ANNALS OF "DUNAREA DE JOS" UNIVERSITY OF GALATI
FASCICLE XI – SHIPBUILDING. ISSN 1221-4620
2015

A PRACTICAL SYSTEM FOR CFD BASED HULL OPTIMISATION

Binoy Pilakkat
University of Liege
Faculty of Applied Sciences – ANAST, Liege
Place du 20 Août, 7 Bât.A1, 4000 Liegè, Belgium
E-mail:binoypilakkat@outlook.com

Florin Păcuraru
"Dunarea de Jos" University of Galati,
Faculty of Naval Architecture, Galati, 47
Domneasca Street, 800008, Romania,
E-mail: florin.pacuraru@ugal.ro

ABSTRACT

Optimisation based on numerical methods has become key tool in the development of environmentally friendly and economically efficient hull forms. The paper is focused on optimization of wave making resistance for a river cruise ferry which will be powered by solar panels. An optimisation framework is developed by the authors considering the practical aspects and the non-linear potential solver which is used. Optimisation is carried out using the open source tool called Dakota which is connected to Rhino and SHIPFLOW using Python programs developed by the author. The advantage of the method is that it is highly customisable and can be easily extended to include any further calculations.

Keywords: CFD, Optimisation, Shipflow, Potential flow, Dakota

1. INTRODUCTION

CFD has become a common tool in ship hull form optimisation. It is possible to analyse a large number of design variations using CFD codes in less time, which is an arduous task using EFD. In this work, an optimisation procedure has been formulated and implemented by considering practical constraints at a general design office. Design costs and computation time are crucial factors. Hence optimisation framework was developed by making use of open source tools and potential solver for CFD computation. The optimiser used was an open source tool developed by Sandia National Laboratories, XPAN was used for CFD calculations and Rhino was used for hull modification. The interface between these tools was developed by the authors using Python. The newly implemented framework is used to optimise a hull form of River Cruise Ferry.

Some similar work has been performed by Raven[1]. The motivation behind this work is to develop a procedure which can be implemented in a ship design office with available tools and reducing the software cost. From the perspective of a design office, time and cost are decisive factors in design. A design process has to be performed with less cost and time. This work aims to achieve an optimised hull, thus acquiring practical knowledge in performing hydrodynamic optimisation by incorporating the widely used CAD and open source tools.

2. POTENTIAL FLOW THEORY

In potential flow theory the flow is assumed to be: inviscid, irrotational, incompressible and steady. The above assumption does not make much variation in the physics of flow around the hull because Reynolds number is high and viscous effects are limited to thin layer and wake. Wave features are not affected by this assumption. This simplifies the mathematics to a great extent. The conservation of mass equation is reduced to

$$div\vec{V} = 0 \quad (3)$$

and the equation (2) becomes

$$\frac{1}{2}\Delta\vec{V}^2 = \rho\vec{g} - \nabla p \quad (3')$$

The velocity vector can be defined as gradient of a scalar function ϕ, known as velocity potential.

$$\vec{V} = \nabla\phi \quad (4)$$

This is substituted in continuity and Navier Stoke's equations:

$$\nabla^2\phi = 0 \quad (5)$$

$$p + \rho gz + \frac{1}{2}(\nabla\phi\nabla\phi) = const \quad (6)$$

The continuity equation changed into Laplace equation which is linear and homogeneous.

2.1. Boundary Conditions

Boundary conditions have to be applied on hull surface as well as free surface.

Kinematic Hull boundary condition: On the surface of the hull, the flow is tangential

$$\phi_n = 0 \quad (7)$$

Dynamic hull boundary condition: In order to determine the dynamic trim and sinkage, an additional boundary condition is introduced for maintaining the equilibrium between hydrodynamic and hydrostatic pressure forces on the ship surface and the ship's weight distribution.

Kinematic free surface condition: It asserts that flow must be tangential to the free surface. Fluid will remain on the free surface $\eta = f(x, y)$

$$\phi_x\eta_x + \varphi_y\eta_y - \phi_z = 0 \quad (8)$$

Dynamic free surface condition: Pressure at the free surface is equal to the atmospheric pressure. Bernoulli's equation can be written as

$$g\eta + \frac{1}{2}\left[\left(\phi_x^2\right) + \left(\phi_y^2\right) + \left(\phi_z^2\right) - U_\infty^2\right] = 0 \quad (9)$$

Bottom Boundary condition : The depth of the water is assumed infinite, and the associated far-field condition is $\phi \to x$ for $z \to -\infty$

Radiation condition : The radiation condition states that free surface waves generated by a ship cannot travel in the upstream direction.

2.2. Linearisation of free surface boundary conditions

The free surface boundary conditions described by equations (8) and (9) are non-linear. Hence the solution depends non-linearly on the location of the free surface and it is difficult to solve them. Hence the equations are linearised by splitting the velocity potential (ϕ) into basic known solution (Φ) and a disturbance (φ)[1]

$$\nabla\phi = \nabla\Phi + \nabla\varphi \quad (10)$$

$$\eta = H + h \quad (11)$$

Then substituting into boundary conditions (equations (8) and (9)) yields

Kinematic free-surface boundary condition:

$$\Phi_x\eta_x + \Phi_y\eta_y + \varphi_x H_x + \varphi_y H_y - \Phi_z - \varphi_z = 0 \quad (12)$$

Dynamic free-surface boundary condition:

$$\eta = \frac{1}{2g}\left(U_\infty^2 - \Phi_x^2 - \Phi_y^2 - \Phi_z^2 - 2\Phi_x\varphi_x - \right.$$
$$\left. - 2\Phi_y\varphi_y - 2\Phi_z\phi_z\right) \quad (13)$$

3. OPTIMISATION ALGORITHM

Dakota is the optimiser used for the work. It is a sequence of iterative mathematical and statistical methods that interface with computational models. The single objective genetic algorithm is available in Dakota as the method called colinyear [2]. The basic steps of the algorithm are depicted in Figure 1 and explained below.

1. Generate a random initial population (hull variations) and perform function evaluations (CFD analysis) on these individuals.

[1] A good estimate will result in a small perturbation by which linearisation can be justified.

2. Perform selection for parents based on the relative fitness (the best hull from the initial population).
3. Apply crossover and mutation to generate new individuals from the selected parents.
4. Apply crossover with a fixed probability from two selected parents.
5. If crossover is applied, apply mutation to the newly generated individual with a fixed probability.
6. If crossover is not applied, apply mutation with a fixed probability to a single selected parent.
7. Perform evaluation for newly generated individuals.
8. Perform replacement to determine the new population.
9. Return to step 2 and continue the algorithm until specified iterations or convergence criteria are met.

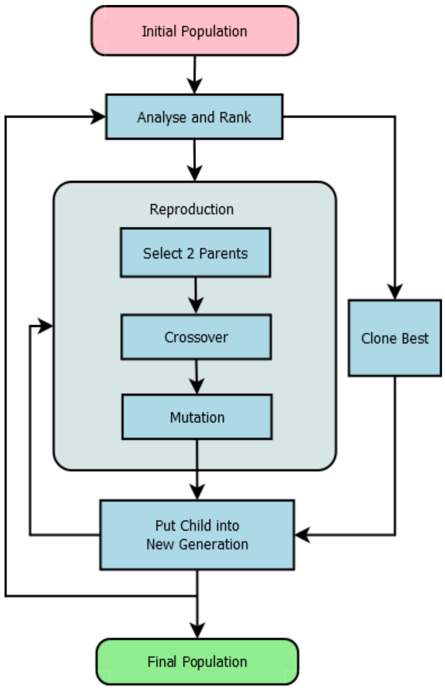

Fig.1. Evolutionary Algorithm [2]

4. DISCRETISATION OF COMPUTATIONAL DOMAIN

The hull form studied in this work (see Figure 2) is of a Luxury Hybrid Motor driven River Cruiser with an electric drive train, intended to operate as river cruise in Western Europe. One of the key design objectives is to limit consumption of fossil fuel compared to conventional designs. The propulsion system consists of two 4-stroke diesel generators combined with one electric propulsion motor, driving a fixed pitch propeller. 80m^2 of solar panels are used to support electricity generation.

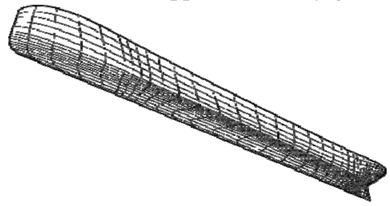

Fig.2. Hull Geometry

Main particulars are given in Table 1.

Table 1. Main Particulars

Length Overall	38.456	m
Length Between Perpendiculars	36.145	m
Stern Over Hanng	2.311	m
Breadth	5	m
Draught	1.4	m
C_B	0.78	

The hull shape has a large block coefficient C_B, without bulbous bow and has parallel midbody. The stern part has a skeg. The hull form was provided by the company as Rhino File format (*.3dm*). In Rhino the hull surface is represented using Non-uniform cubic spline formulation (Nurbs).

The given hull geometry is represented by a network of control points defined as pair of (u,v) coordinates: $u \in [0,9]$ and $v \in [0,5]$.

- $(0,v)$ defines the stern profile.
- $(9,v)$ defines the stem profile.
- $(u,0)$ defines the keel.
- $(u,5)$ defines the deck sidings.

The control points provide high freedom on control of hull geometry which makes nurbs a suitable way to represent hull geometry for optimisation problems. Prior to hull form optimisation, a non linear potential analysis of the hull form is performed to obtain the wave making characteristics and the free surface properties.

4.1 Panelling of hull body

An increase in panel number increases the computation time. It is very important that the panelled hull should have the same surface area as the original hull. Another way to judge the panel quality is to calculate the double body resistance without free surface. By D'Alembert's paradox, the sum of longitudinal pressure distribution is zero for the double model potential flow. But it is not zero due to the discretisation of the hull.

Various hull mesh configurations were tested to assess the influence of mesh on the numerical error and deviation from the actual hull area. The number of panels in longitudinal direction was changed from 40 to 120 and in transverse direction from 10 to 40. Double body calculations are done for each configuration at $fn = 0.21785$ which corresponds to a design velocity of 8 Knots. This gives longitudinal pressure coefficient (C_{xdm}) and normalized surface area $\left(\frac{s}{L_{pp}^2}\right)$. The major results are presented in Figure 3. Surface area for each mesh is compared to the actual surface area of the hull (S_{ref}).

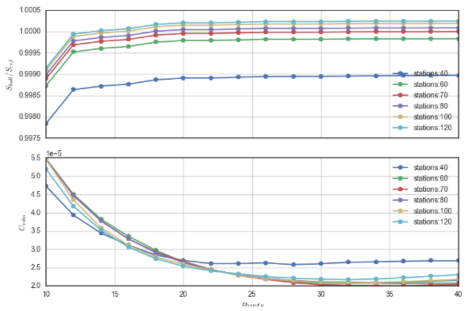

Fig.3. Body panel study

Mesh configuration with station 40 to 60 shows considerable deviation from the actual surface area. Stations = 70 converge to the reference value. The number of panels can be decided from the plot of C_{xdm}. For 80 stations C_{xdm} converges to minimum value from the number of points = 30. Hence (70,30) are chosen for hull meshing.

4.2 Panelling of free surface

The main wave length can be calculated from the equation

$\lambda = 2\pi fn^2 L$

non-dimensionalised quantities using L_{pp}. Hence wavelength for the design $fn = 0.2185$ is $\lambda \approx 0.3$ m. So, minimum panel size should be

$\lambda/10 = 0.03$

Another important parameter is the aspect ratio of the panel which is defined as $AR = dx/dy$. Aspect ratio should be less than 1.

Keeping these key ideas in mind, free surface mesh study has been performed by varying the number of panels along the body (stam) and the panels along the downstream (stad). The domain of calculation has been specified as shown in Figure 4.

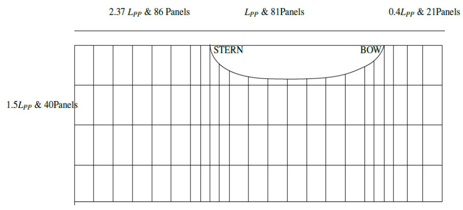

Fig. 4. Free surface panel configuration [3]

Free surface extends from 0.4 from ahead of the bow to 2.37 aft of the stern. Because non-linear effects are strongly closer to the hull in the bow region and the length scale of these effects is a lot smaller than the length scale of the global wave pattern. The longitudinal spacing at bow and stern has been made with fine values, using stretch functions and explicitly applying panel length as 0.01. Hence the spacing increases when moving away from stern and stem on the basis of the number of panels. The results of free surface mesh study are depicted in Figure 5. From the convergence study, it is found that wave resistance values converge from STAM = 81.

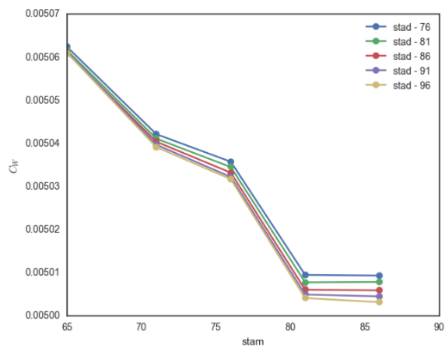

Fig.5. Free surface panel study

Based on this study, the configuration depicted in Figure 6 is adopted for free surface discretisation.

5. OPTIMISATION METHODOLOGY

A design optimisation can be defined briefly as the search for design variable values that improve a set of design criteria. Any optimisation process consists of three parts

- Design variables and constraints
- Design criterion / Response function / Objective
- Optimisation algorithm

An overview of the process is provided in Figure 6.

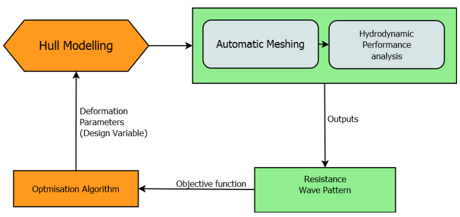

Fig. 6. Overview of Optimisation

5.1 Selection of design variables

Efficient optimisation algorithms are readily available as computer codes, the bottleneck of an optimisation work is the formulation of design space and constraints. A designer has to define this properly for the computer to perform optimisation. The selection of appropriate design space highly depends on the designer's experience in terms of complex problems.

The variables used in the optimisation are parameters meant to modify the hull using Rhino. The hull shape is defined by control points as depicted in Figure 7. A small perturbation to the position of the control points can alter the hull form.

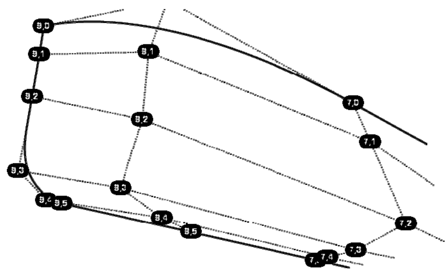

Fig.7. Hull Control points

Control points are indexed by (u,v) where $u \in [0,9]$ and $v \in [0,5]$. The perturbation at the control points is selected as the design parameters for the optimiser. For each control point there is a degree of freedom in each direction. Hence three parameters can be defined for each control point. The number of design parameters should be less for a reasonable optimisation effort. So, it is important to identify the redundant control point for the optimisation and to select only the significant ones.

From the wave pattern analysis it is found that large waves are generated at bow and at stern. So, the hull shape at bow shoulder and aft has got considerable influence on the wave making resistance. Control points defining fore and aft shape are chosen as design variables. The control points for altering the stern shape are (1,2) and (2,2). The fore body can be altered by control points (8,2) and (8,3). Hence four control points are chosen for altering the shape. Hence 12 variables - for the optimisation process.

The limits of the design variables have been analysed using Grasshopper tool in Rhino. Grasshopper is a graphical algorithm editor tightly integrated into Rhino's 3-D modelling tools. A simple algorithm is developed to check the curvature and tangent for water lines and sections at different locations of the hull. The limits of design variables are estimated using an algorithm and are imposed during the optimisation process.

6. OPTIMISATION PROCEDURE

One of the bottlenecks of this optimisation procedure was to interface the Dakota with Rhino and Shipflow. This programs were based on different technologies, Dakota was developed with C++, Rhino was based on .NET. So direct interfacing with these software programs was very difficult and demands a lot of technical expertise in various technologies. The major tasks to be automated during the procedure are:

- Modification of the hull based on parameter file generated by Dakota
- Generation of offsets from the modified hull for SHIPFLOW
- Execution of calculation and saving results for future reference
- Creation of results files for Dakota from output files generated by SHIPFLOW
- Capture of failure in the case of failed calculation
- Coordination of the above processes.

An overview of the implemented optimisation method is depicted in Figure 8. Dakota generates a parameter file during each iteration. Variable values from the parameter file have to be parsed into *csv* format. The file names of the parameter file generated and the resulting file expected are parsed into the script as arguments by Dakota. The parameter file is read and the contents are parsed into the proper format.

The next step is the modification of the hull which is stored as nurb surface in Rhino file format and the generation of the modified offset file for SHIPFLOW. Hull modification using the newly created *csv* file has to be initiated. Rhino has provided Python scripting support to automate a repetitive task and other customisations. Another python script was developed to modify the hull and to generate offsets out of it. The python interface reads the *csv* file and modifies the original hull surface and the function generates the offset file for Shipflow. A status file is also generated to indicate if the operations are successful or not.

One hurdle in the process was the invocation of the later script as it has to be run inside R. Rhino does not provide direct integration with another software program, Python script in the current situation. A solution was to develop an intermediate application to communicate between Rhino and Python interface. Rhino offers a .NET plug-in Software development kit called *RhinoCommon*[2]. A console application was developed in Visual Basic using RhinoCommon SDK to invoke Rhino and run the python script inside it. Script is compiled into windows console program. This console program is invoked by the central python script once the csv file is generated. Console program returns when offset files are successfully generated.

Input of text files and calculations can be initialised through shell commands.

Configuration file remains the same during optimisation, whereas offset file varies. Once Console program finishes its functions and returns, Python script invokes calculation and results are generated in the output directory. Then the result files are parsed by the function and the results are written to Results.

[2]http://wiki.mcneel.com/developer

Scheduling the operations is crucial. It is important to detect the failure in simulations. Wrong simulation results can mislead the optimiser. Hence functions are implemented to detect failure in the simulation and to pass it to the optimiser as failure. Dakota enables to manage failed simulations and recover the optimisation process.

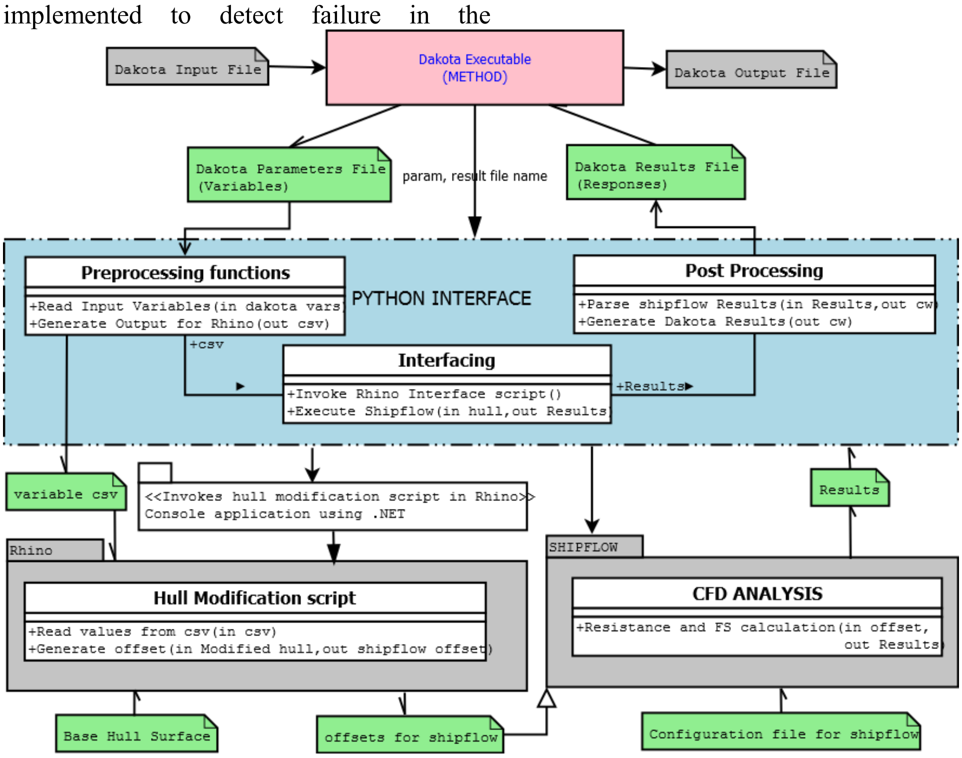

Fig.8. Overview of optimisation platform

7. RESULTS

Using the developed platform, two optimisation processes have been performed using: linear computation and then non-linear computation. The output of the CFD computations is post processed using paraview and the custom programs developed in Python. The total resistance estimate was calculated using empirical relations [4]. ITTC'78 guidelines were used for the estimation of frictional resistance coefficient and Watanabe's formula [5].

7.1 Optimisation using linear computation

Linear calculation provides quick results and is a powerful tool in comparing hull forms and obtaining a comprehensive idea about the flow. So an optimisation employing linear calculation and evolutionary algorithm has been performed. The optimisation process was performed for about 400 iterations. The evolutionary algorithm has weak convergence. In order to get a better understanding of the convergence running, the mean values of the objective function are also plotted in Figure 9. Failed simulations are omitted. It is to be noted that in evolutionary algorithm, instead of an optimal design, a population of optimal designs can be found. Weak convergence is observed.

From the iteration history, a correlation analysis has been performed to understand the influence of the variable on the objective function. The results of the correlation study can be used to identify the redundant variable in the study. An optimal hull form, OD-1, is chosen from the results of the optimisation-1 and is analysed. For analysis, non-linear computation has been performed on the modified hull. Various characteristics were compared with respect to the original hull form.

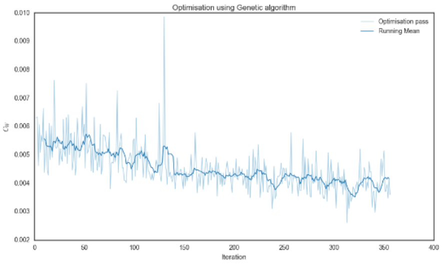

Fig. 9. Optimisation using linear compuation

The design is generated from the final population of the results of optimisation with evolutionary algorithm using linear potential free surface computation. Important particulars are compared and tabulated in Table 2.

Table 2. Result Comparion – Optimisation using linear computation

	OD-1	Original	% Variation
∇	207.5	213.9	-2.98
S_{ref}	231.9	235.1	-1.37
C_W	0.00318	0.00501	-36.58
R_W	6236.0	9969.0	-37.44
R_T	10530.0	14350.0	-26.59

The calculated wave pattern is compared to the reference hull as shown in Figure 10.

The optimized wave pattern is shown at the bottom. The wave system generated at stern shows considerable reduction in the wave system generated in comparison with the original hull form. The main aspect to be considered in analyzing wave making resistance is the far field wave system.

Longitudinal wave cuts are made and compared to the original and shown in Figure 11.

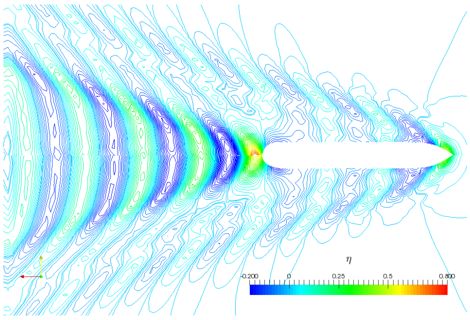

Fig.10. Wave pattern comparison OD-1

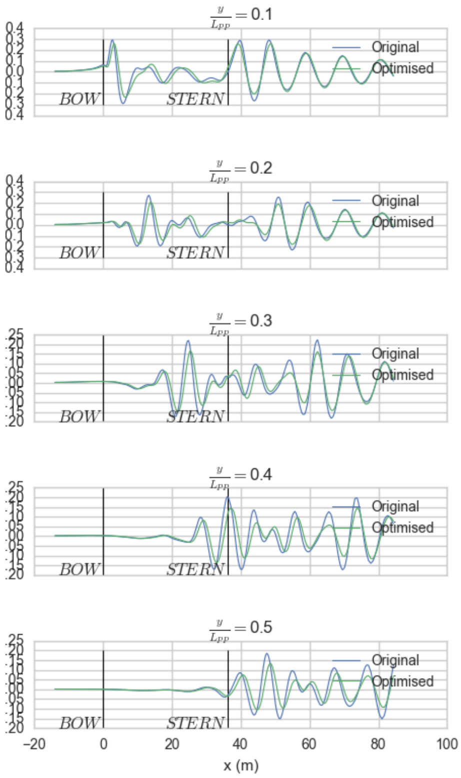

Fig. 11. Longitudinal wave cut comparison OD-1

Longitudinal wave cuts at far field agree with the reduced wave making values for the optimised hull. The amplitudes of the far field waves generated by the modified hull are less compared to the initial hull.

7.2 Optimisation using non-linear computation

Non-Linear computations provide better prediction of wave resistance by considering the non linear effects too. But the calculation demands more computational time and power. The iteration history of optimisation performed using Non-Linear computation is shown in Figure 12. The optimisation convergence history is similar to the one of the Optimisation-1. The same number of iterations has been performed, but it is noted that more simulation failures occurred.

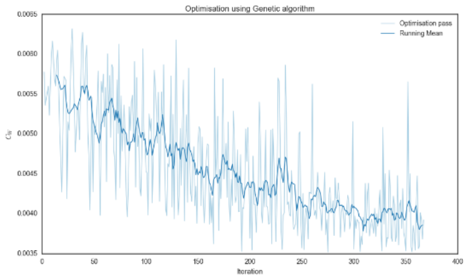

Fig.12. Optimisation using linear compuation

Convergence is weaker when compared to the optimisation using linear computation. Correlation analysis has been performed to assess the influence of the chosen variable on the objective function. Results conclude that the influence of design variable on the objective function is almost the same as in the previous study. The Optimal hull form, *OD-2*, is chosen from the population of optimum designs. The comparison of the characteristic hull features is given in Table 3.

With non-linear computation, wave making resistance of the optimised hull is 27% less than the original hull. But not optimal as the hull obtained from the previous optimisation using linear computation. The comparison between the wave pattern and the far field wave cuts is provided in Figure 13 and Figure 14 respectively. There has been considerable reduction in the stern wave system and the far field system, highlighting the reduction in wave making resistance.

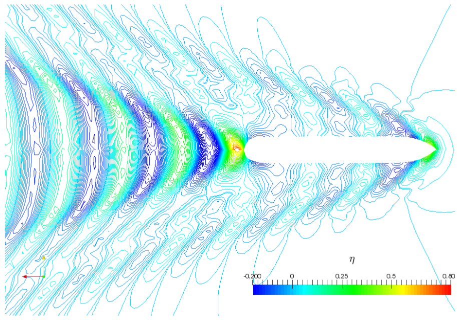

Fig.13. Wave pattern comparison OD-2

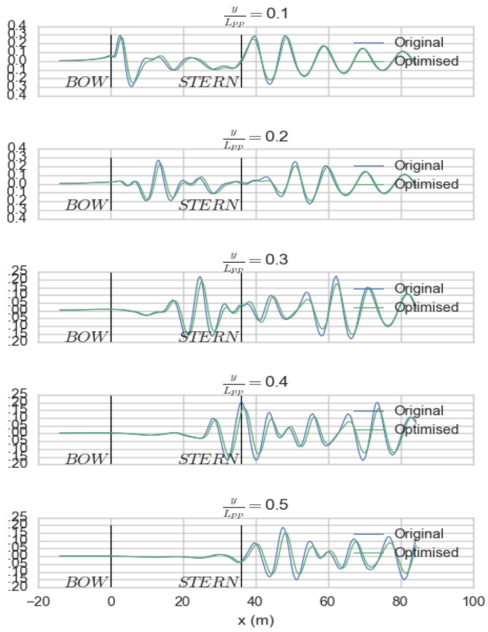

Fig.14. Longitudinal wave cut comparison OD-1

Table 3. Result Comparison – Optimisation using non-linear computation

	OD-2	Original	% Variation
∇	209.2	213.9	-2.187
S_{ref}	233.2	235.1	-0.812
C_W	0.00367	0.00501	-26.63
R_W	7255.0	9969.0	-27.22
R_T	11580.0	14350.0	-19.28

8 CONCLUSION

An optimisation platform incorporating common CAD software programs and open source packages has been successfully developed. The optimisation framework was used to optimise the hull form of a river cruise ferry and the hull form with considerable reduction in wave making resistance was achieved.

ACKNOWLEDGMENTS

This study has been accomplished in the frame of the EMSHIP - European Masters Course in Advanced Ship and Offshore Design Ref. 159652-1-2009-1-BE-ERA MUNDUS-EMMC, at "Dunarea de Jos" University of Galati, Faculty of Naval Architecture and Navyk Design & Engineering, Galati.

REFERENCES

[1]. **Raven, H. C., Hoekstra, M.,** *"A practical system for hydrodynamic optimization of ship hull forms",* VNSI Innovatiedag, 2003.

[2]. **Adams, B.M., Bauman, L.E., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Hough, P.D., Hu, K.T., Jakeman, J.D., Swiler, L.P., and Vigil, D.M.**,*"DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.2 Reference Manual",* 2014.

[3]. **AB, Flowtech Internation.** *"Shipflow 6.0 User Manual",* 2015.

[4]. **Larsson, L., Raven, H.C.,** *"Principles of Naval Architecture Series - Ship Resistance and Flow",* Society of Naval Architects and Marine Engineers (SNAME), 2010.

[5]. **Schneekluth, H., Bertram, V.,** *"Ship Design for Efficiency and Economy (Second edition)",* Butterworth-Heinemann.

Paper received on December 31st, 2015

