng X Position	m	longitudinal position data for the third cross section
<b>9</b>	3rd CrossSecDRAng	deg	deadrise angle for the third cross section
<b>10</b>	drAngCrv fwdEntAng	deg	determines tangency constraint for the fwd part of the deadrise angle curve
<b>11</b>	drAngCrv aftEntAng	deg	determines tangency constraint for the aft part of the deadrise angle curve

Table 3.6: Input for Deadrise Angle Curve with Approximation Method

representation, using the utilities of iGeo library. An example is presented in Figure 3-14. For this representation PHull uses 10 control points that can be controlled with the set of parameters presented in Table 3.6. Looking at the parameters listed in Table 3.6, one can see that PHull requires the deadrise angle values for 4 different longitudinal positions (2 of which are the end points) and the entrance angles for the

curve.

### 3.3.5 Flare Angle at DWL Curve

Flare angle curve carries the data about the tangency constraints for the top end points of the underwater cross section curves and the bottom end points of the above water cross section curves. This means that it controls the position of the control points within the magenta circles in Figure 3-12, which literally determines the flare angle for the cross section. Similar to the deadrise angle curve, PHull generates this curve from stem rise point to aft perpendicular.

The generation of the flare angle curve for both the interpolation and the approximation techniques are pretty similar to the generation of the deadrise angle curve with these two methods. Actually the generation of all the curves in the second group with the interpolation method is the same, 21 data points are required as input and these points are interpolated using a reverse B-spline algorithm. Figure 3-15 shows the flare angle curve with interpolation.

A flare angle curve created with the approximation method is presented in Figure 3-16, and the parameters that are used to define the flare angle curve with the approximation method are listed in Table 3.7.

### 3.3.6 Sectional Area Curve (SAC)

As it is self evident from its name, SAC contains the sectional area data for all the cross sections and it is directly related to the intermediate control point, which is shown within the green circle in Figure 3-12. According to the sectional area value derived from SAC, this control point changes the fullness of the cross section curve iteratively, until the area required by the SAC is achieved within a certain tolerance. So as a summary, SAC controls the fullness of the cross sections.

SAC is the only curve that PHull can create with 3 different methods. First, just as the other control curves in the second group SAC can be represented with the interpolation method using the 21 data points, and it extends from the stem rise

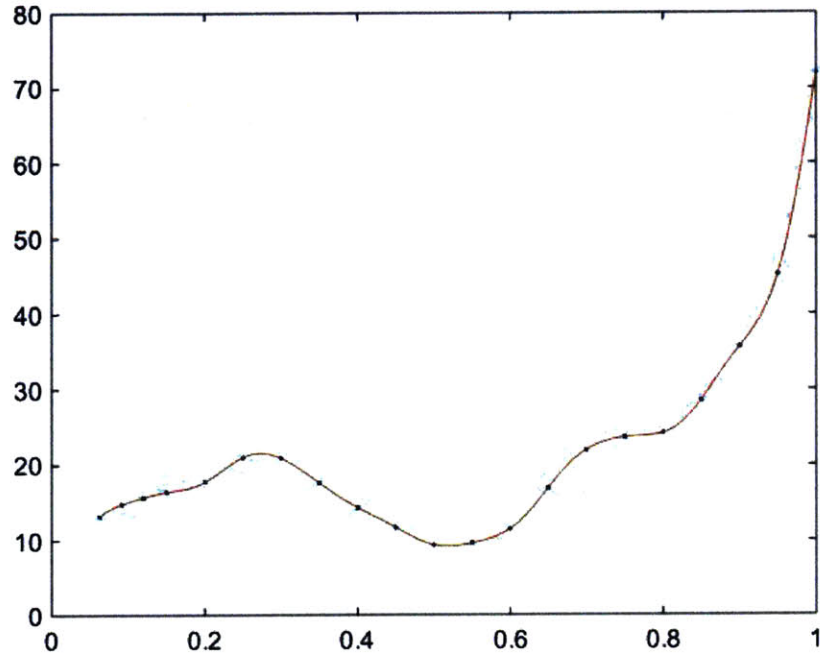


Figure 3-15: Flare Angle Control Curve with Interpolation Method

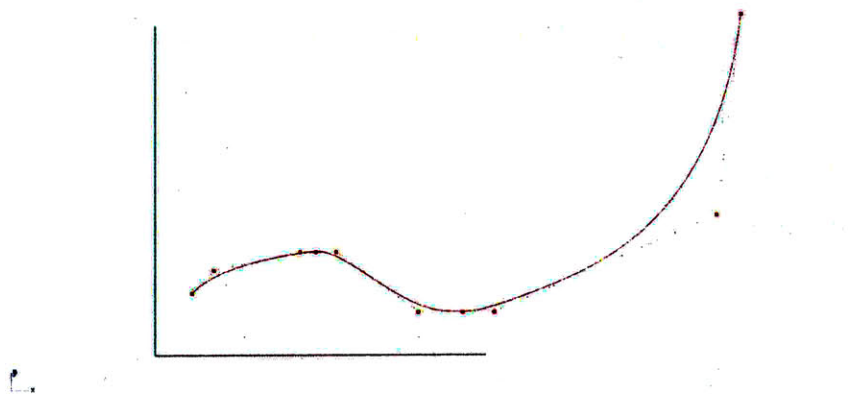


Figure 3-16: Flare Angle Control Curve with Approximation Method



<b>Input Parameters for Flare Control Curve Approximation</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	T	m	draft
<b>3</b>	StemRadius/T	nd	determines the position of stem rise point
<b>4</b>	1st CrossSecFLAng	deg	flare angle for the cross section at the stem rise point
<b>5</b>	Last CrossSecFLAng	deg	flare angle for the cross section at AP
<b>6</b>	2nd CrossSecFLAng X Position	m	longitudinal position data for the second cross section
<b>7</b>	2nd CrossSecFLAng	deg	flare angle for the second cross section
<b>8</b>	3rd CrossSecFLAng X Position	m	longitudinal position data for the third cross section
<b>9</b>	3rd CrossSecFLAng	deg	flare angle for the third cross section
<b>10</b>	flAngCrv fwdEntAng	deg	determines tangency constraint for the fwd part of the flare angle curve
<b>11</b>	flAngCrv aftEntAng	deg	determines tangency constraint for the aft part of the flare angle curve

Table 3.7: Input for Flare Angle Curve with Approximation Method

point until the AP. A SAC curve interpolated through 21 data points is presented in Figure 3-17.

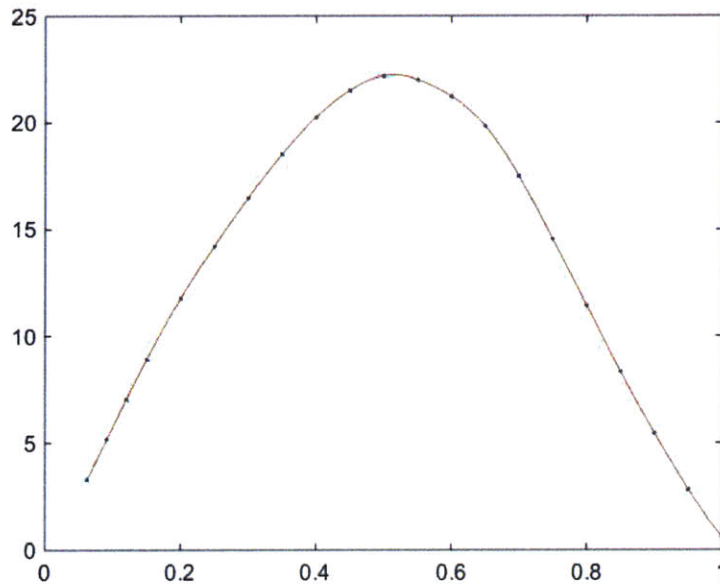


Figure 3-17: Sectional Area Curve with Interpolation Method

In the second method PHull creates the SAC with 2 pieces, the forward and the aft parts. For this representation area under the curve, and the longitudinal centroid location integral parameters are also used as the inputs for both the forward and the aft parts separately. Area under SAC actually gives the displacement volume of the ship, and the longitudinal centroid location of this area gives the longitudinal center of buoyancy (LCB). So PHull uses the displacement and the LCB as its parameters for this representation, but since this method creates SAC in two pieces, it requires two displacement and centroid parameters, one for each part. Besides entrance angle values are also required in the same fashion with the deadrise and flare angle control curves. A SAC which was generated with this method is shown in Figure 3-18 and the parameters required for this method is presented in Table 3.8.

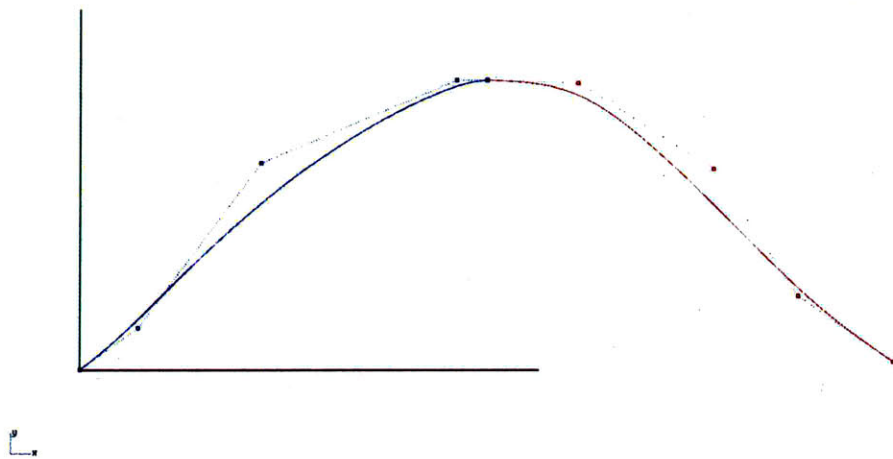


Figure 3-18: SAC with Approximation in Two Pieces

In Figure 3-18 blue curve is the SAC for the forward part of the ship, and the red curve is the SAC for the aft part of the ship. These two curves are separated with the cross section with the maximum sectional area. The points are the control vertices of the curves and the dashed lines are the defining control polygons.

The third and the last method is pretty similar to the second method, but instead of defining the curve in two pieces, this method defines the SAC as a single piece. So instead of two separate displacement and LCB values, this method requires only one input for each. The parameters required by this method is listed in Table 3.9.

<b>Input Parameters for Sectional Area Curve with Fwd and Aft Parts</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
1	LBP	m	length between perpendiculars
2	sacEntranceAngFwd	deg	determines tangency constraint for the fwd part of the SAC
3	sacEntranceAngAft	deg	determines tangency constraint for the aft part of the SAC
4	LCBfwd	m	ratio of the longitudinal location of max beam and lbp
5	LCBaft	m	determines tangency constraint for the forwardmost point of DWL curve
6	dispVolumeFwd	m <sup>3</sup>	determines tangency constraint for the aftmost point of DWL curve
7	dispVolumeAft	m <sup>3</sup>	waterplane area coefficient for aft part
8	maxBeamArea	m <sup>2</sup>	waterplane area coefficient for fwd part
9	transomArea	m <sup>2</sup>	changes the longitudinal position of the weight point on forward part of DWL

Table 3.8: Input for SAC with Approximation in Two Pieces

<b>Input Parameters for Single Part Sectional Area Curve</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
1	LBP	m	length between perpendiculars
2	sacEntranceAngFwd	deg	determines tangency constraint for the fwd part of the SAC
3	sacEntranceAngAft	deg	determines tangency constraint for the aft part of the SAC
4	LCB	m	ratio of the longitudinal location of max beam and lbp
5	dispVolume	m <sup>3</sup>	waterplane area coefficient for aft part
6	maxBeamArea	m <sup>2</sup>	waterplane area coefficient for fwd part
7	transomArea	m <sup>2</sup>	changes the longitudinal position of the weight point on forward part of DWL

Table 3.9: Input for SAC with Approximation as a Single Piece

The reason for having 2 additional methods over the interpolation method for the generation of SAC might raise some questions on the reader's mind. During the development of PHull it was seen that the effect of SAC on the cross section's form is greater than the effect of the deadrise and flare angles. Which means that small changes in the deadrise and flare angle curves will not change the form of the cross sections and hull significantly, but even a slight modification of SAC will have a considerable impact on the hull form. So the second method gives the user the chance to control the form of the SAC more strictly, and again during the development stage good shape representation results were obtained using the second method instead of the interpolation method. On the other hand, the third method makes the definition of SAC simpler, but it really doesn't come up with the same amount of controllability as the second method provides. Furthermore the second method also makes it possible to change the form of the ship more locally.

### **3.3.7 Flare Angle at the Weather Deck Curve**

Flare angle at the weather deck curve transmits the data about the tangency constraints for the top end points of the above water cross section curves. So this curve does exactly the same job with the flare angle curve, but it only provides the data for the above water part of the cross section curves. In Figure 3-12 the control vertex that the flare angle at weather deck curve controls can be seen within the brown circle.

The flare angle at weather deck curve that is obtained with the interpolation of 21 data points can be seen in Figure 3-19, whereas the one obtained with the approximation method is shown in Figure 3-20. The set of parameters used for the approximation method is listed in Table 3.10.

### **3.3.8 Width of the Keel Curve**

Final control curve that is used by PHull is the width of the keel curve. This curve was not part of the plan in the beginning of the thesis, but it was devised because of a

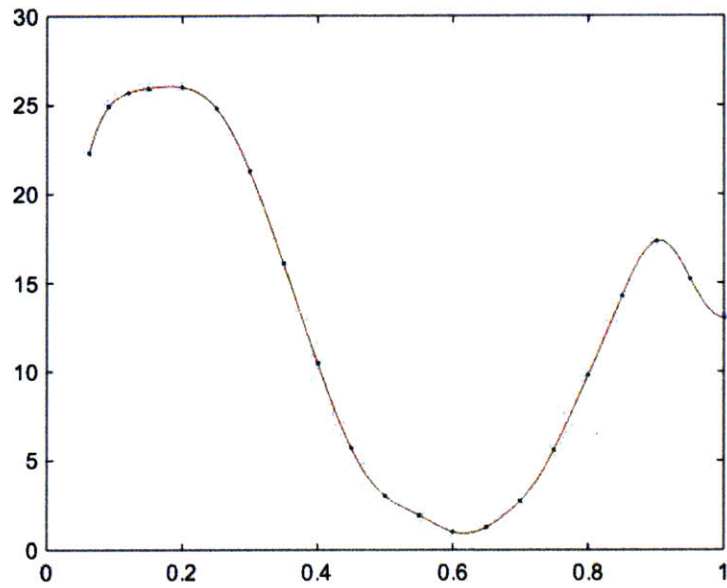


Figure 3-19: Flare Angle at the Weather Deck Curve with Interpolation Method

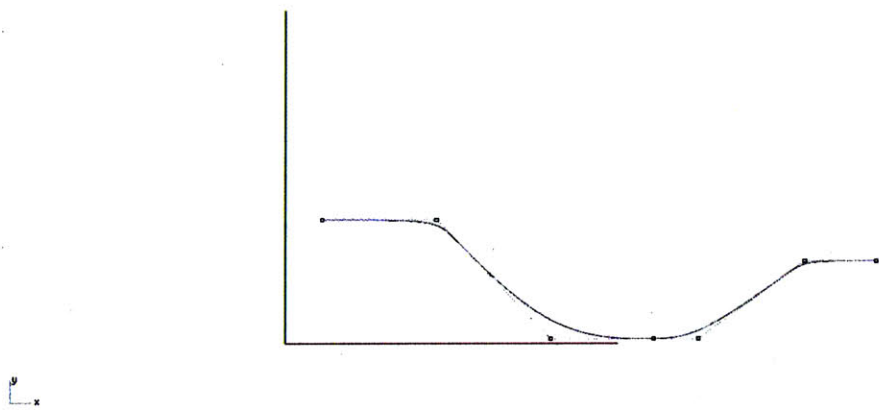


Figure 3-20: Flare Angle at the Weather Deck Curve with Approximation Method

<b>Input Parameters for Flare at Weather Deck Control Curve Approximation</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	T	m	draft
<b>3</b>	StemRadius/T	nd	determines the position of stem rise point
<b>4</b>	1st CrossSecWDAng	deg	flare angle at weather deck for the cross section at the stem rise point
<b>5</b>	Last CrossSecWDAng	deg	flare angle at weather deck for the cross section at AP
<b>6</b>	2nd CrossSecWDAng X Position	m	longitudinal position data for the second cross section
<b>7</b>	2nd CrossSecWDAng	deg	flare angle at weather deck for the second cross section
<b>8</b>	wdAngCrv fwdEntAng	deg	determines tangency constraint for the fwd part of the flare angle at weather deck curve
<b>9</b>	wdAngCrv aftEntAng	deg	determines tangency constraint for the aft part of the flare angle at weather deck curve

Table 3.10: Input for Flare Angle at the Weather Deck Curve with Approximation Method

necessity during the development phase. Basically this curve carries out the data for the transversal position of the bottom points of the underwater cross sections. It is only to be used with hulls, whose deadrise doesn't start directly from the centerline. In Figure 3-12, the distance between the centerline and the control point within the red circle (which is also the bottom point for the cross section obtained from the profile curve) is controlled by this curve.

The keel width curve obtained with the interpolation can be seen in Figure 3-21, and the one obtained with the approximation method is presented in Figure 3-22. The set of parameters used for the approximation method is listed in Table 3.10.

### 3.4 Representing the Hull Form Using the Control Curves

The control curves, and their importance in the hull form generation in PHull were discussed in the previous section. Which part of the cross section each control curve controls was also introduced. But the ultimate purpose of PHull is to produce

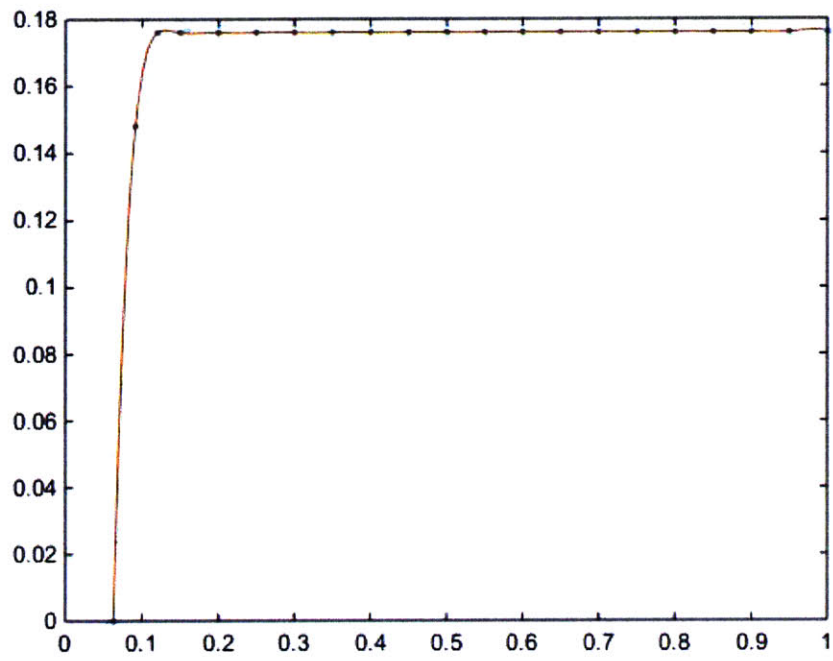


Figure 3-21: Keel Width Curve with Interpolation Method

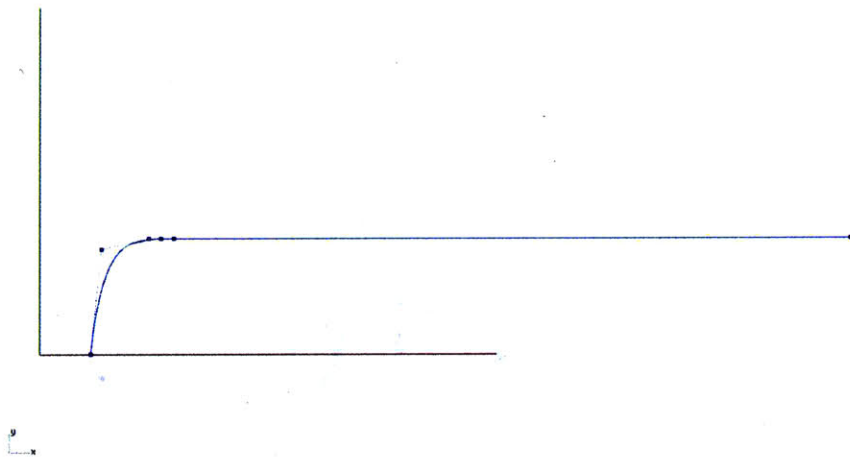


Figure 3-22: Keel Width Curve with Approximation Method

<b>Input Parameters for Keel Width Control Curve Approximation</b>			
<b>#</b>	<b>Parameter Name</b>	<b>Unit</b>	<b>Explanation</b>
<b>1</b>	LBP	m	length between perpendiculars
<b>2</b>	T	m	draft
<b>3</b>	StemRadius/T	nd	determines the position of stem rise point
<b>4</b>	1st CrossSecKW	m	keel width for the cross section at the stem rise point
<b>5</b>	Last CrossSecKW	m	keel width for the cross section at AP
<b>6</b>	kwStraightLinePt xPos	m	longitudinal position for the start of the straight segment of keel width curve
<b>7</b>	kwCrv fwdEntAng	deg	determines tangency constraint for the fwd part of the keel width curve
<b>8</b>	kwCrv aftEntAng	deg	determines tangency constraint for the aft part of the keel width curve

Table 3.11: Input for Keel Width Curve with Approximation Method

the hull form representation itself, and provide the relevant data to the CFD tool for hydrodynamic analysis. So, the focus of this section will be the generation of the hull form representation from the control curves.

It's important to underline that PHull has the ability to provide hull form data for hydrodynamic analysis with both the RANS and the panel methods. It has different pieces of codes that generates the hull form with different number of cross sections for the desired CFD tool. This will be discussed with more detail further in this section.

### 3.4.1 Cross Section Curves

It will not be an overstatement to say that the purpose of PHull is to generate the cross section curves. At least this is the first step of the process. All the control curves that were introduced so far are for defining the cross section curves along the length of the ship. By only looking at the cross section curves a naval architect can get a good feeling about the characteristics of the hull. With an adequate number of cross section curves it is possible to generate an accurate hull surface with lofting method or to perform hydrodynamic analysis for the hull. To put it shortly, generating the cross section curves means generating the hull.

Cross section curves in PHull are generated in two parts, namely the underwater



and the above water parts. The underwater part of a cross section curve can be seen as the blue curve and the above water part as the red curve in Figure 3-23. Underwater parts of the cross section curves are created as NURBS curves by PHull, whereas the above water parts are non-rational (regular, without the weight parameter) B-splines.

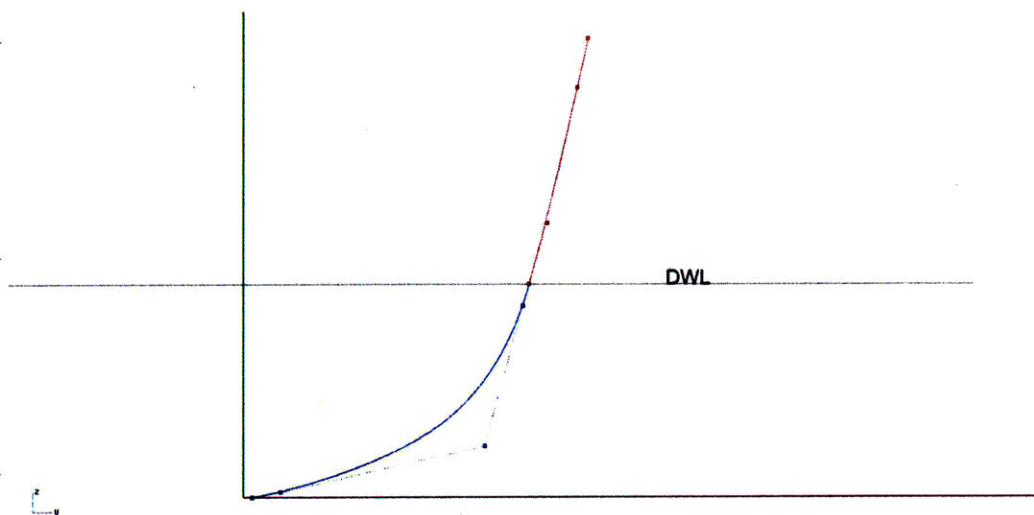


Figure 3-23: Underwater and Above Water Parts of a Cross Section Curve

How these parts are generated will be discussed next.

### The Underwater Part

PHull generates the underwater cross sections with 5 control vertices. Figure 3-24 displays the underwater part of the cross section curve and its 5 control vertices. Note that the red vertex on top is exactly on the DWL and commonly used by both the underwater and the above water parts.

For a given x coordinate,

- The position of the bottom control point is determined by the profile curve and the keel width curve. Profile curve provides the z coordinate for this point, whereas the keel width curve determines the y coordinate.
- Bottom tangency constraint point's (second control point from the bottom) y coordinate is assigned automatically by the PHull as the one-tenth of the

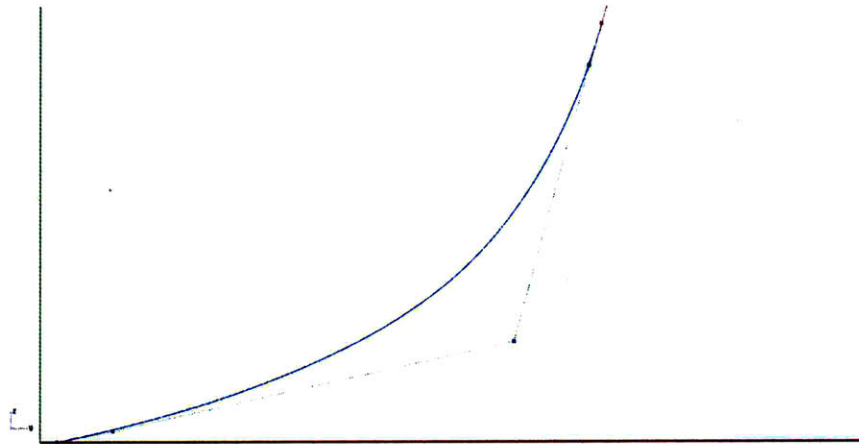


Figure 3-24: Underwater Cross Section Curve

beam width for this cross section. The  $z$  coordinate is calculated with the data provided from the deadrise angle curve.

- The position of the top control point is directly given by the DWL curve.
- Top tangency constraint point's position is found in a way similar to bottom tangency constraint point's position. This time PHull automatically assigns the  $z$  coordinate as the nine-tenths of the draft, and the  $y$  coordinate is calculated with the data from the flare angle curve.
- The weight of the intermediate control point is determined by the SAC. But the position of the intermediate control point is quite important for the general form of the cross section curve as well, and is determined with two different methods depending on the longitudinal position of the cross section.
  1. If the longitudinal position of the cross section is greater than the 40% of the LBP, then the position of the intermediate point is calculated as the intersection of the two lines connecting the top end point with its tangency constraint point, and the bottom end point with its tangency constraint point. For the cross section in Figure 3-25 intermediate control point's position is calculated with this method.

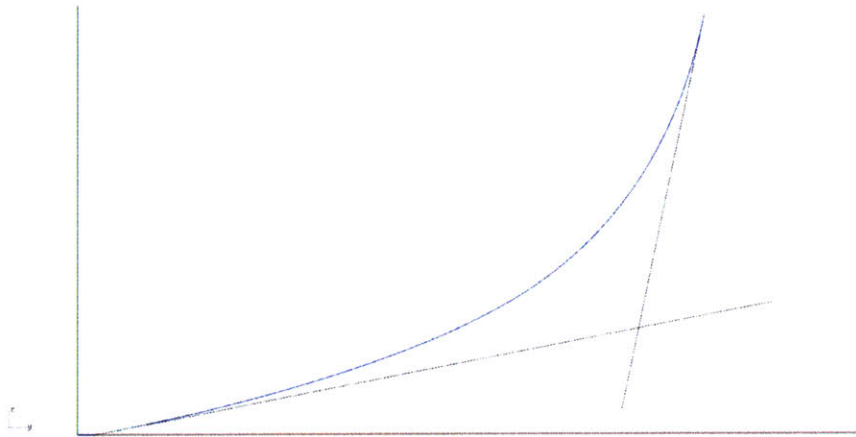


Figure 3-25: Intermediate Control Point as the Intersection of Two Lines

2. However, when the longitudinal position of the cross section is smaller than the 40% of the LBP another approach is used by PHull. In this approach two cubic spline fits, that are created using data from 3 cross sections, are used to determine the y and the z coordinates of the intermediate control point. An example is presented in Figure 3-26. In this figure the cross section curve is the blue curve, and the control point within the red circle is the intermediate control point for this curve, position of which is obtained from the two cubic spline fits. The point within the blue circle, which is at the intersection of the two black lines, shows where the intermediate control point would be if it were calculated with the first method.

The reason for using two different methods to find the position of the intermediate control point is simply the difference of the form of the cross sections for the forward part of the ship from the rest. For example in Figure 3-27 two cross section curves are displayed together, one is the cross section curve at stem rise point and the other one is the cross section curve at amidships.

It is obvious that the forms of these two curves are significantly different. Using the intersection of two lines approach yields a good position for the intermediate control point of the cross section at amidships, but it yields a position with really small z coordinate value for the cross section at the stem rise point as can be seen

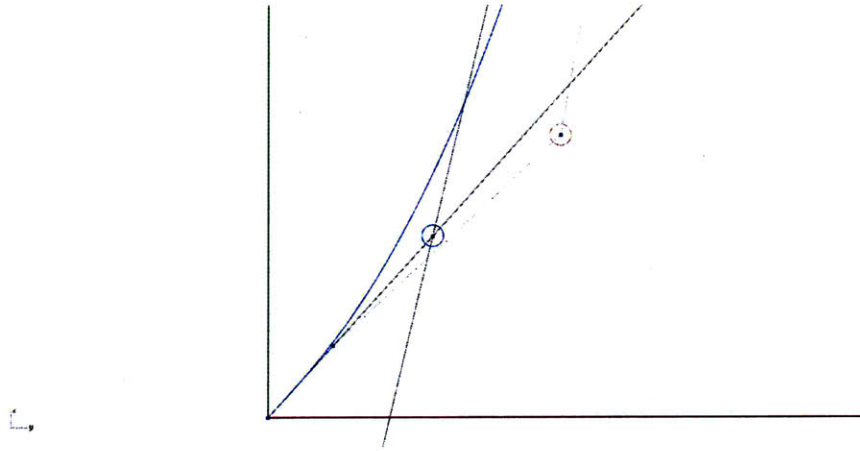


Figure 3-26: Intermediate Control Point as Obtained from the Cubic Spline Fit

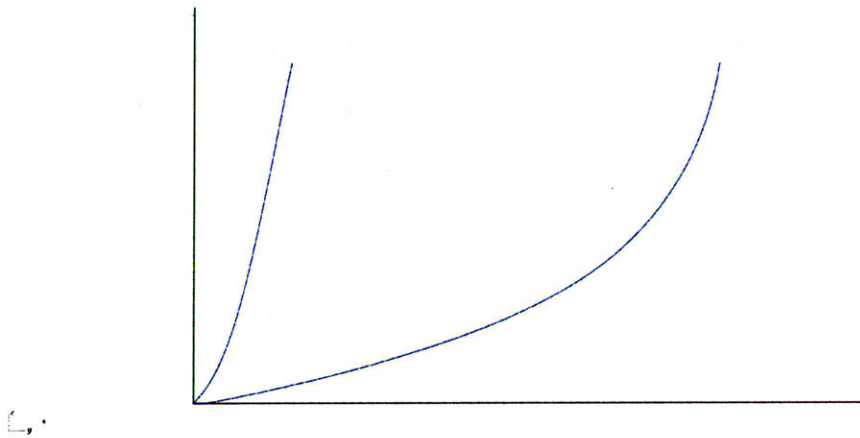


Figure 3-27: Comparison of the Cross Section Forms for Bow and Amidships

from Figure 3-26. So the method that uses cubic spline fit is being employed by PHull to achieve better positions for the intermediate control points of the cross sections in the forward part of the hull.

### The Above Water Part

The above water parts of the cross section curves are generated with 4 control points by PHull. An example for the above water part of a cross section is shown in Figure 3-28 as the red curve.

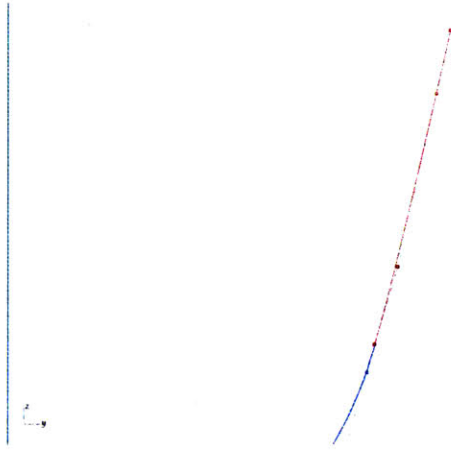


Figure 3-28: Above Water Cross Section Curve

For a given x coordinate,

- The position of the bottom control point is determined by the DWL curve. As stated above, this point is the same with the top end point of the underwater cross section curve.
- Bottom tangency constraint point's z coordinate is again determined by the flare angle control curve, the reason for this application will be discussed soon. The z coordinate for this point is automatically assigned by PHull in a way that the continuity between the underwater and the above water parts of the cross section curve will be achieved.

- The position of the top control point is directly given by the edge of the weather deck curve.
- PHull calculates the z coordinate for the top tangency constraint point by subtracting one-fifth of the freeboard height at the section from the height of the edge of the weather deck curve again at this section. Then the y coordinate for the point is determined by the flare angle at the weather deck curve.

The continuity between the two parts of the cross section is assured by providing the same tangency constraint for both sides of the control vertex at DWL, which is the common control vertex for both parts. Figure 3-29 displays this cross section from a close view. Again the red curve represents the above water part, blue curve represents the underwater part, and the black line represents the DWL. The control point at the DWL and its associated tangency control points on both sides can also be seen from this figure. Note that these three control points sit on a straight line. This was intentionally arranged by PHull in order to assure the continuity.

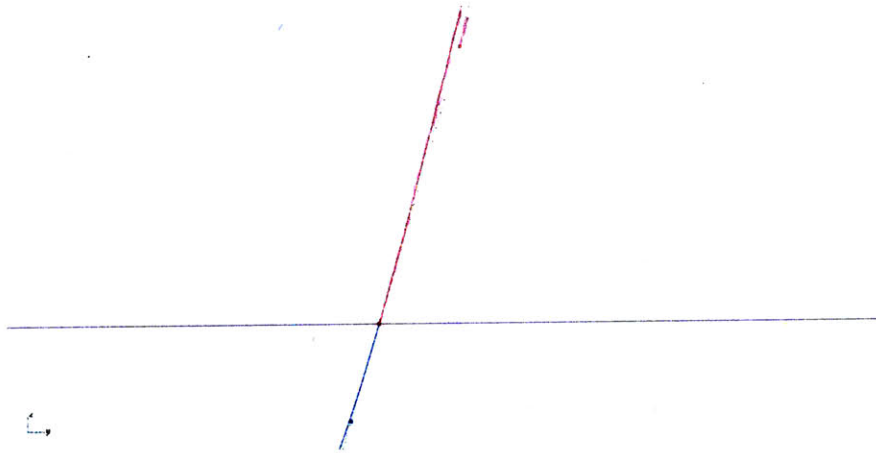


Figure 3-29: Transition Region between Two Parts of the Cross Section Curve

### 3.4.2 Surface Generation

So far, it was mentioned a few times that PHull has the ability to generate the hull surfaces as well. And this feature makes it possible for the output of PHull to

be evaluated in a CFD tool that utilizes the RANS method. But how the surfaces are generated in PHull hasn't been discussed so far. The following part will provide detailed information about the generation of the hull surfaces in PHull.

First thing that has to be mentioned in this part is PHull generates only the port side of the hull and then mirrors it to obtain the complete hull. This means that PHull only generates symmetric hulls. This is exactly the case for surfaces with two exceptions, first the ones on the port side are generated, and then these surfaces are mirrored to generate the ones on the starboard side. The exceptions are related to the generation of the transom surfaces and will be discussed in the relevant part.

There are 7 different surfaces that are generated by PHull for hull form representation. These surfaces are:

- Stem Underwater Surface
- Stem Above Water Surface
- Main Body Underwater Surface
- Main Body Above Water Surface
- Transom Underwater Surface
- Transom Above Water Surface
- Keel Width Surface

From these, stem underwater, stem above water, transom underwater, and the transom above water surfaces can be grouped as the patch surfaces. Basically these surfaces are defined by their boundaries. Stem underwater (dark red surface) and stem above water (bright red surface) surfaces can be seen in Figure 3-30. Stem underwater surface is defined by the underwater stem curve, DWL curve, and the underwater part of first cross section curve at stem rise point. In a similar fashion stem above water surface is defined by above water stem curve, DWL curve, edge of the weather deck curve, and the above water part of first cross section curve at stem rise point.

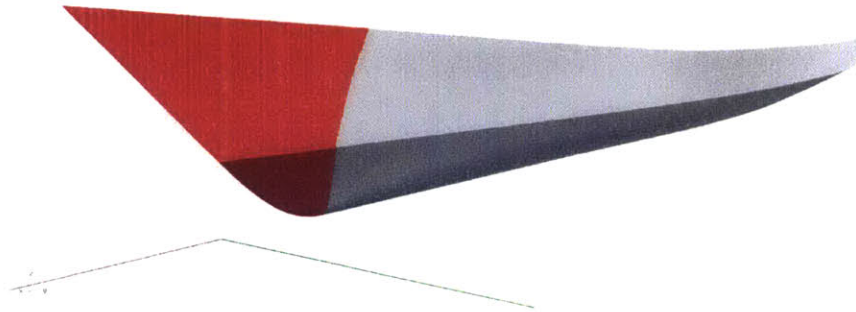


Figure 3-30: Stem and Main Body Surfaces

Transom surfaces are the only surfaces that are not mirrored in PHull. Because these two surfaces are generated as the final step, after everything else have been mirrored. So the transom underwater surface is defined by the underwater parts of the last cross sections at both the port and the starboard sides, and the DWL curve. Similarly the transom above water surface is defined by the above water parts of the last cross sections at both the port and the starboard sides, DWL curve, and the edge of the weather deck curve. These surfaces can be seen in Figure 3-31, underwater part is the surface in dark blue, and the above water part is in bright blue colors. Note that the transom underwater surface is really small in this particular example and one might need to look carefully to see it.



Figure 3-31: Transom Surfaces



The rest of the surfaces, namely main body underwater, main body above water, and keel width surfaces can be grouped together as the lofted surfaces. Main body underwater (dark grey) and the main body above water (bright grey) surfaces can be seen from Figure 3-30. The keel width surface is created around the centerline of the ship, adjacent to the main body under water surface, if the keel width curve is defined. All these surfaces are created from the stem rise point to the aft perpendicular, the same interval with the control curves, by skinning through the generated cross sections.

In theory control curves provide the data for the generation of infinite cross section curves. If infinite number of cross sections can be created with this data, main body surfaces would be obtained automatically. But in practice defining infinite number of cross sections from the control curves is not possible. But keeping the number higher will result in more accurate surfaces. The number of cross sections that PHull uses in order to create the lofted surfaces is 334, which is found to be a good number after some trial and error.

For both types of surface generation iGeo design library provides the necessary functions, and they are being utilized by PHull.

### **3.5 Validation of PHull with FFG-7 and ATHENA Hull Forms**

Once the coding part was completed, it was the time for the validation of PHull. It was thought that the best way to validate the tool is to use it for shape representation, and then to compare the resulting models with the existing real models. So, the FFG-7 and the ATHENA Model 5365 were selected as the validation cases. These choices are not surprising in fact, considering that PHull is specifically developed for high speed displacement monohulls. The CASHD model for FFG-7 was obtained from NAVATEK Ltd in .3dm format. For the ATHENA hull an accurate CASHD model couldn't be found, so the offset table and the ship lines given in the paper from David

W. Taylor Naval Ship Research and Development Center [21] were transferred into a CASHD model using Rhinoceros. These CASHD models were used as both the sources for input data, and the references for the comparison.



Figure 3-32: ATHENA Hull

First csv files were prepared as input files for both hulls, using the data derived from the real models. Then PHull generated the hull representations from this data. Only part that took some time was to derive all the necessary data from the existing model, but using an efficient tool as Rhinoceros made this part easier. In the following part the comparison of the profile curves, DWL curves, edge of the weather deck curves, and the cross section curves between the real model and the model created by PHull are presented.

### 3.5.1 Comparison of the Profile Curves

Profile curve is the first curve that PHull generates, and it directly controls the bottom point of the cross section curves. So, for shape representation accuracy of the profile curve is quite important. If a good accuracy for the profile curve cannot



Figure 3-33: FFG-7 Hull

be obtained this will deteriorate the form of the cross sections, because the tool will generate the section curve with the same area constraint without any consideration on the location of the bottom point. The profile curve comparison between the real model and the model generated by PHull for ATHENA Model 5365 and for FFG-7 are presented in Figure 3-34 and Figure 3-35 respectively. In the figures red curves are from the original models, whereas the blue curves are from the output of PHull.

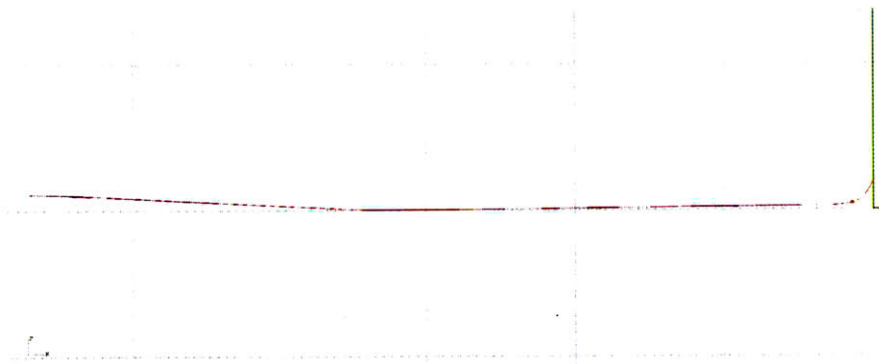


Figure 3-34: Profile Curve Comparison for ATHENA Model 5365

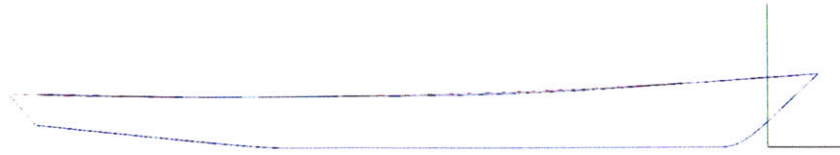


Figure 3-35: Profile Curve Comparison for FFG-7

From both images it is possible to see that a very good representation of the profile curves are achieved for both cases.

### 3.5.2 Comparison of the DWL Curves

The second curve that PHull generates is the DWL curve, which controls the top point of the underwater parts of the cross section curves, and the bottom point of the above water parts. The accurate representation of this curve is also of great importance, following the same reasoning with the profile curve. DWL curve comparison between the real model and the model generated by PHull for ATHENA Model 5365 and for FFG-7 are presented in Figure 3-36 and Figure 3-37 respectively. In the figures red curves are from the original models, whereas the blue curves are from the output of PHull.

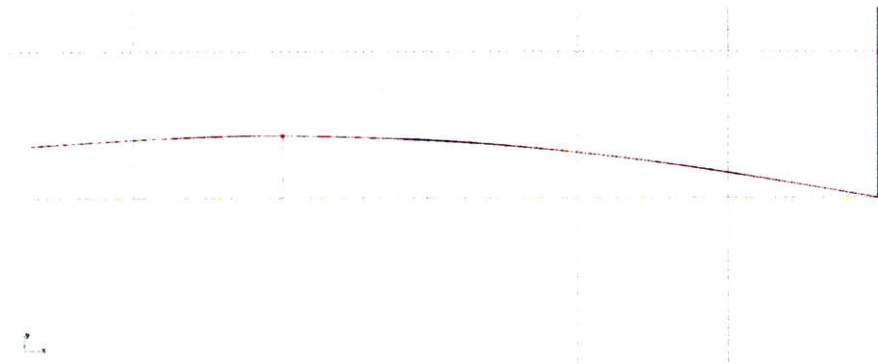


Figure 3-36: DWL Curve Comparison for ATHENA Model 5365

Again, it's possible to see from these figures that PHull did a pretty good job at modelling the DWL curves for both cases.



Figure 3-37: DWL Curve Comparison for FFG-7

### 3.5.3 Comparison of the Edge of the Weather Deck Curves

Edge of the weather curve is the third curve that PHull generates, and the last curve before the generation of the cross section curves. It controls the top points of the above water part of the cross section curves. Since it only controls a portion of the hull that is above the waterline and PHull generates the hull forms for hydrodynamic analysis, the accurate representation of this curve is not as important as the previous ones. But still, in order to have a fair curve for the above water part of the cross sections PHull tries to achieve an accurate representation for this curve as well. The comparisons for this curve are presented in Figure 3-38 and Figure 3-39 for ATHENA Model 5365 and for FFG-7 respectively. In the figures red curves are from the original models, whereas the blue curves are from the output of PHull.

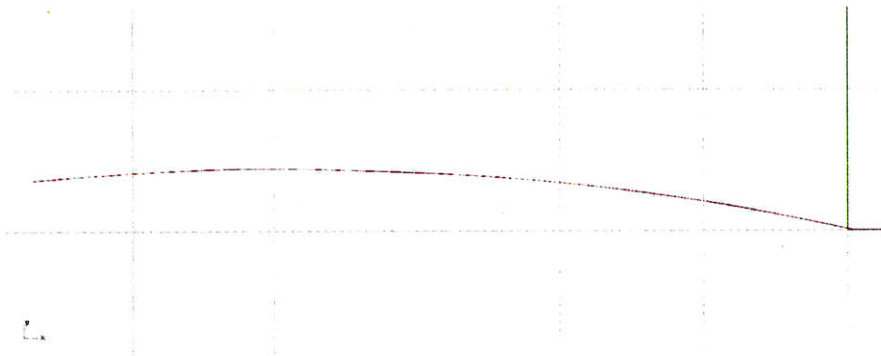


Figure 3-38: Edge of the Weather Deck Curve Comparison for ATHENA Model 5365

Looking at the figures it can be said that accurate representations are achieved for this curve as well. But note that there are some small discrepancies on the forward part of the FFG-7 representation, which were considered to be insignificant.

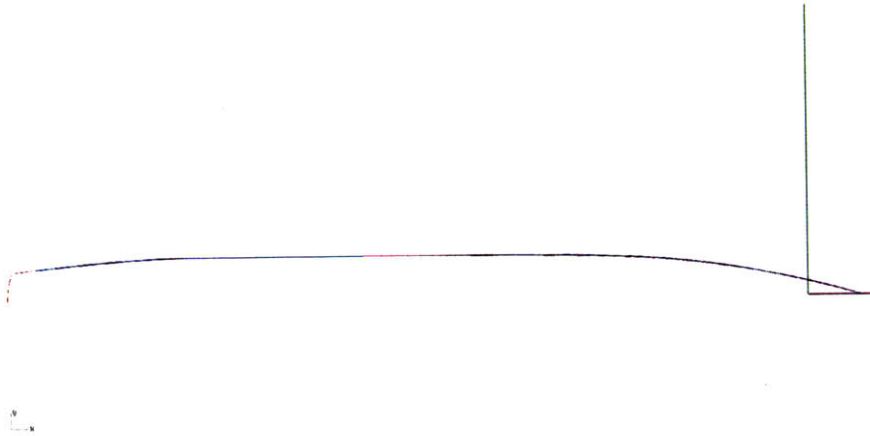


Figure 3-39: Edge of the Weather Deck Curve Comparison for FFG-7

### 3.5.4 Comparison of the Cross Section Curves

So far it was discussed a few times that, the real purpose of PHull is to generate the cross section curves. Which means that the comparison between the original cross sections and the ones created by PHull is the most important. Accurate representations of the cross sections is the thing that is sought after for the validation of PHull.

However, there is one thing to note here. As discussed earlier cross section curves are created with the data coming from 8 different control curves, profile curve, DWL curve, edge of the weather deck curve, sectional area curve, deadrise angle control curve, flare angle curve, flare angle at weather deck curve, and the keel width curve. And none of these curves perfectly represents the data from the real model, meaning that there is always a small amount of error coming from the control curves. Even though these discrepancies are small for their respective control curves, cross section representation suffers from the accumulation of them. So even before the validation cases, it was known that there would be some discrepancies at the cross section curve representations. The aim was to have these discrepancies at an acceptable level.

Comparison for the ATHENA Model 5365 is presented in Figure 3-40. In the figure red straight line is the design waterline, and as in the figures for the previous curves the red curves are belong to the original model whereas the blue curves are

from the model created by PHull.

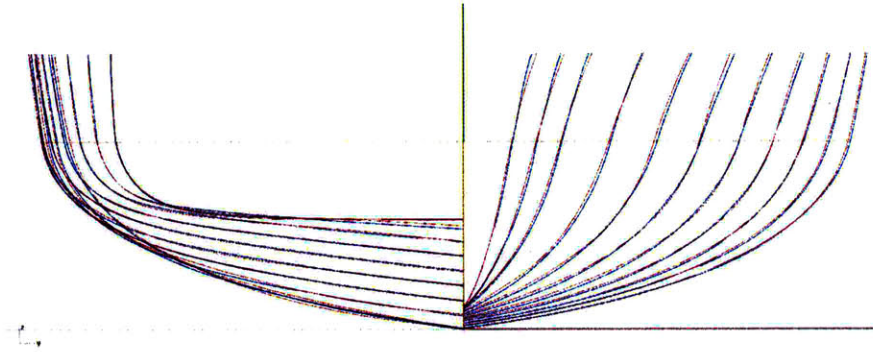


Figure 3-40: Cross Section Curve Comparisons for ATHENA Model 5365. Sectional curves are colored in red (original) and blue (PHull).

From Figure 3-40 it can be seen that there are some discrepancies in the cross section curves as expected. But the curves generated by PHull generally follows the original curves. There are some sections that has an error in the position of their bottom points, and there are some with small errors at their end point at DWL. Basically these errors resulted in some additional errors at the in between parts of the cross sections, but nothing significant. As discussed previously, the errors at the weather deck level are not a cause of concern. So overall, an accurate representation of the ATHENA Model 5365 was achieved by PHull, which was considered to be appropriate for hydrodynamic analysis.

For FFG-7 the comparison for the forward part is presented in Figure 3-41 and for the aft part is presented in Figure 3-42. Color convention for the curves is consistent with the figures for ATHENA Model 5365, however the design water line is shown as the thick blue line in these figures.

Pretty much all the comments made for the ATHENA case are also valid for FFG-7 case. In short, there are some discrepancies as expected, but none of them is significant. So, PHull generated an accurate model for FFG-7 as well.

As the final step of the hull generation process PHull creates the surfaces, if it's required. In Figure 3-41 the surfaces generated for ATHENA Model 5365 are presented. Note that this is the final form of the hull that PHull creates.

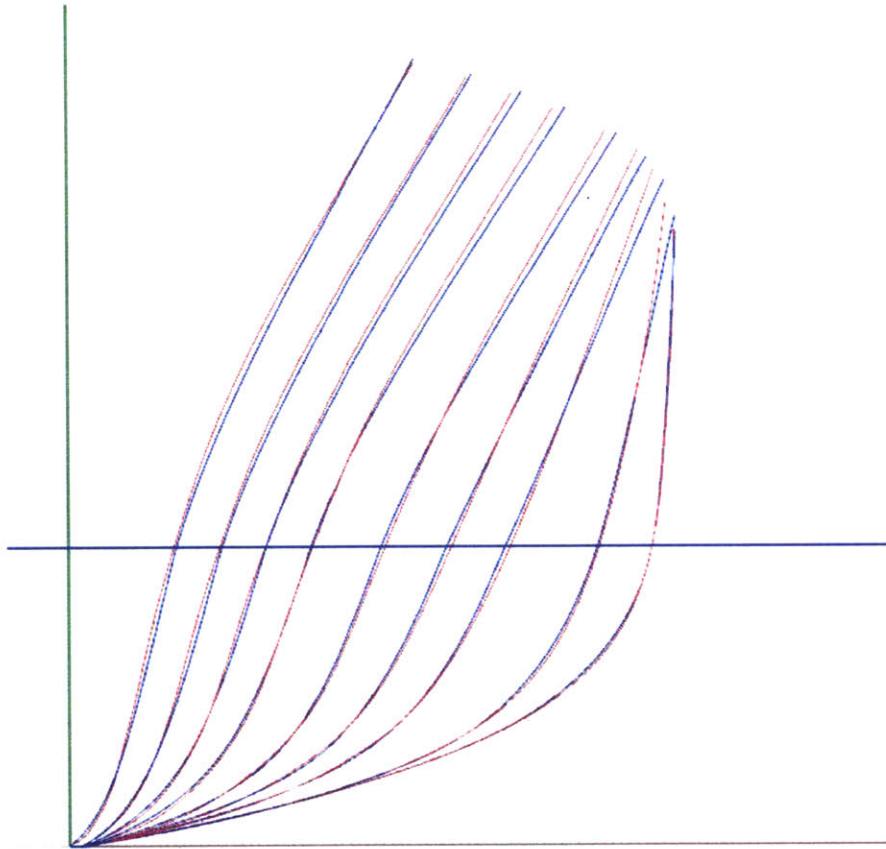


Figure 3-41: Cross Section Curve Comparisons for Forward Part of FFG-7. Sectional curves are colored in red (original) and blue (Phull).



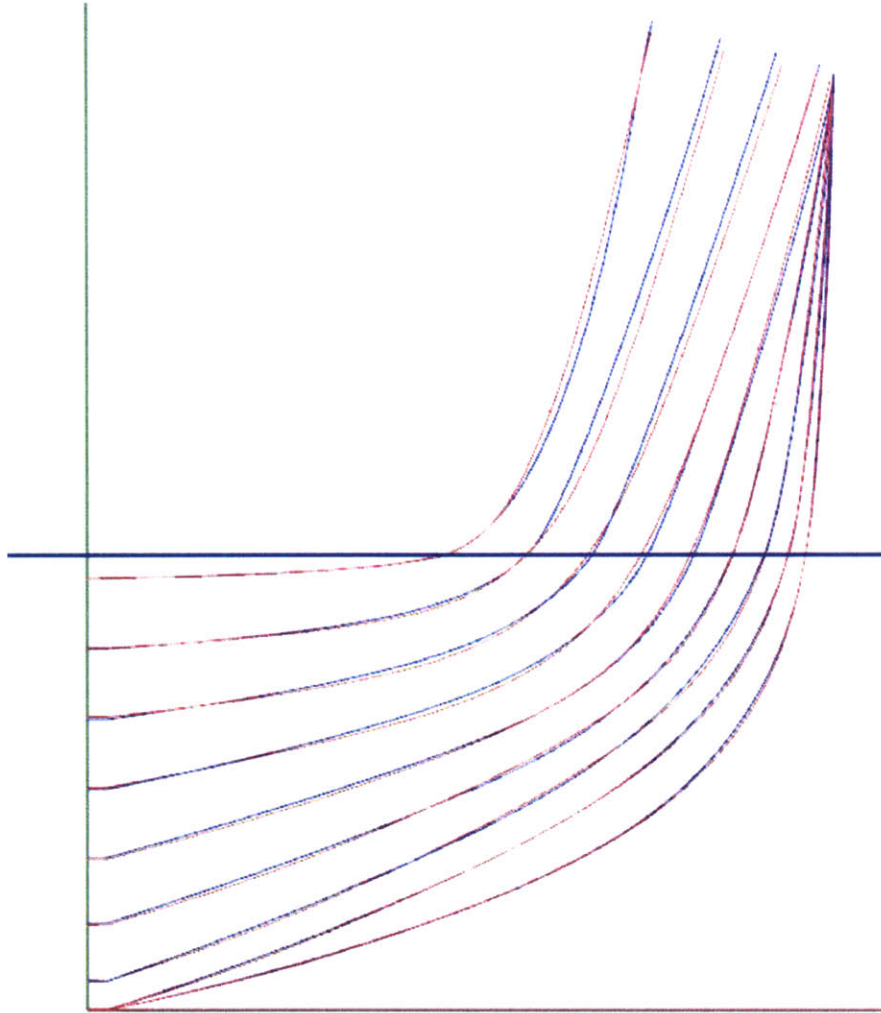


Figure 3-42: Cross Section Curve Comparisons for Aft Part of FFG-7. Sectional curves are colored in red (original) and blue (Phull).

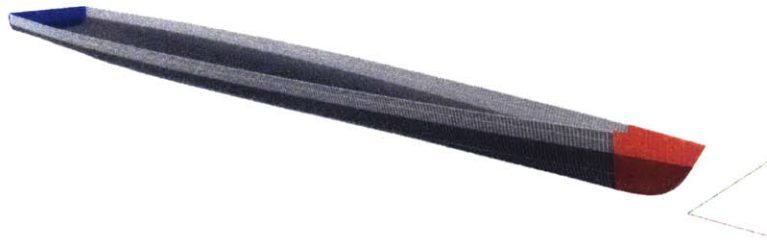


Figure 3-43: Surfaces for the ATHENA Model 5365 as Generated by PHull

The surfaces generated for FFG-7 are presented together with the real model in Figure 3-44. In this figure the model on the top is the real model whereas the one on the bottom was generated by PHull.

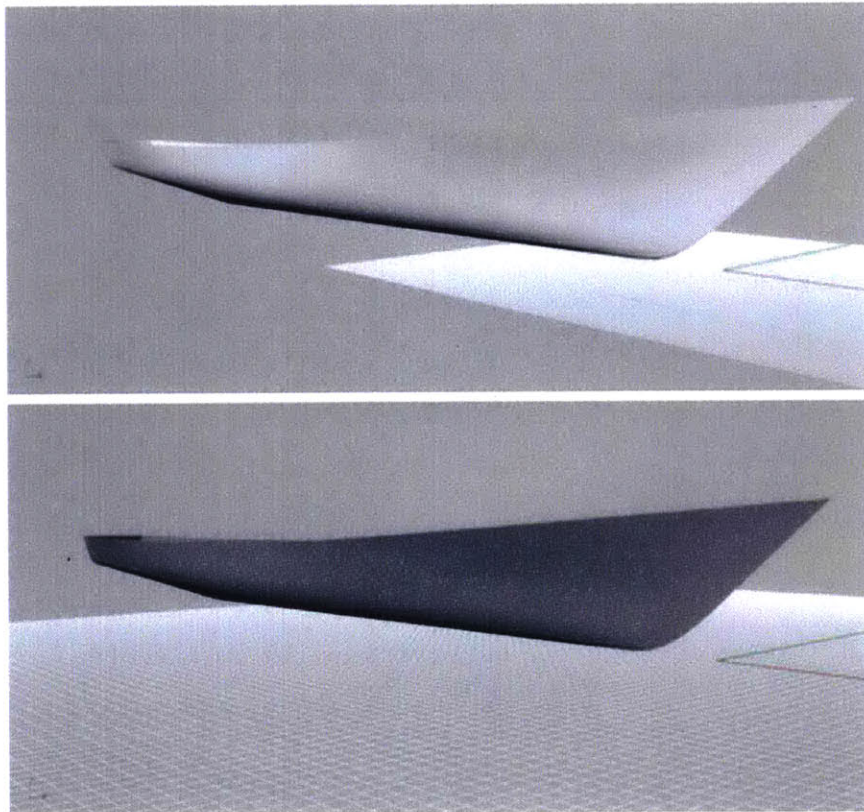


Figure 3-44: Comparison of the Original FFG-7 Model (top) and the FFG-7 Model Obtained with PHull

Of course it's not easy to compare and evaluate two hulls with solely examining by eyes, but the two models in Figure 3-44 look quite similar.



## Chapter 4

# Parametric Hull Form Modifications with PHull

As stated in the introduction parts, PHull is primarily developed for hydrodynamic optimization of hulls. Of course it is important for PHull to represent or create a hull form for the very first time. But after that it also has to provide the necessary tools to modify this initial hull form for optimization.

Being a fully parametric modeling tool, PHull is inherently good for this task. It is possible to modify the hull form by changing some of the parameters, that are coming from the input csv file. So the parametric modelling tool generates the hull form from scratch after every set of modifications, with the new set of input parameters. This was the method of choice since it is also appropriate for the optimization algorithm, which is the differential evolution algorithm. Furthermore, with this approach it's possible to keep the desired variables fixed while modifying the hull form, since the modified hull will have the same set of parameters except the ones that were modified. So not changing the variables is enough to keep them fixed.

This chapter will focus on to the parametric hull modification ability of PHull. Firstly a few simple examples of parametric hull form modifications on FFG-7 will be discussed to give some intuition about the ability of PHull. Then, a simple approach will be introduced for systematic hull form generation with PHull by only changing a set of parameters. Finally how PHull can be used with the differential evolution

algorithm will be discussed.

## 4.1 Simple Examples of Parametric Hull Form Modification with PHull on FFG-7

The purpose of this section is to show how a modification on a single parameter will affect the hull form, to give an idea about what can be achieved with the parametric modeling tool and what are the limits, and to convince the reader that it is a viable tool.

First of all there are no parameters that the user cannot reach in PHull, so every parameter can be changed. But this does not mean that every parameter should be changed independently. As any naval architect can easily figure out some parameters will have interdependencies. A good example for these interdependencies is the cross section curve. If someone only increases the displacement of the ship while keeping everything else fixed, remember that the displacement parameter controls the SAC and SAC controls the sectional areas, after one point it will not be possible to achieve the sectional area with the given deadrise and flare angles. This would either make the resulting curve an infeasible curve or make it impossible to generate the curve. So some engineering judgement have to be employed for the modifications.

In the first example DWL\_FWD\_ANGLE parameter for FFG-7, which determines the entrance angle of DWL curve in the bow, is changed from 11.02 degrees to 10 degrees. As one can say this is really a small change and it is indeed a small change, that results in a small modification of the hull form as presented in Figure 4-1.

The modification of the DWL curve and the cross sections in the forward part can be seen from Figure 4-1. As stated in the figure, blue curves are from the original hull and the red curves are from the modified one. The modifications are really subtle. But for some parameters even a small modification can yield significant changes, at least locally. For example Figure 4-2 shows how a change in KEEL\_RISE\_POINT

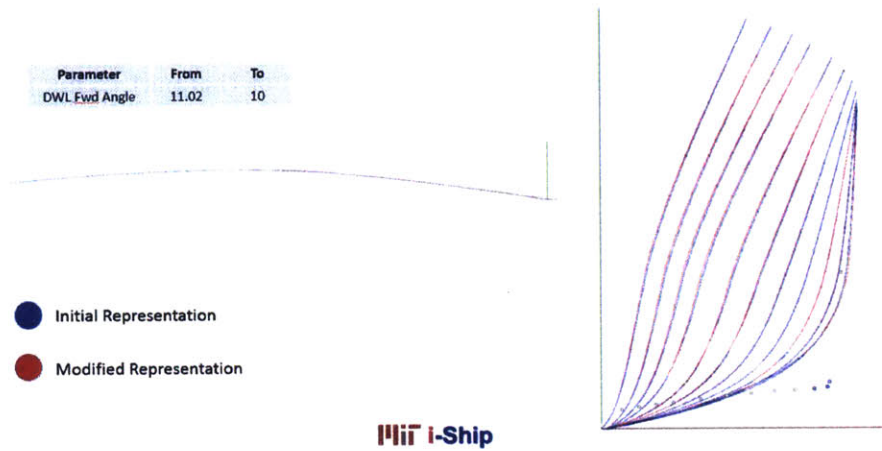


Figure 4-1: DWL Entrance Angle Parameter Modification

parameter would affect the hull form.

KEEL\_RISE\_POINT parameter determines the point where the hull starts rising in the stern part, and is given as a fraction of LBP. In this example keel rise point is moved closer to aft, as can be seen from the profile view comparisons of the initial and modified representations. This actually resulted in lower  $z$  coordinates for the cross sections around this part. But since the displacement, and the sectional areas as a consequence, hasn't been modified the upper part of the underwater cross section curves are reduced in order to compensate for the extra area coming from the keel rise point modification.

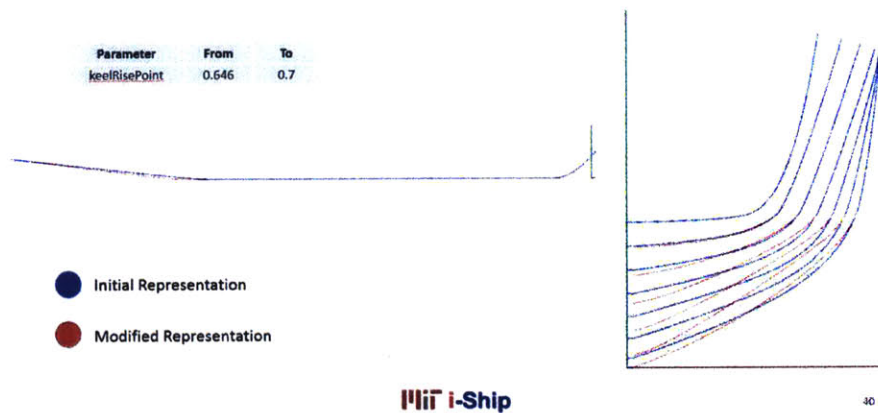


Figure 4-2: Keel Rise Point Parameter Modification

The effect of changing the longitudinal position of the section with the maximum beam is investigated for the third and the last example of this part. This parameter determines the separation point between the forward and aft part of the hull for PHull, and is given as a fraction of LBP just as the keel rise point parameter. A change in this parameter also has a significant effect on the hull form, and this time it is somewhat global as can be seen from Figure 4-3. It is not surprising, since the construction of both the forward and the aft parts depend on this parameter.

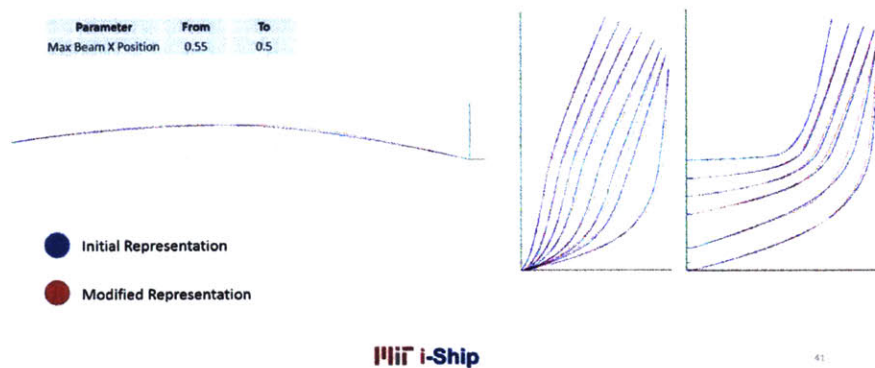


Figure 4-3: Longitudinal Position of the Section with the Maximum Beam Parameter Modification

By now it should be obvious that modifying the hull form is not a hard task with the parametric modelling tool. But of course there are things to be careful about. Interdependencies of parameters is very important, not paying attention to it may result in infeasible shapes for the curves as discussed earlier. However, the examples in this part were quite simple, only a single parameter was modified at a time. In a real scenario this will not be the case. Moreover, more than one control curve, which were introduced as the main building block of PHull, hasn't been used for these examples as well. This will be the topic of the next section.



## 4.2 A Simple Approach for the Systematic Hull Form Generation with Shape Variation Using PHull

For each of the simple examples presented in the last section only one parameter was modified at a time. And it was seen that significant modifications can be achieved up to a level with this method. However, more significant and meaningful modifications can only be achieved by considering the interdependencies of the parameters, and changing multiple parameters at a time. In order to present this a simple systematic hull form generation approach was devised using PHull, which depends on the displacement volume. In this approach, PHull assigns a new multiplier for the input displacement volume value in every new run, and generates a new hull form with a new displacement volume value. Within this approach the displacement volume is changed linearly. The variables that affect or are dependent on the displacement volume are also assigned some multipliers and changed with the displacement volume. But obviously not all of them are related to displacement volume linearly. The relationship between these parameters and the displacement volume were determined using regression analysis and with trial and error method. The parameters that are used in this example are presented in Table 4.1.

For the case of FFG-7 model, the relationship between the displacement volume and the LCB, and SAC entrance angle parameters were determined by simple linear regression analyses. Then, the relationship between the displacement volume and the DWL related parameters was found to be linear by trial and error. Whereas the relationship between the displacement volume and the deadrise, and flare angle related parameters was found to be polynomial.

For this approach 6 possible different scenarios were determined:

1. Increasing the displacement of the forward part
2. Decreasing the displacement of the forward part
3. Increasing the displacement of the aft part

#	Parameter Name	Parameter Explanation
1	Fwd Displacement	Displacement volume of the forward part of the hull
2	Aft Displacement	Displacement volume of the aft part of the hull
3	Cwp Fwd	Waterplane coefficient for the forward part of the hull
4	Cwp Aft	Waterplane coefficient for the aft part of the hull
5	SAC Entrance Ang Fwd	Entrance angle for the forward part of sectional area curve
6	SAC Entrance Ang Aft	Entrance angle for the aft part of sectional area curve
7	DWL Fwd Angle	Entrance angle for the forward part of design waterline curve
8	DWL Aft Angle	Entrance angle for the aft part of design waterline curve
9	LCB Fwd	Longitudinal center of buoyancy for the forward part of the hull
10	LCB Aft	Longitudinal center of buoyancy for the aft part of the hull
11	1st Cross Sec Deadrise Ang	First cross section's deadrise angle
12	Deadrise Ang Crv Fwd Entrance Ang	Entrance angle for the forward part of deadrise angle curve
13	Deadrise Ang at 2nd Input Position	Second positional deadrise angle value from forward
14	Last Cross Sec Deadrise Ang	Last cross section's deadrise angle
15	Deadrise Ang Crv Aft Entrance Ang	Entrance angle for the aft part of deadrise angle curve
16	Deadrise Ang at 2nd Input Position (for aft)	Second positional deadrise angle value from aft
17	1st Cross Sec Flare Ang	First cross section's flare angle
18	Flare Ang Crv Fwd Entrance Ang	Entrance angle for the forward part of flare angle curve
19	Flare Ang at 2nd Input Position	Second positional flare angle value from forward
20	Last Cross Sec Flare Ang	Last cross section's flare angle
21	Flare Ang Crv Aft Entrance Ang	Entrance angle for the aft part of flare angle curve
22	Flare Ang at 2nd Input Position (for aft)	Second positional flare angle value from aft

Table 4.1: Parameters that Are Used in the Systematic Hull Form Generation

4. Decreasing the displacement of the aft part
5. Increasing the displacement of both the aft and the forward parts
6. Decreasing the displacement of both the aft and the forward parts

the scenario of interest can be chosen via the input file. It is also necessary to fill in the parameter about the number of runs (models) required. PHull will change the multiplier of the displacement volume as many times as the value of this parameter, and consequently the value of this parameter will be the total number of different model outputs. For the examples in this section this number was chosen as 10, so 10 different models, that changes systematically, were obtained for each of the scenarios. In the following part each of these scenarios will be described briefly, and their impacts on the hull form will be presented with the images from their applications on FFG-7 hull.

The first scenario is increasing the displacement volume of the forward part. In this scenario displacement volume of the forward part of FFG-7 was increased by using the parameters presented in Table 4.2. The comparison of the initial and the

tenth model's cross section curves in the forward part (only the underwater part) is presented in Figure 4-4.

Procedure	Variable Values to Increase	Variable Values to Decrease
<b>Increase Fwd Fullness</b>	Fwd Displacement DWL Fwd Angle Cwp Fwd SAC Entrance Ang Fwd	LCB Fwd 1st Cross Sec Deadrise Ang Deadrise Ang Crv Fwd Entrance Ang Deadrise Ang at 2nd Input Position 1st Cross Sec Flare Ang Flare Ang Crv Fwd Entrance Ang Flare Ang at 2nd Input Position

Table 4.2: Parameters that Are Used for Increasing Displacement in the Forward Part

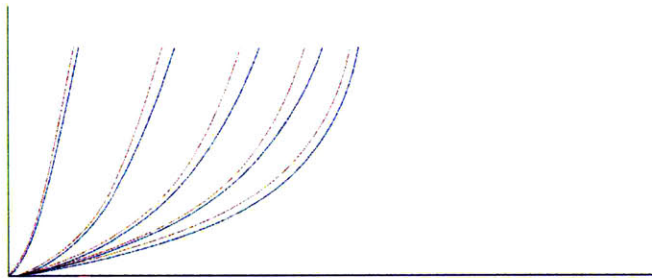


Figure 4-4: Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 1

The opposite of increasing the displacement volume in the forward part is decreasing it. All the same parameters are used with the previous scenario, but the direction each parameter changes is reversed as shown in Table 4.3. Figure 4-5 displays the variation of the cross sections for this scenario.

Procedure	Variable Values to Increase	Variable Values to Decrease
<b>Decrease Fwd Fullness</b>	LCB Fwd 1st Cross Sec Deadrise Ang Deadrise Ang Crv Fwd Entrance Ang Deadrise Ang at 2nd Input Position 1st Cross Sec Flare Ang Flare Ang Crv Fwd Entrance Ang Flare Ang at 2nd Input Position	Fwd Displacement DWL Fwd Angle Cwp Fwd SAC Entrance Ang Fwd

Table 4.3: Parameters that Are Used for Decreasing Displacement in the Forward Part

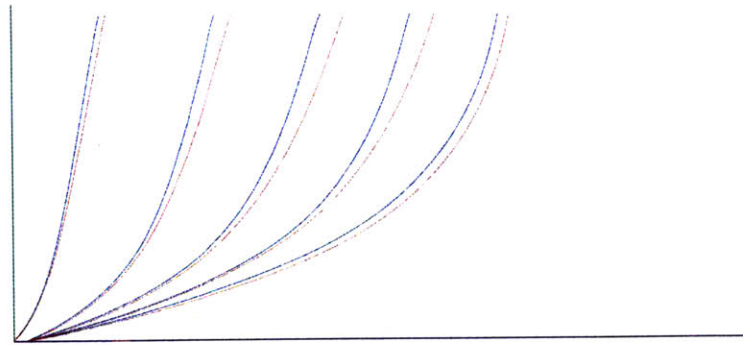


Figure 4-5: Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 2

Increasing the displacement volume of the aft part is the third scenario. In order to change the form of the aft part a different set of parameters are required. These parameters are presented in Table 4.4, and the comparison of the initial model and the tenth variant is displayed in Figure 4-6.

Procedure	Variable Values to Increase	Variable Values to Decrease
<b>Increase Aft Fullness</b>	Aft Displacement	Last Cross Sec Deadrise Ang
	DWL Aft Angle	Deadrise Ang Crv Aft Entrance Ang
	Cwp Aft	Deadrise Ang at 2nd Input Position (for aft)
	SAC Entrance Angle Aft	Last Cross Sec Flare Ang
	LCB Aft	Flare Ang Crv Aft Entrance Ang
		Flare Ang at 2nd Input Position (for aft)

Table 4.4: Parameters that Are Used for Increasing Displacement in the Aft Part

The opposite case of Scenario 3 is decreasing the displacement volume in the aft part, that uses all the same parameters but changes them in the reverse direction as presented in Table 4.5. Figure 4-7 displays the variation of the cross sections for this scenario.

Procedure	Variable Values to Increase	Variable Values to Decrease
<b>Decrease Aft Fullness</b>	Last Cross Sec Deadrise Ang	Aft Displacement
	Deadrise Ang Crv Aft Entrance Ang	DWL Aft Angle
	Deadrise Ang at 2nd Input Position (for aft)	Cwp Aft
	Last Cross Sec Flare Ang	SAC Entrance Angle Aft
	Flare Ang Crv Aft Entrance Ang	LCB Aft
	Flare Ang at 2nd Input Position (for aft)	

Table 4.5: Parameters that Are Used for Decreasing Displacement in the Aft Part

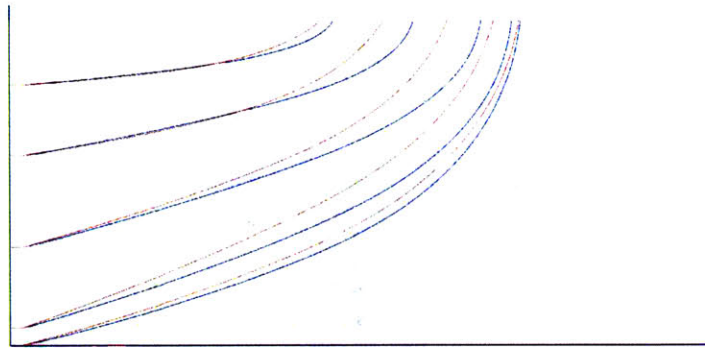


Figure 4-6: Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 3

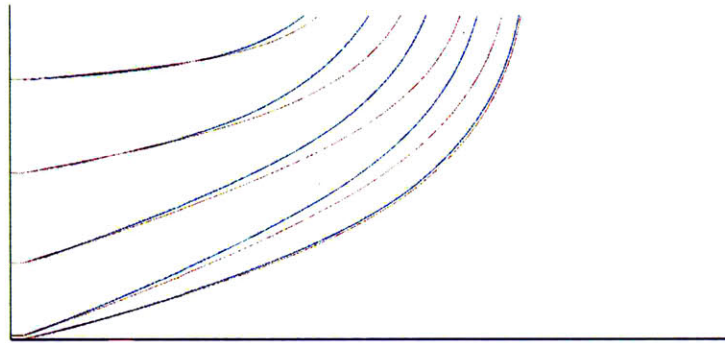


Figure 4-7: Comparison of the Cross Section Curves of the Initial Model and the Tenth Variant for Scenario 4

Increasing the displacement volume of both the aft and the forward parts, which is Scenario 5, is exactly the same with performing Scenarios 1 and 3 simultaneously. This scenario modifies the parameters that are presented in Table 4.6, which is the combination of Table 4.2 and Table 4.4. As a consequence, the resulting model would have the same cross sections as shown in Figure 4-6 and Figure 4-4.

Similarly, decreasing the displacement of both the aft and the forward parts is exactly the same with performing Scenarios 2 and 4 simultaneously. The list of parameters for this scenario is presented in Table 4.7.

Procedure	Variable Values to <i>Increase</i>	Variable Values to <i>Decrease</i>
<b>Increase Fullness</b>	Fwd Displacement	LCB Fwd
	DWL Fwd Angle	1st Cross Sec Deadrise Ang
	Cwp Fwd	Deadrise Ang Crv Fwd Entrance Ang
	SAC Entrance Ang Fwd	Deadrise Ang at 2nd Input Position
	Aft Displacement	1st Cross Sec Flare Ang
	DWL Aft Angle	Flare Ang Crv Fwd Entrance Ang
	Cwp Aft	Flare Ang at 2nd Input Position
	SAC Entrance Angle Aft	Last Cross Sec Deadrise Ang
	LCB Aft	Deadrise Ang Crv Aft Entrance Ang
		Deadrise Ang at 2nd Input Position (for aft)
	Last Cross Sec Flare Ang	
	Flare Ang Crv Aft Entrance Ang	
	Flare Ang at 2nd Input Position (for aft)	

Table 4.6: Parameters that Are Used for Increasing Displacement on Both Parts

Procedure	Variable Values to <i>Increase</i>	Variable Values to <i>Decrease</i>
<b>Decrease Fullness</b>	LCB Fwd	Fwd Displacement
	1st Cross Sec Deadrise Ang	DWL Fwd Angle
	Deadrise Ang Crv Fwd Entrance Ang	Cwp Fwd
	Deadrise Ang at 2nd Input Position	SAC Entrance Ang Fwd
	1st Cross Sec Flare Ang	Aft Displacement
	Flare Ang Crv Fwd Entrance Ang	DWL Aft Angle
	Flare Ang at 2nd Input Position	Cwp Aft
	Last Cross Sec Deadrise Ang	SAC Entrance Angle Aft
	Deadrise Ang Crv Aft Entrance Ang	LCB Aft
	Deadrise Ang at 2nd Input Position (for aft)	
	Last Cross Sec Flare Ang	
	Flare Ang Crv Aft Entrance Ang	
	Flare Ang at 2nd Input Position (for aft)	

Table 4.7: Parameters That Are Used for Decreasing Displacement on Both Parts

### 4.3 Parametric Modifications for Hydrodynamic Optimization Using PHull

PHull was developed to be integrated to evolutionary algorithms for hull form optimization. In this process a set of selected parameters are controlled by the evolutionary algorithm, and they are modified in order to achieve better results from the CFD tool.

Evolutionary algorithms are literally based on the principle of evolution. They are stochastic algorithms, with a great emphasis on randomness, that focus on population of solutions [22]. Since the evolutionary algorithms change the selected set of parameters randomly and independently, without considering any interdependencies, trying to find the relationship between the parameters is needless for the optimization case for most of the time. However, sometimes it might be helpful to define some of the variables as dependent variables to achieve better results.





# Chapter 5

## Integration with CFD and Automatic Optimization Procedure

### 5.1 Integration with the Panel Method

In order to evaluate hydrodynamic characteristics of the hulls that are generated by PHull, tool was integrated with the panel method developed by Brizzolara and Bruzzone [23].

Panel method gets its input, which is the offset table of the hull form to be evaluated, in text file format, and it provides the wave resistance coefficient of the hull form and the wave pattern that the hull generates as its outputs. PHull generates the cross section curves and the stem curve, that are necessary for the derivation of the offset data. For the integration of these two tools PHull was modified for generating only 40 cross sections and the stem curve, and then iGeo and Processing libraries were utilized to generate the offset data from these curves and store it to a text file as an output of PHull. Offset data includes the positions of 31 points for each cross section that rises directly from the keel, 33 points for each one that has a straight part before it rises, and positions of 31 points for the stem curve. These points are ordered from keel to deck for both cross section curves and the stem curve.

In order to test the integration, FFG-7 hull was used from the validation cases of PHull. Wave pattern output from this test is displayed in Figure 5-1.

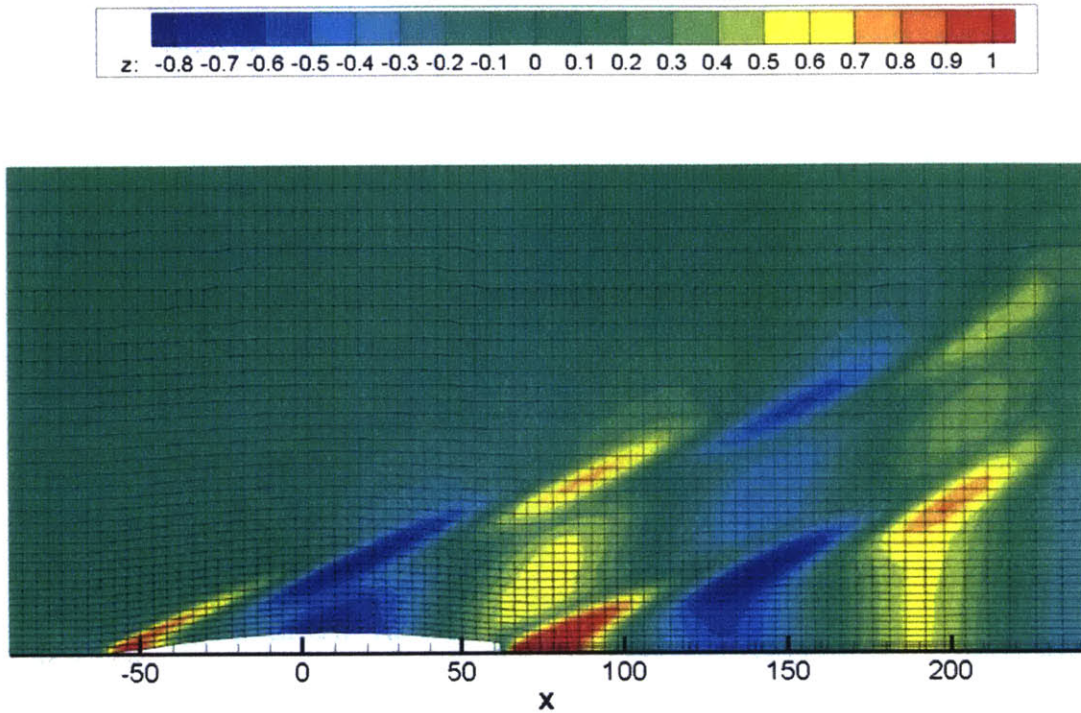


Figure 5-1: Wave Pattern Plot of FFG-7 Hull Created with the Output of the Panel Method

Whereas in left side of Table 5.1 LBP, some hydrostatic characteristics, and the wave resistance coefficient of the hull form are shown as they are given in the output file, and on the right side friction resistance coefficient (calculated with ITTC-1957 formula) and the total resistance values are presented. The total resistance is calculated in order to be used as a reference value for the optimization process.

In the left side of Table 5.1 LBP, some hydrostatic characteristics, and the wave resistance coefficient of the hull form are shown as they are given in the output file, and on the right side friction resistance coefficient (calculated with ITTC-1957 formula) and the total resistance values are presented. The total resistance is calculated in order to be used as a reference value for the optimization process.

In the end, it was concluded that the integration of PHull with the panel method was successful.

**Panel Method Outputs and Total Resistance Calculation Results  
for Integration Test Case**

<b>LBP (m)</b>	124.04	<b>Cf (ITTC-1957)</b>	0.0014582
<b>Area at WL (m<sup>2</sup>)</b>	1236.383	<b>RT (N)</b>	462544.83
<b>LCF (m)</b>	7.893		
<b>Disp Volume (m<sup>3</sup>)</b>	3245.366		
<b>LCB (m)</b>	2.734		
<b>ZCB (m)</b>	-1.637		
<b>Wetted Surf Area (m<sup>2</sup>)</b>	1651.366		
<b>V(m/s)</b>	13.886		
<b>Cw</b>	0.001376		

Table 5.1: Panel Method Outputs and Total Resistance Calculation Results for Integration Test Case

## 5.2 Integration with the Optimization Algorithm

As the optimization method Differential Evolution was chosen, since this approach was successfully applied to hydrodynamic optimization of hull forms by Brizzolara [4].

Differential Evolution (DE) is a population based optimizer like the Genetic Algorithms, first it generates a random population and evaluates them according to the objective function and each individual in the population is represented as a vector. Then DE generates a new population by perturbing the existing one in order to evaluate and compare the individuals from both populations. Here DE performs a pairwise comparison using the index numbers and select each of the better individuals to form the new population. This process continues until either the required value or the given maximum number of iterations is reached [24].

DE has a wide spectrum of applications in many different engineering fields in order to deal with complex problems. Some of these applications and the DE programs for many different languages are freely available in the Differential Evolution website [25]. Scilab programming language version of DE from this website was modified by Brizzolara for automatic hydrodynamic optimization of hull forms [4].

A parametric representation of the hull form is essential for the optimization with DE. In the optimization procedure firstly a number of parameters are chosen

as the free variables, and a range of values are determined for each of them. Then the optimization algorithm randomly assigns a value for the free variables from the determined range and enables PHull to generate the specified number of different hull representations. After that the panel method is called and all of these individuals are evaluated, and the optimization algorithm works its way to the optimum solution as discussed above.

However running an optimization procedure in serial takes quite a long time for hull form optimization case, since the calculation of the objective function requires the use of the CFD tools. One of the purposes of this thesis was to modify the existing code in order to use it in parallel processing. This was achieved with the Scilab programming language's recently introduced *parallel\_run* function.

In DE objective function is evaluated in two different parts of the code. First is the initialization part, which occurs only once. In this part the individuals of the first generation, which are randomly generated, are evaluated to find the member with the best objective function value. Second part is the principal iteration of the optimization process. This second part occurs until either value to reach is achieved or the maximum number of iterations is reached, and selects which new individuals are allowed to enter the new population by evaluating them. In the new version of the DE Scilab code both of these parts are parallelized by utilizing the *parallel\_run* function.

Eventually PHull parametric modeling tool, the panel method, and the DE optimization algorithm were integrated through Scilab programming language. As presented in Figure 5-2, in order to start the procedure the input csv files of PHull for the first generation are generated with free, fixed and dependent variables as the first step. After this PHull is called in parallel for the generation of the hull representations. Right after the generation of the hull representations panel method is called in parallel for their hydrodynamic evaluation. As the final step the objective function values of the last generation is investigated by the code to see if the required value is reached and if not the input files for the next generation is generated. This process continues until the specified maximum number of iterations is reached.

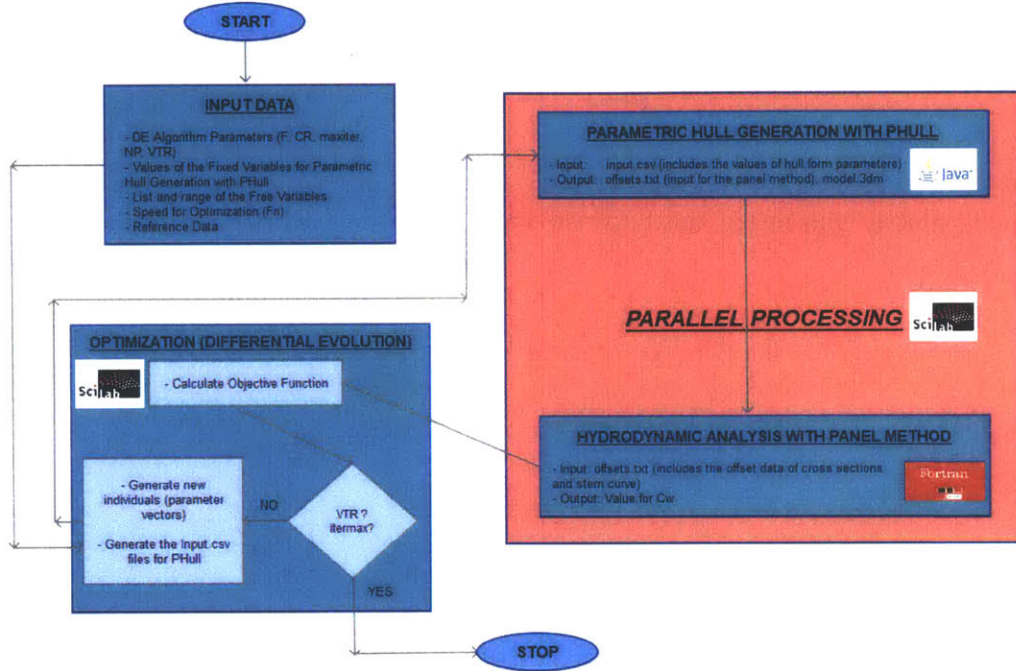


Figure 5-2: General Outline of the Automatic Optimization Process with PHull

### 5.3 Hydrodynamic Optimization of FFG-7 Hull

As the final step of the thesis all the work that has been done so far was tested with a hydrodynamic optimization case of the FFG-7 hull. In this optimization objective was to minimize the ratio of the Total Resistance / Total Displacement Volume of the new hull and the original (reference) hull for a single speed. 13 parameters were selected as free parameters, maximum number of iterations was stated as 40, and as the number of individuals per generation 10 times the number of free parameters, 130, was chosen following the directions of Price et al. [24]. The free parameters that were selected for this optimization case are as follows:

- Keel Rise Point
- Keel Angle at Transom
- DWL Forward Entrance Angle

- DWL Aft Entrance Angle
- Waterplane Coefficient for Forward Part
- Waterplane Coefficient for Aft Part
- Displacement Volume of the Forward Part (the area under the forward part of the SAC)
- Deadrise Angle for the Section at Stem Rise
- Deadrise Angle for the Section at Aft Perpendicular
- Deadrise Angle Curve Forward Entrance Angle
- Deadrise Angle Curve Aft Entrance Angle
- Deadrise Angle for the Section at 0.45 LBP
- Deadrise Angle for the Section at 0.7 LBP

It is necessary to give some more information about the usage of displacement volume of the forward part as a free parameter. In this optimization the total displacement volume of the ship was accepted as a fixed variable and the value of the displacement volume of the aft part parameter as a dependent variable. So the displacement volume of the aft part was simply calculated by subtracting the forward part's value from the total displacement volume.

Furthermore, in the first a few trials it was seen that PHull was unsuccessful at the generation of the sectional area curve for most of the individuals. And the reason of this was found to be the interdependence of the parameters that are used to generate the sectional area curve. Following this reasoning the parameters listed below were defined as dependent on the displacement volume of the forward part parameter. The values of these parameters were inferred with simple linear regression analysis.

- SAC Forward Entrance Angle
- SAC Aft Entrance Angle

- LCB Forward Part (longitudinal position of the centroid of the area under the forward part of SAC)
- LCB Aft Part (longitudinal position of the centroid of the area under the aft part of SAC)
- The variable that determines the longitudinal position of the top tangency control vertex of the forward part of SAC
- The variable that determines the longitudinal position of the top tangency control vertex of the aft part of SAC

When the optimization case was completed objective function value of each hull was plotted according to the case ID number of the hull. This plot is presented in Figure 5-3. In this plot the hull with the minimum objective function value, the optimum hull, is shown with the red dot. The individuals that got a penalty since their hull forms could not be generated by PHull are excluded. Because the objective function value for these hulls were around 1000 because of the penalties that they were assigned, and it was not appropriate to have them in the same plot with the rest.

For the optimum hull more than 25 percent reduction in the objective function value was achieved. However general form of the plot shows that the optimization didn't converge, since the data points are distributed around a horizontal straight line. This is believed to be related to the way that the sectional area curve is generated, and the dependence of its parameters to each other. Comparison of the the original FFG-7 model and the model of the hull form with the minimum objective function value are presented in the following figures.

From the comparison of the SAC's in Figure 5-4 it was seen that the displacement volume of the forward part of the hull is slightly increased, whereas there is a reduction in the displacement volume of the aft part. Figure 5-6 shows that the waterplane area for the forward and aft parts followed the same pattern with the displacement volumes, it increased in the forward part and decreased in the aft part. Figure 5-7

	<b>Original Model</b>	<b>Optimized Model</b>
<b>Keel Rise Point</b>	0.645678813	0.6950343
<b>Keel Angle @ Transom</b>	6.21	7.9891124
<b>DWL Fwd Angle</b>	11.02	11.157031
<b>DWL Aft Angle</b>	11.78	11.532298
<b>Cwp Aft</b>	0.814	0.7855679
<b>Cwp Fwd</b>	0.613	0.6361764
<b>FwdDisplacement</b>	1615	1642.8029
<b>Deadrise angle for the section at stem rise</b>	47.917	44.806447
<b>Deadrise angle for the section at AP</b>	1.312	0.7123125
<b>DrAngCrv FwdEntAng</b>	13.138	12.128625
<b>DrAngCrv AftEntAng</b>	67.111	66.847757
<b>Deadrise angle for the section at 0.45*LBP</b>	10.47	11.286263
<b>Deadrise angle for the section at 0.7*LBP</b>	19.083	20.109864

Table 5.2: Comparison of the Set of Free Parameters for the Original and the Optimum Hulls



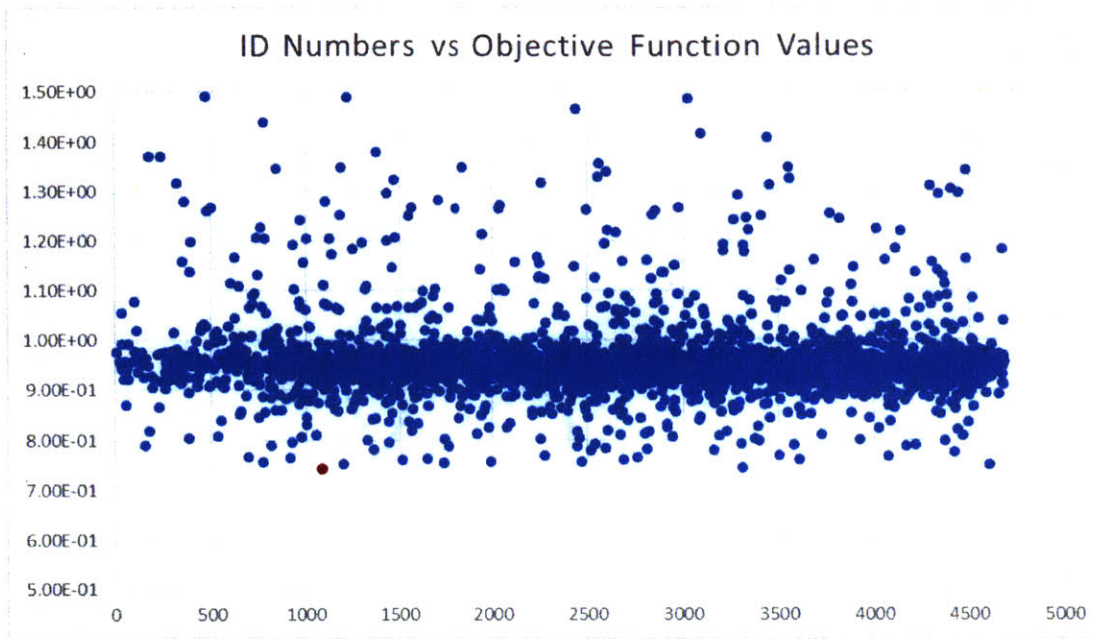


Figure 5-3: Plot of the Objective Function Values of the Hulls That Were Evaluated in the Optimization

shows that the keel rise point's longitudinal position got closer to AP for the optimum case. Finally the comparison of the cross section curves of these two hulls can be seen from Figure 5-8.

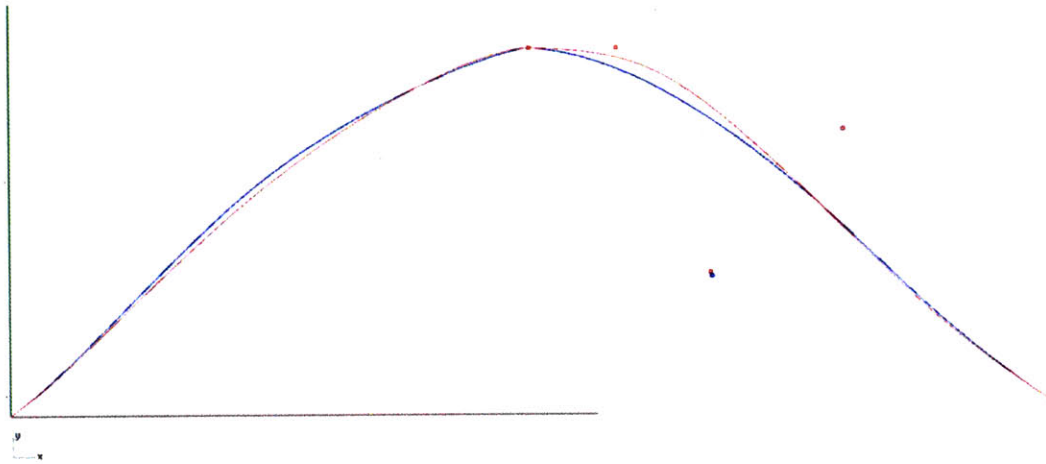


Figure 5-4: Comparison of the Sectional Area Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue) (note that forward part is to the left)

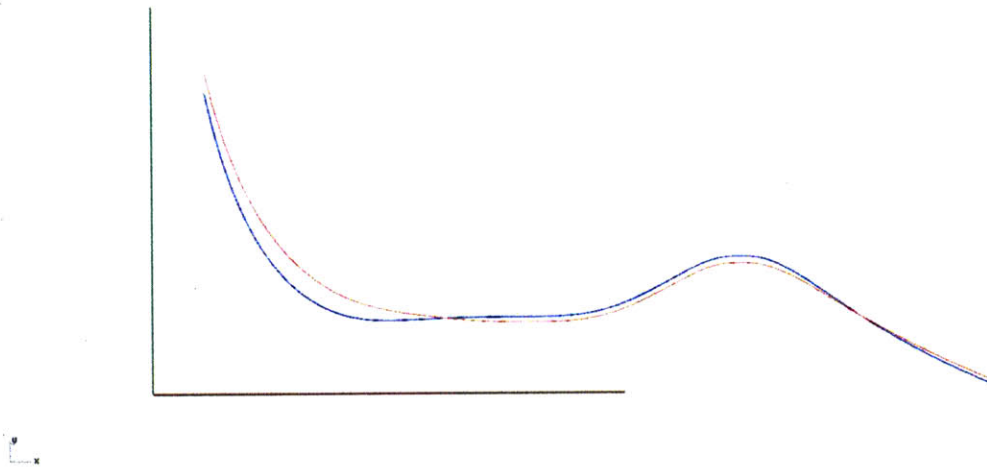


Figure 5-5: Comparison of the Deadrise Angle Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue)

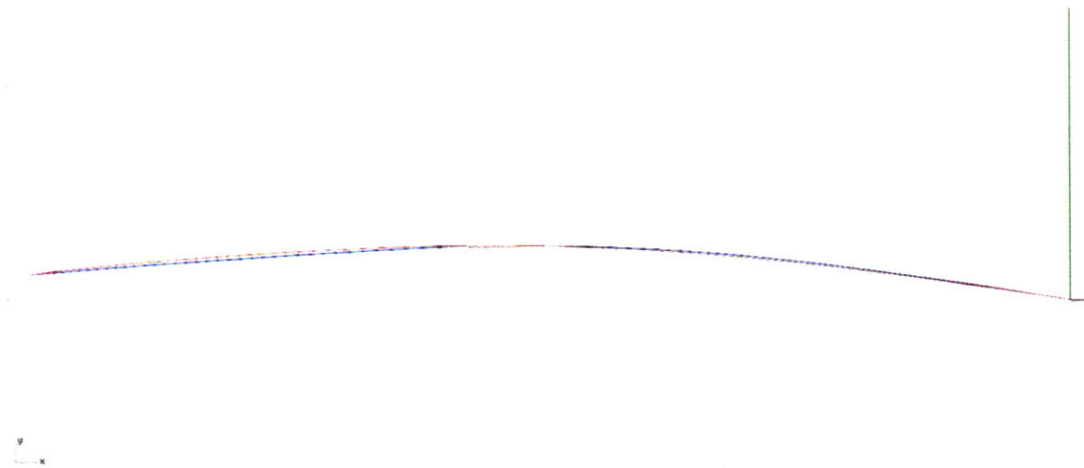


Figure 5-6: Comparison of the DWL Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue)

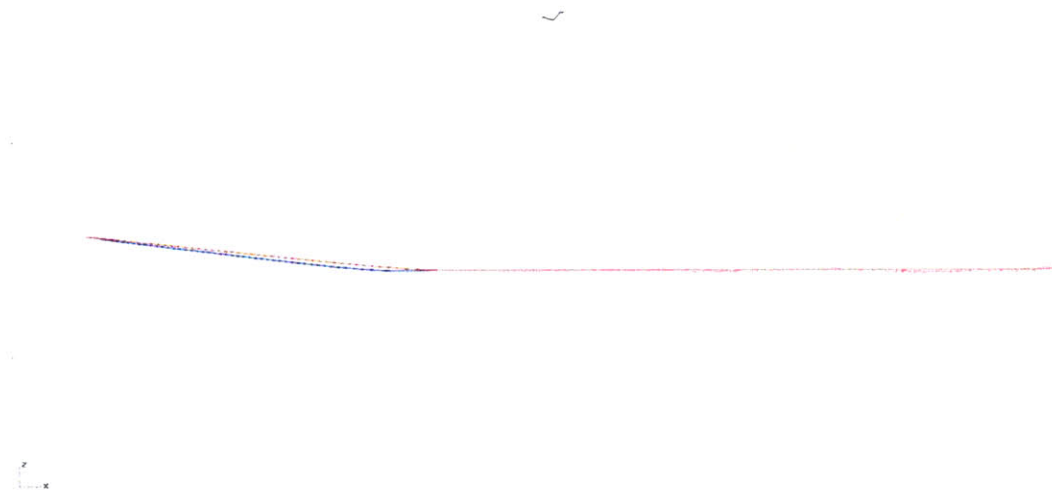


Figure 5-7: Comparison of the Profile Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue)

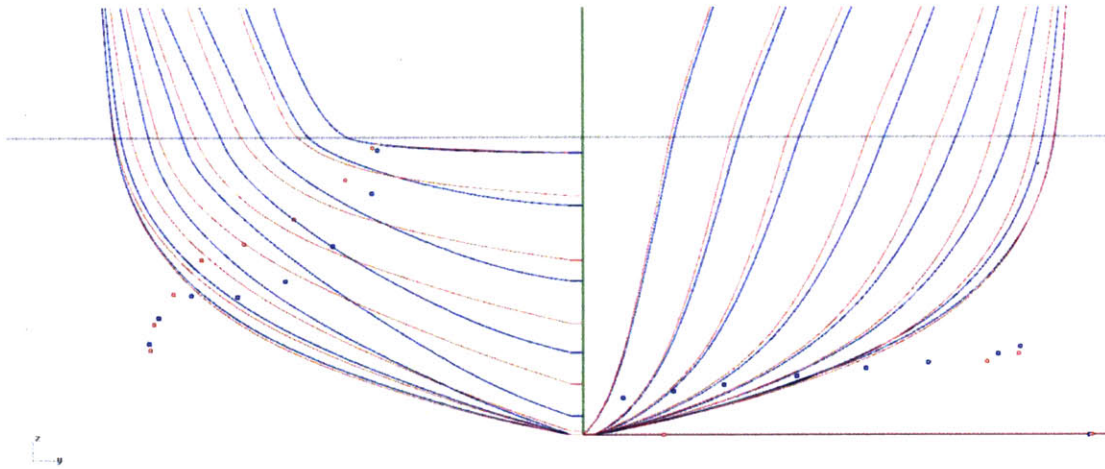


Figure 5-8: Comparison of the Cross Section Curves of the Original FFG-7 Model (red) and the Optimum Hull (blue)

# Chapter 6

## Conclusions and Recommendations

### 6.1 Conclusions

This thesis developed a fully parametric modeling tool, PHull, for the design and hydrodynamic optimization of high speed displacement monohulls. This tool is to be used in the concept design phase of the ship, and its goal is to enable the designer to make more accurate decisions in this early stage.

A fully parametric hull form definition, which of course can be obtained by means of a fully parametric modeling tool, comes up with many advantages:

- Definition and control of parameters along the hull length is easier with a fully parametric hull form definition.
- Instead of using the random positions of the control vertices, hull surfaces can be controlled with more meaningful and intuitive geometric parameters.
- By controlling different sets of parameters both local and global modifications on the hull form can be achieved.
- Integration with CFD and optimization tools are easier.
- Variants of an original hull can be easily obtained by modifying parameters, and by means of a systematic modification of the parameters a family of the original hull can be easily obtained.

All these advantages are inherent to PHull, and this was verified with the example cases of FFG-7 and ATHENA hull forms.

Furthermore the hydrodynamic optimization case for the FFG-7 hull showed that PHull's integration with a CFD tool and an optimization method was accomplished successfully, and resulted in a hull form with a significantly reduced total resistance.

## 6.2 Recommendations for Future Work

PHull is a new tool with a number of shortcomings that is known to the author, and these issues should be addressed as the next step in the development of this tool.

First of all PHull doesn't treat the above water part of the hull as elaborately as the underwater part, since it was primarily developed for hydrodynamic optimization of the hulls. But during the optimization case of the FFG-7, discontinuities between the above water and the underwater parts of the cross section curves were discovered as a consequence of keeping the parameters that are used in the generation of the edge of the weather deck curve fixed. This problem can be addressed by defining these parameters as dependent on some underwater hull parameters.

Secondly in the optimization case it was seen that the technique that is used by PHull for the generation of the sectional area curve was not really flexible. So in order to make the optimization case work all parameters, except the displacement volume, that are used to define the SAC were defined as dependent variables, and this fact limited the ability of PHull to obtain significantly different SAC forms.

Finally, in its current version PHull only takes LCB parameter into consideration as a hydrostatic parameter. As a future work other hydrostatic parameters like the metacentric height can be integrated into PHull.

Of course more shortcomings will be discovered that are not known to the author yet as PHull will be used for other cases. These areas with shortcomings will be improved with the user feedback.

# Appendix A

## User Guide for PHull

In this part how to use PHull will be discussed with the assumption that PHull will be used for hydrodynamic hull form optimization of an existing model and not for *ab initio* design. Generally the content of Chapter 3 will be followed and the insight that the developer obtained throughout the example cases of FFG-7 and Athena hull forms will be provided to the reader. Some of the same figures and tables from Chapter 3 will be used in this part as well, just to assure the integrity of the user guide.

The general outline of the user guide will be as follows:

- Preparation of the input file
- Representation of the hull form using the interpolated and approximated control curves together
- Reproducing the hull form using only approximated control curves

### A.1 Preparation of the Input File

Input file is a csv (comma separated value) file that contains all the necessary data for PHull to generate a fully parametric hull form. As the first step of the hull generation this csv file has to be prepared by the user. In this section the structure of the csv file will be introduced first, and then the explanations for the required parameters will be presented.

The input file for the generation of the original FFG-7 hull is displayed in Table A.1. Parameters in this part are grouped together according to their functions, and different groups are shown with different colors.

In this table, data related to the principal dimensions, hull form coefficients, and the important points, e.g. stem rise point and keel rise point, are defined in the blue section of the spreadsheet. In the yellow section position, slope, and area data about the cross sections are defined that are used for the generation of the control curves by interpolation. Whereas the orange section contains the data for the generation of the control curves by approximation. Finally in the green section some parameters are defined for the systematic generation by PHull. In the following part these different sections will be described.

The input file's blue section has to be filled for any type of hull form generation, since it contains the most essential data that contains the principal dimensions. Parameter list of this section is presented in Table A.2 with the explanation of the parameters.

Yellow section has to be filled when the control curves except the DWL, profile, and edge of the weather deck curves are to be generated with interpolation. This is most likely the case for the shape representation. Parameter list of this section is presented in Table A.3. As you can see from Table A.3, data from 21 cross sections of the hull for sectional area, deadrise angle, flare angle, keel width, and flare angle at weather deck have to be provided in this section. If all the control curves are to be generated with approximation method, this section can be left blank.

Orange section has to be filled when all the control curves will be generated with approximation methods. Parameter list of this section is presented in Table A.4 with the explanation of the parameters. This section can be left blank in the cases where yellow section of the input file is used.

Green section is not as important as the rest of the input file, and should be left blank for most of the time. This section is only used for automatic systematic hull form generation. Parameter list of this section is presented in Table A.5 with the explanation of the parameters.



LBP	124.04	DeadriseAngle1	47.917	DeadriseAngle1	10.596	DeadriseAngle21	1.312
L/B	9.107195	DeadriseAngle2	38.21	DeadriseAngle2	11.788		
T	4.38	DeadriseAngle3	28.512	DeadriseAngle3	13.898		
Top Stem Angle	46.12	DeadriseAngle4	21.332	DeadriseAngle4	17.942		
Bottom Stem Angle	1.5	DeadriseAngle5	15.298	DeadriseAngle5	18.993		
Stem Corner Point Weight	0.85	DeadriseAngle6	13.602	DeadriseAngle6	17.481		
StemRadius/T	1.783105	DeadriseAngle7	12.998	DeadriseAngle7	15.093		
Keel Rise Point	0.645679	DeadriseAngle8	12.381	DeadriseAngle8	11.166		
Transom Depth	0.23	DeadriseAngle9	11.447	DeadriseAngle9	7.297		
Keel Angle @ Transom	6.21	DeadriseAngle10	10.7	DeadriseAngle20	3.642		
Transom Angle	50.94	FlareAngle1	13.175	FlareAngle11	9.366	FlareAngle21	71.945
Beam@Transom/MaxBeam	0.502203	FlareAngle2	15.65	FlareAngle12	9.592		
Max Beam X Location	0.554418	FlareAngle3	15.695	FlareAngle13	11.475		
DWL Fwd Angle	11.02	FlareAngle4	16.38	FlareAngle14	16.908		
DWL Aft Angle	11.78	FlareAngle5	17.766	FlareAngle15	21.949		
Cwp Aft	0.814	FlareAngle6	21.003	FlareAngle16	23.63		
Cwp Fwd	0.613	FlareAngle7	20.884	FlareAngle17	24.19		
DWL Fwd Wt Pts ShiftX	-3.25	FlareAngle8	17.638	FlareAngle18	28.575		
DWL Fwd Wt Pts ShiftY	-0.19	FlareAngle9	14.329	FlareAngle19	35.622		
DWL Aft Wt Pts ShiftX	3	FlareAngle10	11.7	FlareAngle20	45.182		
DWL Aft Wt Pts ShiftY	-0.1	SectionalArea1	3.298	SectionalArea11	22.192	SectionalArea21	0.6053
xProfileEndToTransom	0.15	SectionalArea2	5.172	SectionalArea12	22.01		
fbAft	5.147	SectionalArea3	7.01	SectionalArea13	21.224		
fbFwd	8.37	SectionalArea4	8.898	SectionalArea14	19.822		
fbLow	4.505	SectionalArea5	11.752	SectionalArea15	17.484		
fbFwdXPos	-8.537	SectionalArea6	14.212	SectionalArea16	14.532		
fbLowXPos	88.784	SectionalArea7	16.466	SectionalArea17	11.413		
fbAftAng	1.623	SectionalArea8	18.51	SectionalArea18	8.312		
fbFwdAng	5.7	SectionalArea9	20.253	SectionalArea19	5.437		
weatherDeckFwdAng	16.231	SectionalArea10	21.507	SectionalArea20	2.82		
weatherDeckAftAng	6.478	KeelWidth1	0	KeelWidth11	0.176	KeelWidth21	0.176
fwdDisplacement	1615	KeelWidth2	0.148	KeelWidth12	0.176		
areaAtTransom	1.16	KeelWidth3	0.164	KeelWidth13	0.176		
fwdLcb	40.29	KeelWidth4	0.176	KeelWidth14	0.176		
sacEntranceAngFwd	35.964	KeelWidth5	0.176	KeelWidth15	0.176		
sacEntranceAngStem	34.767	KeelWidth6	0.176	KeelWidth16	0.176		
areaAtMaxBeam	44.389	KeelWidth7	0.176	KeelWidth17	0.176		
tolLCB	0.1	KeelWidth8	0.176	KeelWidth18	0.176		
tolDisp	1	KeelWidth9	0.176	KeelWidth19	0.176		
maxSecAreaXPos	0.5	KeelWidth10	0.176	KeelWidth20	0.176		
aftDisplacement	1660	WDFlareAng1	22.309	WDFlareAng11	3.025	WDFlareAng21	13.025
aftLcb	83.522	WDFlareAng2	24.913	WDFlareAng12	1.937		
Displacement (for design)	3295.73	WDFlareAng3	25.688	WDFlareAng13	1.018		
LCB (for design)	62.237	WDFlareAng4	25.948	WDFlareAng14	1.278		
1st CrossSec drAng	47.917	WDFlareAng5	26.007	WDFlareAng15	2.734		
Last CrossSec drAng	1.312	WDFlareAng6	24.823	WDFlareAng16	5.597		
drAngCrv fwdEntAng	13.138	WDFlareAng7	21.295	WDFlareAng17	9.794		
drAngCrv aftEntAng	67.111	WDFlareAng8	16.099	WDFlareAng18	14.254		
drAng at 0.45	10.47	WDFlareAng9	10.488	WDFlareAng19	17.326		
drAng at 0.7	19.083	WDFlareAng10	5.703	WDFlareAng20	15.23		
1st CrossSec flAng	13.175	# of Outputs	1				
Last CrossSec flAng	71.945	Increase Fwd Fullness	0				
flAngCrv fwdEntAng	43.83	Decrease Fwd Fullness	0				
flAngCrv aftEntAng	6.835	Increase Aft Fullness	0				
flAng at 0.275	21.915	Decrease Aft Fullness	1				
flAng at 0.525	9.117	Increase Fullness	0				
1st CrossSec wdAng	26.101	Decrease Fullness	0				
Last CrossSec wdAng	17.324						
wdAngCrv fwdEntAng	90						
wdAngCrv aftEntAng	90						
wdAng at 0.625	1						
1st CrossSec kw	0						
Last CrossSec kw	0.176						
kwCrv fwdEntAng	5.91						
kwCrv aftEntAng	90						
kw straightLinePt xPos	0.15						
xMultiplierFwd	1						
xMultiplierAft	1						
intermediateDrAngPos1	0.45						
intermediateDrAngPos2	0.7						
intermediateFlAngPos1	0.275						
intermediateFlAngPos2	0.525						
intermediateWdAngPos	0.625						
weightPtYPos@0.4LBP	5.333						
weightPtYPos@0.4SLBP	5.865						
weightPtYPosDenominatorForStemRise	24						
weightPtZPos@0.4LBP	1.08						
weightPtZPos@0.4SLBP	1.108						
weightPtZPosDenominatorForStemRise	8						
transomHalfbeamMultiplier	1.57						
maxHalfbeamMultiplier	1.04						

Table A.1: Complete Set of Parameters for PHull

<b>LBP</b>	124.04	length between perpendiculars
<b>L/B</b>	9.107195	lbp - max beam ratio
<b>T</b>	4.38	draft
<b>Top Stem Angle</b>	46.12	entrance angle of the underwater stem curve at DWL
<b>Bottom Stem Angle</b>	1.5	entrance angle of the underwater stem curve at stem rise point
<b>Stem Corner Point Weight</b>	0.85	controls the fullness of the underwater stem curve
<b>StemRadius/T</b>	1.783105	determines the position of the stem rise point (if 1, stem rise point's distance from the origin will be the same as draft)
<b>Keel Rise Point</b>	0.645679	determines the position of the keel rise point as a fraction of lbp
<b>Transom Depth</b>	0.23	height of the submerged part of transom
<b>Keel Angle @ Transom</b>	6.21	entrance angle of the keel at transom
<b>Transom Angle</b>	50.94	not being used
<b>Beam@Transom/MaxBeam</b>	0.502203	ratio of beam at transom and the max beam
<b>Max Beam X Location</b>	0.554418	longitudinal position of the section with max beam
<b>DWL Fwd Angle</b>	11.02	entrance angle of the DWL curve in the fwd part
<b>DWL Aft Angle</b>	11.78	entrance angle of the DWL curve in the aft part
<b>Cwp Aft</b>	0.814	waterplane coefficient for the aft part
<b>Cwp Fwd</b>	0.613	waterplane coefficient for the fwd part
<b>DWLFwdWtPtShiftX</b>	-3.25	manipulates the longitudinal position of the intermediate control vertex of Fwd DWL curve
<b>DWLFwdWtPtShiftY</b>	-0.19	manipulates the transversal position of the intermediate control vertex of Fwd DWL curve
<b>DWLAftWtPtShiftX</b>	3	manipulates the longitudinal position of the intermediate control vertex of Aft DWL curve
<b>DWLAftWtPtShiftY</b>	-0.1	manipulates the transversal position of the intermediate control vertex of Aft DWL curve
<b>xProfileEndToTransom</b>	0.15	not being used
<b>fbAft</b>	5.147	above water part
<b>fbFwd</b>	8.37	above water part
<b>fbLow</b>	4.505	above water part
<b>fbFwdXPos</b>	-8.537	above water part
<b>fbLowXPos</b>	88.784	above water part
<b>fbAftAng</b>	1.623	above water part
<b>fbFwdAng</b>	5.7	above water part
<b>weatherDeckFwdAng</b>	16.231	above water part
<b>weatherDeckAftAng</b>	6.476	above water part

Table A.2: Parameters of Blue Section with Explanations

DeadriseAngle1	47.917	DeadriseAngle11	10.596	DeadriseAngle21	1.312
DeadriseAngle2	38.21	DeadriseAngle12	11.788		
DeadriseAngle3	28.512	DeadriseAngle13	13.898		
DeadriseAngle4	21.332	DeadriseAngle14	17.942		
DeadriseAngle5	15.298	DeadriseAngle15	18.993		
DeadriseAngle6	13.602	DeadriseAngle16	17.481		
DeadriseAngle7	12.998	DeadriseAngle17	15.093		
DeadriseAngle8	12.381	DeadriseAngle18	11.166		
DeadriseAngle9	11.447	DeadriseAngle19	7.297		
DeadriseAngle10	10.7	DeadriseAngle20	3.642		
FlareAngle1	13.175	FlareAngle11	9.366	FlareAngle21	71.945
FlareAngle2	15.65	FlareAngle12	9.592		
FlareAngle3	15.695	FlareAngle13	11.475		
FlareAngle4	16.38	FlareAngle14	16.908		
FlareAngle5	17.766	FlareAngle15	21.949		
FlareAngle6	21.003	FlareAngle16	23.63		
FlareAngle7	20.884	FlareAngle17	24.19		
FlareAngle8	17.638	FlareAngle18	28.575		
FlareAngle9	14.329	FlareAngle19	35.622		
FlareAngle10	11.7	FlareAngle20	45.182		
SectionalArea1	3.298	SectionalArea11	22.192	SectionalArea21	0.6053
SectionalArea2	5.172	SectionalArea12	22.01		
SectionalArea3	7.01	SectionalArea13	21.224		
SectionalArea4	8.898	SectionalArea14	19.822		
SectionalArea5	11.752	SectionalArea15	17.484		
SectionalArea6	14.212	SectionalArea16	14.532		
SectionalArea7	16.466	SectionalArea17	11.413		
SectionalArea8	18.51	SectionalArea18	8.312		
SectionalArea9	20.253	SectionalArea19	5.437		
SectionalArea10	21.507	SectionalArea20	2.82		
KeelWidth1	0	KeelWidth11	0.176	KeelWidth21	0.176
KeelWidth2	0.148	KeelWidth12	0.176		
KeelWidth3	0.164	KeelWidth13	0.176		
KeelWidth4	0.176	KeelWidth14	0.176		
KeelWidth5	0.176	KeelWidth15	0.176		
KeelWidth6	0.176	KeelWidth16	0.176		
KeelWidth7	0.176	KeelWidth17	0.176		
KeelWidth8	0.176	KeelWidth18	0.176		
KeelWidth9	0.176	KeelWidth19	0.176		
KeelWidth10	0.176	KeelWidth20	0.176		
WDFlareAng1	22.309	WDFlareAng11	3.025	WDFlareAng21	13.025
WDFlareAng2	24.913	WDFlareAng12	1.937		
WDFlareAng3	25.688	WDFlareAng13	1.018		
WDFlareAng4	25.948	WDFlareAng14	1.278		
WDFlareAng5	26.007	WDFlareAng15	2.734		
WDFlareAng6	24.823	WDFlareAng16	5.597		
WDFlareAng7	21.295	WDFlareAng17	9.794		
WDFlareAng8	16.099	WDFlareAng18	14.254		
WDFlareAng9	10.488	WDFlareAng19	17.326		
WDFlareAng10	5.703	WDFlareAng20	15.23		

Table A.3: Parameters of Yellow Section

<b>fwdDisplacement</b>	1615	area under the forward part of SAC
<b>areaAtTransom</b>	1.16	sectional area of the last cross section at AP (of complete cross section not half)
<b>fwdLcb</b>	40.29	longitudinal centroid of the area under the forward part of SAC
<b>sacEntranceAngFwd</b>	35.964	entrance angle of the SAC in the forward part
<b>sacEntranceAngStern</b>	34.767	entrance angle of the SAC in the aft part
<b>areaAtMaxBeam</b>	44.389	sectional area of the section with max area (of complete cross section not half)
<b>tolLCB</b>	0.1	tolerance for the accuracy of LCB calculations
<b>tolDisp</b>	1	tolerance for the accuracy of displacement volume calculations
<b>maxSecAreaXPos</b>	0.5	longitudinal position of the section with max area
<b>aftDisplacement</b>	1660	area under the aft part of SAC
<b>aftLcb</b>	83.522	longitudinal centroid of the area under the aft part of SAC
<b>Displacement (for design)</b>	3295.73	not being used
<b>LCB (for design)</b>	62.237	not being used
<b>1st CrossSec drAng</b>	47.917	deadrise angle for the cross section at the stem rise point
<b>Last CrossSec drAng</b>	1.312	deadrise angle for the cross section at AP
<b>drAngCrv fwdEntAng</b>	13.138	entrance angle of the deadrise curve in the forward part
<b>drAngCrv aftEntAng</b>	67.111	entrance angle of the deadrise curve in the aft part
<b>drAng at 0.45</b>	10.47	deadrise angle for the cross section at 0.45° LBP
<b>drAng at 0.7</b>	19.083	deadrise angle for the cross section at 0.70° LBP
<b>1st CrossSec flAng</b>	13.175	flare angle for the cross section at the stem rise point
<b>Last CrossSec flAng</b>	71.945	flare angle for the cross section at AP
<b>flAngCrv fwdEntAng</b>	43.83	entrance angle of the flare angle curve in the forward part
<b>flAngCrv aftEntAng</b>	6.835	entrance angle of the flare angle curve in the aft part
<b>flAng at 0.275</b>	21.915	flare angle for the cross section at 0.275° LBP
<b>flAng at 0.525</b>	9.117	flare angle for the cross section at 0.525° LBP
<b>1st CrossSec wdAng</b>	26.101	weather deck
<b>Last CrossSec wdAng</b>	17.324	weather deck
<b>wdAngCrv fwdEntAng</b>	90	weather deck
<b>wdAngCrv aftEntAng</b>	90	weather deck
<b>wdAng at 0.625</b>	1	weather deck
<b>1st CrossSec kw</b>	0	keelWidth
<b>Last CrossSec kw</b>	0.176	keelWidth
<b>kwCrv fwdEntAng</b>	5.91	keelWidth
<b>kwCrv aftEntAng</b>	90	keelWidth
<b>kw straightLinePt xPos</b>	0.15	keelWidth
<b>xMultiplierFwd</b>	1	controls the longitudinal position of the top tangency constraint control point of fwd part of the SAC
<b>xMultiplierAft</b>	1	controls the longitudinal position of the top tangency constraint control point of aft part of the SAC
<b>intermediateDrAngPos1</b>	0.45	Longitudinal position for the cross section, from which the first intermediate deadrise angle data is given
<b>intermediateDrAngPos2</b>	0.7	Longitudinal position for the cross section, from which the second intermediate deadrise angle data is given
<b>intermediateFlAngPos1</b>	0.275	Longitudinal position for the cross section, from which the first intermediate flare angle data is given
<b>intermediateFlAngPos2</b>	0.525	Longitudinal position for the cross section, from which the first intermediate flare angle data is given
<b>intermediateWdAngPos</b>	0.625	Longitudinal position for the cross section, from which the intermediate flare angle at weather deck data is given
<b>weightPTYPos@0.4LBP</b>	5.333	Transversal position for the weight point of the cross section at 0.4° LBP. Used for cubic spline interpolation.
<b>weightPTYPos@0.45LBP</b>	5.865	Transversal position for the weight point of the cross section at 0.45° LBP. Used for cubic spline interpolation.
<b>weightPTYPosDenominatorForStemRise</b>	24	This number is used to divide the max beam. The result is used as the transversal position of the weight point of the section at stemRise
<b>weightPtZPos@0.4LBP</b>	1.08	Vertical position for the weight point of the cross section at 0.4° LBP. Used for cubic spline interpolation.
<b>weightPtZPos@0.45LBP</b>	1.108	Vertical position for the weight point of the cross section at 0.45° LBP. Used for cubic spline interpolation.
<b>weightPtZPosDenominatorForStemRise</b>	8	This number is used to divide the draft. The result is used as the transversal position of the weight point of the section at stemRise
<b>transomHalfbeamMultiplier</b>	1.57	Multiplies the beam at transom on WL to obtain beam at transom on weather deck level
<b>maxHalfbeamMultiplier</b>	1.04	Multiplies the max beam on WL to obtain max beam at on weather deck level

Table A.4: Parameters of Orange Section with Explanations

# of Outputs	1
Increase Fwd Fullness	0
Decrease Fwd Fullness	0
Increase Aft Fullness	0
Decrease Aft Fullness	1
Increase Fullness	0
Decrease Fullness	0

Table A.5: Parameters of Green Section

## A.2 Representation of an Existing Hull Using the Interpolated and Approximated Control Curves Together

For hydrodynamic optimization of an existing hull, first thing that has to be done is to represent the model in PHull. In order to do this blue and yellow sections of the input file has to be filled with the values that are read from the real model. Profile, DWL, and edge of the weather deck curves will be generated with the data from the blue section of the input file, whereas the rest of the control curves will be generated with the data from the yellow section.

Accurate generation of the profile, DWL, and edge of the weather deck curves are essential for a successful hull representation with PHull. And all of these curves are represented as either B-spline or NURBS curves, which are the approximation techniques. So achieving the best result requires the use of trial and error method. As described in Table A.2, some of the parameters lets the user to manipulate the form of these curves locally in order to achieve better representations. DWLFwdWtPTShift and the Stem Corner Point Weight parameters are a few examples for that.

Filling in the yellow part of the input file enables PHull to generate the sectional area, deadrise angle, flare angle at DWL, flare angle at weather deck, and the keel width control curves with an interpolation method, which uses a reverse B-spline algorithm to fit the curve to data points. This method is obtained from MATLAB Central, and is called SPLINEFIT [20]. By filling in the yellow part the data from 21 cross sections is passed to PHull as discussed in the previous section, then PHull

uses this data as the data points for SPLINEFIT method. The use of this method for shape representation is encouraged, since most of the time more accurate curves are obtained by employing interpolation methods.

In the end of this part the user has to make sure that he/she obtained an accurate hull form representation. If this is the case, the control curves obtained during this procedure will be either directly used, or serve as a reference for the next step, reproducing the hull form using only approximated control curves.

### **A.3 Reproducing the Hull Form Using Only Approximated Control Curves**

Even though the hull form generation technique that was discussed in the previous section is great for hull form representation, it's not really good for the optimization procedure. First reason for that is the interpolation techniques don't give much flexibility for the modifications of the hull form, which is crucial in hydrodynamic optimization. And second reason is each control curve that is produced with cubic spline fitting requires 21 input parameters. This is bad because the number of parameters that is necessary for the hull modification will be higher than the desired number of parameters to optimize the hull on.

So it is recommended to represent the sectional area, deadrise angle, flare angle at DWL, flare angle at weather deck, and the keel width control curves with approximation methods, namely the B-splines and NURBS. This can be achieved by filling in the orange part of the input file instead of the yellow part. The parameters in this part, which is presented in Table A.4, are used to define the control vertices of these curves. The best way to obtain this data is to use the resulting control curves from the previous section. The definition of the parameters are presented in Table A.4.

In the end of this part resulting hull model should be compared to the original model. The accurate representation of the hull form will again require some trial and error. When an accurate representation is obtained, the set of input parameters are

ready for the hydrodynamic optimization process.





# Bibliography

- [1] Abt, C.; Bade, S.D.; Birk, L., Harries, S.: *Parametric Hull Form Design - A Step Towards One Week Ship Design*, 8th International Symposium on Practical Design of Ships and Other Floating Structures, PRADS 2001, Shanghai, 2001.
- [2] Harries, S.: *Parametric Design and Hydrodynamic Optimization of Ship Hull Forms*, PhD Thesis, Mensch and Buch Verlag, 1998
- [3] Nowacki, H.; Bloor, M.I.G.; Oleksiewicz, B.: *Computational Geometry for Ships*, World Scientific Publishing, 1995
- [4] Brizzolara, S.: *Parametric Optimization of SWAT-Hull Forms by a Viscous-Inviscid Free Surface Method Driven by a Differential Evolution Algorithm*, 25th Symposium on Naval Hydrodynamics, St. John's, Newfoundland and Labrador, Canada, 8-13 August 2004
- [5] Brizzolara, S.; Vernengo, G.: *Automatic Optimization Computational Method for Unconventional S.W.A.T.H. Ships Resistance*, International Journal of Mathematical Models and Methods in Applied Sciences, 2011
- [6] Vernengo, G.; Biliotti, I; Brizzolara, S; Viviani, M; Ruscelli, D; Bonvicini, A; Galliussi, M; Manfredini, A: *Influence of Form Parameters Selection on the Hull Surface Shape Control for Hydrodynamic Design*, 16th International Conference of Ship and Shipping Research, Messina, Italy, 25-27 November 2009

- [7] Vernengo, G.; Brizzolara, S; Bruzzone, D: *Resistance and Seakeeping Optimization of a Fast Multihull Passenger Ferry*, International Journal of Offshore and Polar Engineering (ISSN 1053-5381), Vol. 25, No. 1, pp. 26–34 March 2015
- [8] Fuller, A.L.; Aughey, M.E.; Billingsley, D.W.: *Computer-Aided Ship Hull Definition at the Naval Ship Engineering Center*, Computer-Aided Hull Surface Definition Symposium, Annapolis, Md., September 26-27, 1977
- [9] Yang, C.; Bowers, D: *Hydrodynamic Design Optimization Tool*, Technical Report, Center for Innovation in Ship Design, NSWCCD-CISD-2011/014, 2011
- [10] Yang, C.; Kim, H; Kim, H; Chun, H.H: *A Combined Local and Global Hull Form Modification Approach for Hydrodynamic Optimization*, 28th Symposium on Naval Hydrodynamics, Pasadena, California, 12-17 September 2010, 2010
- [11] Harries, S.; Abt, C.: *Parametric Curve Design Applying Fairness Criteria*, Institut für Schiffs- und Meerestechnik, Technische Universität Berlin
- [12] Le, T.H.; Kim, D.J.: *Application of a Real-Coded Genetic Algorithm for the Fitting of a Ship Hull Surface Through A Single Non-Uniform B-spline Surface*, 2009
- [13] Rogers, D.F.; Adams, J.A.: *Mathematical Elements for Computer Graphics*, McGraw-Hill Publishing Company, 1990
- [14] Patrikalakis, N. M.; Maekawa, T; Cho, W: *Shape Interrogation for Computer Aided Design and Manufacturing*, Springer Science and Business Media, 2009
- [15] Nowacki, H.: *Five Decades of Computer-Aided Ship Design*, Technical University of Berlin, 2009
- [16] <http://igeo.jp/>
- [17] <https://processing.org/>
- [18] <https://www.eclipse.org/users/>

- [19] Nestoras, K.: *A Tool to Create Hydrodynamically Optimized Hull-Forms with Geometrical Constraints from Internal Arrangements*, Master of Science Thesis, Massachusetts Institute of Technology, 2013
- [20] <http://www.mathworks.com/matlabcentral/fileexchange/13812-splinefit>
- [21] Jenkins, D. S.: *Resistance Characteristics of the High Speed Transom Stern Ship R/V ATHENA in the Bare Hull Condition, Represented by DTNSRDC Model 5365*, Ship Performance Department Research and Development Report, David W. Taylor Naval Ship Research and Development Center, 1984
- [22] Sivanandam, S. N.; Deepa, S.N.: *Introduction to Genetic Algorithms*, Springer, 2008
- [23] Brizzolara, S.; Bruzzone, D: *Numerical Wave Resistance and Dynamic Trim of High Speed Crafts*, International Conference on Ship and Shipping Research (2000), vol. 1, p. 4.2.1-13, ISBN 88- 900487-0-0
- [24] Price, K.V.; Storn, M.S; Lampinen, J.A.: *Differential Evolution - A Practical Approach to Global Optimization*, Springer, 2008
- [25] <http://www1.icsi.berkeley.edu/~storn/code.html>