Proceedings of MARTEC 2010
The International Conference on Marine Technology
11-12 December 2010, BUET, Dhaka, Bangladesh

HULL FORM PARAMETERIZATION TECHNIQUE WITH LOCAL AND GLOBAL OPTIMIZATION ALGORITHMS

Professor Ho-Hwan Chun

Department of Naval Architecture and Ocean Engineering,
Pusan National University, KOREA
Email: chunahh@pusan.ac.kr

The paper is presented on the development of a designer friendly hull form parameterization and its coupling with a local and a global optimization algorithm: the well known Sequential Quadratic Programming (SQP) and the more recent evolutionary Particle Swarm Optimization (PSO). These two algorithms are representative of classes with rather opposite characteristics (derivative–based and derivative–free, respectively) and their relative performances in solving some typical ship design optimization problem will be discussed in the paper. Following a well known naval architect's design practice, a parametric modification tool is developed for modifying the ship's geometry. The original geometry can be easily deformed by direct selection of some standard design parameters and useful information about the effect of the changing in the parameters are immediately obtained and visualized. At the same time, design parameters are assumed as design variables in the formulation of the optimization problem. In the examples, both potential flow and RANS solvers have been used. Numerical results for both single and multi-objective problems are presented.

1. INTRODUCTION

Simulation Based Design (SBD) is an emerging engineering tool to deal with complicate optimization problems which come out in diverse technical sectors, ship hydrodynamics included. The developments in CFD and computer power offer to the design engineers the chance for a more integrated and more frequent use of SBD in the ship design process.

The fundamental elements of SBD in the ship design involve numerical solver, optimization technique and geometry modeling and modification. Several advanced optimization techniques are introduced and applied: multi objective global optimization [1-4], variable-fidelity approach [3] and multi-disciplinary design optimization [5]. RANS solver [2-3, 6-8] and potential flow solver [1, 9-15] are used for the numerical solver. Various hull form modeling and manipulation methods are introduced: Point manipulation or vertex control [7, 11, 14, 16, 17], perturbation surfaces generated by geometric modification functions such as B-spline definition and Bezier patch [3, 8, 13, 18] and parametric modeling [19-22].

However, arguably, these methods are not as generally accepted or widely used in practical ship design as the optimization community initially hoped. The explanation is not straightforward. It is certainly true that the consideration of a number of different problems is required: robust and automated grid generation and manipulation, the need to account for complex, real-industrial constraints, the difficulty of generating the objective function values computed by solving Partial Differential Equations (PDE) to number a few.

But a more basic point is the parameterization of the problem (which has to be as familiar as possible to the ship designer), an issue that must be overcome before SBD can make a widespread impact on the practice of ship design, but this doesn't seems to be the fundamental point.

The potential benefits and pay-offs of the impact of SBD on the ship design process are so great, however, that despite the damping effects of reality on the immediate expectations, research on SBD has continued, yielding promising results and revealing specific new challenges and directions of research, which are reflected in several open literature[1, 2, 17, 19, 20]

In the paper a designer hull form parameterization is presented and tested by coupling it with advanced local and global optimization algorithms. As optimization algorithms, we choose to implement the Sequential Quadratic Programming (SQP, e.g. Rao [23]) and the Partial Swarm Optimization (PSO, e.g. Pinto et al. [1]) as representative of algorithms with rather opposite characteristics, and their relative performances are discussed.

2. PROBLEM FORMULATION AND SBD ENVIRONMENT

The mathematical formulation of the optimization problem follows that of a Nonlinear

Programming (NLP) problem: all the design variables $x_1, x_2, \ldots, x_N$ constitute a vector $x = (x_1, x_2, \ldots, x_N)^T$ belonging to a subset χ of the N-dimensional real space $\Re^N$, that is $x \in \chi \subseteq \Re^N$ (upper $x^u{}_i$ and lower bounds $x^l{}_i$ are typical enforced onto the design variables), satisfying also some auxiliary condition like $h(x) = 0$ and/or $g(x) < 0$. The objective of the optimization f under the equality and inequality constraints (h and g) are functions of the variables x and of the state of the system u. In its general form, the constrained NLPs problem is to find the particular vector $\tilde{x}$ in the subset χ which solves the following:

$$\min f(x, u(x)) \qquad x = (x_1, x_2, \ldots, x_N)^T \qquad (1)$$

$$x \in \chi \subseteq \Re^N$$

subject to

$$h_j(x) = 0, \qquad j = 1, \ldots, M$$

$$g_j(x) = 0 \qquad j = 1, \ldots, M$$

$$x^l{}_i \le x \le x^u{}_i \qquad i = 1, \ldots, N$$

By solving a system of PDE of the general form $A(x, u(x)) = 0$, the physical state of the system is obtained. To this aim, the use of some numerical tool - the first constitutive element of the SBD frameworks - to solve the system $A(x, u(x))$ and evaluate the current design x is necessary. If the function used to define the optimization problem is of fluid dynamic nature, as in our case, the step requires the evaluation of the design x via a CFD solver, a process which is itself computationally intensive. Within a standard optimization algorithm - the second fundamental element of a SBD - the solution of these differential equations is required at each iteration of the algorithm. In addition to these two elements, a third one is necessary to complete a SBD environment: a geometry modelling method to bi-univocally provide a link between the design variables and a body shape. When the analysis tools is based on the solution of a PDE on some volume grid around a complex geometry, this task is not trivial and often requires large attention. The flexibility of this element may greatly affect the freedom of the optimizer to explore the design space and also the variety of produced shapes.

3. THE ADOPTED PARAMETERIZATION APPROACH

Following the classical naval architect's approach as well as office design practice, an user-friendly parametric modification tool is adopted for modifying the ship's geometry.

Among the large number of methods available in principle to generate (and change!) a hull form, ship designers, historically, are interested in systematic variations of some general global parameter. The major benefit of this classical parametric modification approach is that the original ship geometry can be easily deformed by direct selection of a limited number of well known design parameters. As a consequence, useful information about the effect of the changing in the parameters are immediately obtained, and easily visualized and understood. At the same time, these design parameters can be considered as design variables in the formulation of an optimization problem. These parameters are then varied systematically one by one, keeping constant all the others, and the ship performances are eventually evaluated with some easy tool to extract design sensitivities.

For a number of reasons, this is not the approach followed by most of the current generation of optimization codes. The geometrical manipulation modules of these codes are indeed typically based on CAD systems (or CAD emulators) and hence make use of mathematical surfaces instead, i.e. describing the ship surface based on NURBS or splines patches, with a limited number of control points, freely adjustable by the designer, in order to obtain the required hull shape. As a consequence, the changes of the main hull parameters are computed a posteriori, and they are not the direct output of the code.

Moreover, a single design variable has typically an influence on more hull parameters, so that the relation between the changes in the hull (from the designer standpoint) and the benefit in performances, is – at least – unclear.

This paper, somewhat, goes back to a practical hull form design approach, in which the hull parameters are directly the variables of the optimization problem, and the parametric modification is fully integrated with both the geometry modification module and the CFD analysis.

The initial hull surface is represented using B-spline surfaces (to generate a grid for the CFD solver) and then design parameters by use of modification(additive) functions are adopted for the hull modification to force finally that changes back onto the CFD grid.

The initial hull surface is represented by using the following B-spline surfaces:

$$Q(u, v) = \sum_{i=1}^{n_i+1} \sum_{j=1}^{n_j+1} B_{i,j} N_{i,k}(u) M_{k,l}(v) \qquad (2)$$

where the $B_{i,j}$ are the vertices of a polygon net, $N_{i,k}$ and $M_{j,l}$ the B-spline basis function in the bi-parametric u and v directions, respectively.

The parametric modification function is superimposed on the initial hull (H_{old}) to obtain modified geometry (H_{new}):

$$H_{new}(x,y,z) = H_{old}(x,y,z) + r(x)\cdot s(y)\cdot t(z) \quad (3)$$

The three parametric modification functions (r, s and t) are polynomial functions defined along the x, y, z direction, respectively. The design parameters for optimization can be changed using the functions and the modified geometry can be obtained using the perturbation with specific direction depends on design parameters. In this paper,

(I) sectional area curve (SAC),
(II) section shape and
(III) bulb shape

are used as design parameters of fore-body hull in Table 1.

This parametric modification approach can also be applied to multi-block grids or unstructured grid because the modification functions are based on its position. The smoothness is guaranteed because the modified geometry is constricted by modification functions. The main disadvantage of this approach is that it is not fully flexible and it allows us to obtain the modified geometry according parametric modification function which is already defined.

3.1 SAC Parametric Modification function

Various hull forms can be derived by parametric modification of SAC.
The given SAC is modified by adding the parametric modification function of SAC shape, $r^{(6)}(x)$ which is a 6th order polynomial function defined with only x direction. The modification function, $r^{(6)}(x)$ will be parametrically changed by variation of $\Delta x_{SAC}, x_c, x_0$ and x_1 considered as design variables as shown in Figure 1. Δx_{SAC} is maximum longitudinal movement, x_c is a fixed position where the shape of section is fixed and x_0 and x_1 represent the reference range of modification. To preserve the total volume of the ship, the maximum longitudinal movement needs to be in the middle between x_0 and x_C, and also between x_C and x_1 with the opposite direction as shown in Figure 1.

In order to determine the polynomial coefficients of $r^{(6)}(x)$, seven boundary conditions need to be satisfied(i.e. $r(x_0)=0,\ r(x_1)=0$, $r(x_C)=0,\ r'(x_0)=0,\ r'(x_1)=0$, $r(0.5x_0+0.5x_C)=\Delta x_{SAC}$, $r(0.5x_C+0.5x_1)=-\Delta x_{SAC}$)

Table 1 Design parameters for the hull form optimization

Design parameter		Symbol	Annotation
SAC Shape		Δx_{SAC}	Max movement
		x_1, x_0, x_C	Fixed section
Section Shape	U-V type	Δy_{UV}	Max movement
		$z_{0_{UV}}$	Fixed waterline
		$z_{1_{UV}}$	Max movement
	DLWL type	Δy_{DLWL}	Max movement
		$z_{0_{DLWL}}$	Fixed waterline
		$z_{1_{DLWL}}$	Max movement
Bulb Shape	Bulb area	Δy_{BA}	Max movement
	Bulb length	Δx_{BL}	Max movement
	Bulb height	Δz_{BH}	Max movement
	Bulb size	Δz_{BS}	Max movement
Transom Height		Δz_{TS}	Max movement

As a result of the modification of the original SAC, the new section shape at any longitudinal position can be obtained by the original sectional area curve by Lackenby-method (see Lackenby [23]). The new grid point can be obtained by moving the grid point with x direction according to the modification function as follows,

$$x_{new} = x_{old} + r^{(6)}(x) \tag{4}$$

3.2 Section shape parametric modification function

There are two types of the modification functions: DLWL type and U-V type. DLWL type is the function to modify the design load waterline (DLWL) and U-V type is the function to modify the section shape into U-shaped section or V-shaped section.

The parametric modification functions of section shape are polynomial functions which are three polynomials defined with x, y, z direction, respectively. The modified grid point can be obtained using the perturbation with y direction where the amount of perturbation can be obtained by multiplying three modification functions as follows,

$$y_{new} = y_{old} + r^{(4)}(x) \cdot s^{(5)}(y) \cdot t^{(3)}(z) \tag{5}$$

Where $r^{(4)}(x)$ is a 4th order polynomial function defined with only x direction, $s^{(5)}(y)$ a 5th order polynomial function defined with only y direction and $t^{(3)}(z)$ a 3rd order polynomial function defined with only z direction. These parametric modification functions will be parametrically changed by variation of $\Delta y, z_0$ and z_1 . Δy_{DLWL} and Δy_{U-V} represents maximum horizontal movement while z_0 is kept fixed and z_1 is the position where horizontal movement is maximum. Figure 2(upside) shows distributions of the DLWL type modification functions and Figure 2(downside) shows distribution of the U-V type modification functions for combinations of $\Delta y, z_0, z_1$. Figure 3 depict the original section shape and modified section shape according to DLWL type modification function shown in left figure and U-V type modification function shown in right figure.

3.3 Bulb shape parametric modification function

Modification of bulb shape can be conducted by four design parameters: bulb area, bulb height, bulb length and size.

Following the similar procedure with section shape parametric modification, the modified grid point can be obtained adding the perturbation with each direction where the amount of perturbation can be obtained by multiplying three modification functions as follows,

$$\text{Bulb are } y_{new} = y_{old} + r^{(3)}(x) \cdot t^{(5)}(z) \tag{6}$$

$$\text{Bulb length : } x_{new} = x_{old} + r^{(4)}(x) \tag{7}$$

$$\text{Bulb height : } z_{new} = z_{old} + r^{(4)}(x) \cdot s^{(1)}(y) \cdot t^{(5)}(z) \tag{8}$$

$$\text{Bulb size : } z_{new} = z_{old} + r^{(4)}(x) \cdot s^{(1)}(y) \cdot t^{(6)}(z) \tag{9}$$

These parametric modification functions will be parametrically changed by variation of Δy_{BA}, Δx_{BL}, Δz_{BH} and Δz_{BS} which represent maximum movement with each direction as shown Figure 3.

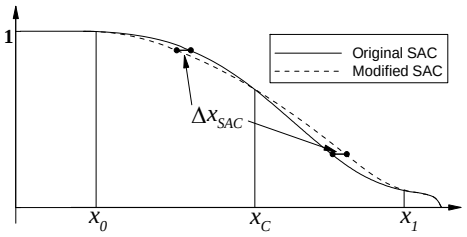


Figure 1 The four design variables ($\Delta x_{SAC}, x_c, x_0, x_1$) description of the fore body part of the sectional area curve (SAC)

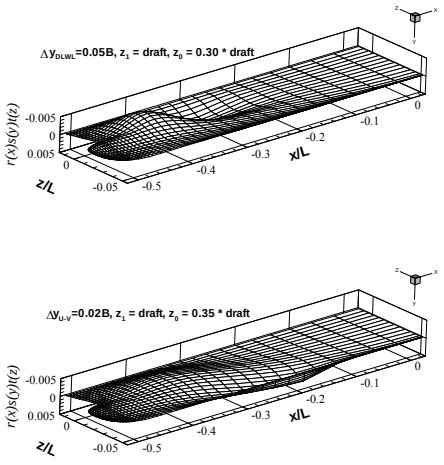


Figure 2 The distributions of the DLWL type modification functions(upside) and the distribution of the U-V type modification functions(downside) for combinations of $\Delta y, z_0, z_1$

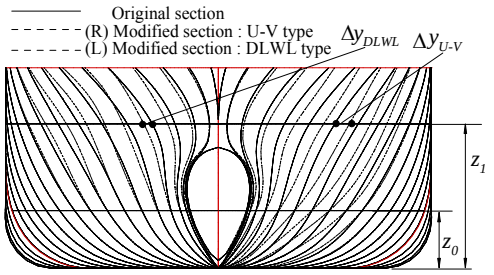


Figure 3 The six design variables description of the section shape: $\Delta y_{DLWL}, z_{0_{DLWL}}, z_{1_{DLWL}}$ (DLWL type in left side); $\Delta y_{UV}, z_{0_{UV}}, z_{1_{UV}}$ (U-V type in right side)

Figure 3(upside) shows the original bulb area and modified bulb area bulb area according the bulb area modification. Bulb area is defined in the area of section in F.P. (i.e. fore perpendicular). Figure 3(downside) depicts original and modified bulb length, height and size according the modification functions, respectively.

4. THE OPTIMIZATION ALGORITHMS

Most general-purpose optimization software used in industrial applications makes use of gradient-based algorithms, mainly due to their convergence properties and computational efficiency when a relatively small number of variables is considered. However, local optimizers have difficulties with local minima and non-connected feasible regions. Because of the increase of computer power and of the development of efficient Global Optimization (GO) methods, in recent years nongradient-based algorithms have attracted much attention. Furthermore, GO method provide several advantages over local approaches. They are generally easy to program and to parallelize, do not require continuity in the problem definition, and are generally better suited for finding a global, or near global, solution. In particular, these algorithms are ideally suited for solving discrete and/or combinatorial type optimization problem. In this paper, the derivative-based SQP and the GO approach PSO are compared, focusing on their effectiveness and efficiency.

4.1 A local, gradient-based optimization algorithm: SQP

Sequential quadratic programming (SQP) is an efficient, gradient-based, local optimization algorithm. The method has a theoretical basis that is related to the solution of a set of nonlinear equations using Newton's method, and the derivation of simultaneous nonlinear equations using Karush-Kuhn-Tucker conditions to the Lagrangian of the constrained optimization problem.

The equations are approximated with a quadratic form:

$$\min \quad 1/2 d^T B d + f(x)^T d \tag{10}$$

subject to

$$h_j{}^T d + h_j(x) = 0, \qquad j = 1,\ldots,M$$

$$g_j{}^T d + g_j(x) = 0 \qquad j = 1,\ldots,M$$

$$x^l{}_i \le x \le x^u{}_i \qquad i = 1,\ldots,N$$

where d is search direction vector and B is approximate Hessian matrix of the Lagrangian. During the optimization process the optimum d is determined and x is updated by

$$x_{n+1} = x_n + d$$

at each iteration (see e.g. Rao [24]).

4.2 A global, derivative-free optimization algorithm: PSO

The growing interest for PSO to solve distinctive global optimization problems (e.g. ship design) is encouraged by the following appealing features: (i) balance, between the computation involved and the precision of the solution detected; (ii) constant computational cost and memory engagement at each iteration; (iii) availability of a current approximate solution; (iv) derivatives of the objective function not required; (v) easy implementation and parallelization of the method. However PSO iteration is neither able to guarantee the convergence to a global minimum nor to a local minimum. Indeed, PSO is a heuristic method, and its reformulations in the literature are heuristics as well.

PSO simulates the social behaviour of a group of individuals by sharing information among them while they are exploring the design space. Each particle of the swarm has its own (individual) memory to remember the places visited during the exploration, whereas the swarm has its own (collective) memory, to memorize the best locations ever visited by any of the particles. The particles have an adaptable velocity and investigate the design space analyzing their own flying experience, and the one of all the particles of the swarm. Each particle is a potential solution of the optimization problem under consideration. The basic algorithm is simple:

- **Step 0** (Initialize)

Distribute a set of particles inside the design space, using some user-defined distribution. Evaluate the objective function in the particles' position and find the best location ($\mathbf{p}_b$).

- **Step 1** (Compute particle's velocity)

At the step $n+1$ calculate the velocity vector $\mathbf{v}_i$ for each particle i using the equation:

$$\mathbf{v}_i^{n+1} = \chi\left[w^n \mathbf{v}^n + c_1 r_1^n (\mathbf{p}_i^n - \mathbf{x}_i^n) + c_2 r_2^n (\mathbf{p}_b^n - \mathbf{x}_i^n)\right] \quad (11)$$

Where χ is a speed limit, w is the inertia of the particles, controlling the impact of the previous velocities onto the current one. The second and third terms, with weights c_1 and c_2, are the individual and collective contributions, respectively and finally, r_1 and r_2 are random coefficients uniformly distributed in [0,1].

- **Step 2** (Update position)

Update the position of each particle:

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \mathbf{v}_i^{n+1} \quad (12)$$

- **Step 3** (Check convergence)

Go to Step 1 and repeat until some convergence criterion (e.g. the maximum distance among the particles) is matched.

DPSO is a deterministic version of the basic PSO for constrained single objective problems which includes several algorithmic improvements. More details are given in Campana et al. 2006. A multi-objective version of the DPSO has been recently presented in Pinto et al. 2007 to which the interested reader is referred.

Experimental results indicate that a large value of the inertia w promotes a wide exploration of the global search space. Hence w is initially set to a high value and then gradually decreased ($w_{n+1} = w_n K$, with $K < 1$) to facilitate the fine-tuning of the current search area. The set of parameters adopted in the computations is given in Table 2.

Table 2 Particle swarm optimization parameters for single objective problem

Parameters	Value
Constriction parameter (speed limit) χ	1.0
Initial inertia weight w_0	1.4
Decreasing coefficient K for the inertia	0.975
Individual parameter(c_1)	0.4
Social parameter(c_2)	0.3

5. NUMERICAL RESULTS

5.1 Single objective tests

In order to demonstrate the effectiveness of the parametric modification and its coupling with local and global optimizer, a container hull-form optimization problem is solved with both the SQP and the PSO optimization algorithms for single objective problem. The optimizers are applied to the KRISO Container Ship (KCS, main dimensions: $LBP = 230m$, $Breadth = 32.2m$, $draft = 10.8m$). The geometrical constraint is on the displacement (which is kept fixed ± 1% of the original value). The keel line is kept fixed, but bulb profile can be changed.

Finally, thedesign variables are limited by some box constraints, defining the range that we like to explore: $40\% \times LBP$, from Station 12 to the bulb tip. The offset should join smoothly the original hull at Station 12.

For the single objective function problem the goal is simply to reduce the wave resistance coefficient C_W at fixed speed (F_N = 0.26). The motivation for the choice of using C_W is given by the fact that the focus of this paper is on linking the parametric modification approach with local and global optimizers, and therefore there was no need for using more complicated objective functions.

The numerical solver adopted for this test is a standard free surface potential flow solver with nonlinear free surface conditions satisfied at the exact free surface position. The solution procedure used to solve Laplace equation subject to nonlinear free surface boundary conditions and other boundary conditions is based on the Rankine source panel method (Choi et al. [26]; Raven [27]). The wave-resistance coefficient C_W is evaluated by integrating the pressure over the wetted hull surface. The details of the nonlinear potential flow solver are given in Choi et al. [26].

Even if the adopted panel solver has the capability of imposing the exact non-linear boundary conditions on the free surface, for the single objective problem test here, it is decided to use the linearized version of the code solving only the linearized free surface boundary condition. This allowed for a significant decrease of the computational complexity without having diminished the scope of the numerical experiment. However, in multi-objective problem test, non-linear boundary conditions are adopted to enhance the numerical accuracy.

For the single objective problem we will provide the numerical results of relative to five different cases, consisting of different number of design

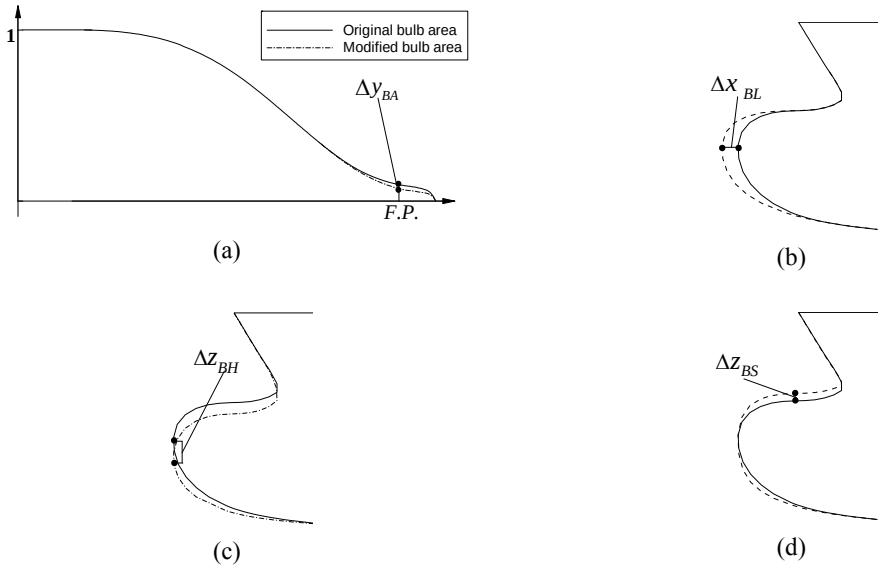


Figure 4 (a, b, c, d) The four design variables description of bulb shape: Δy_{BA} (bulb area in upside); Δx_{BL}, Δz_{BH} and Δz_{BS} (bulb length, bulb height and bulb size in downside)

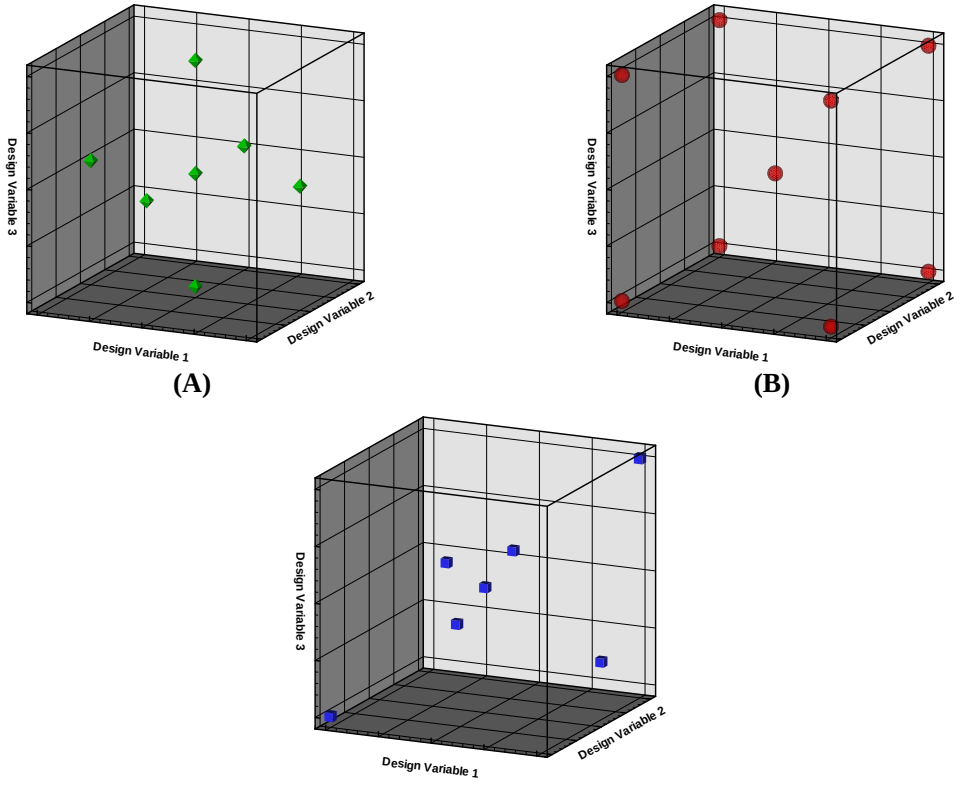


Figure 5 Initial distribution of the swarm particles for 3 design variables.
(A) PSO-hcf, (B) PSO-hcv (C) PSO-sobol.

variables, from 2 to 6. Table 3 summaries all the cases. Results indicate, quite naturally, that by increasing the number of variables we give more freedom to the optimizers, therefore allowing the algorithms find better optima.
The numerical results for the single objective tests are summarised in Table 4 as the percentage C_W reduction with respect to the original value (0.3544×10^{-3}).

Table 3 Problem number N is obtained by adding the corresponding design parameter to those used in the problem number (N-1).

Problem	Number of variables	Design parameters
OPT 1	2	Entrance angle in SAC* + Section Shape (U / V type)
OPT 2	3	Add : Section shape(DLWL type)
OPT 3	4	Add : Bulb area
OPT 4	5	Add : Bulb height
OPT 5	6	Add : Bulb size

* SAC is the Section Area Curve.

Table 4 Numerical results of the five different single objective problems indicated in Table 3. ΔF is the C_W reduction (%) obtained, whereas N_j (reported in brackets) is the number of function evaluations. Original C_W value at F_N =0.24 is 0.3544×10^{-3}.

	SQP	PSO-hcv	PSO-sobol	PSO-hcf
	ΔF % (N_f)	ΔF % (N_f)	ΔF % (N_f)	ΔF % (N_f)
OPT 1	15.2 (44)	15.6 (35)	15.7 (35)	9.1 (40)
OPT 2	20.2 (37)	19.3 (90)	18.6 (56)	19.3 (63)
OPT 3	24.5 (56)	24.2 (221)	22.3 (72)	21.3 (63)
OPT 4	29.0 (90)	27.8 (429)	18.2 (88)	21.3 (77)
OPT 5	32.8 (76)	32.4 (650)	29.9 (104)	21.6 (91)

Three different PSO implementations have been tested, based on different initial distribution of the particles in the design space (see Figure 5), and each of the three might require different swarm sizes:

- **PSO-sobol** (with the swarm particles initially distributed according the Sobol quasirandom sequence[1]) in general do not require a fixed size of the swarm. Here, the selected swarm size is of $2 \cdot N\nu + 1$ ($N\nu$ is the number of design variables);
- **PSO-hcv**, with the swarm particles initially distributed at the vertices of the hypercube representing the feasible set and center point, requires $2^{N\nu}+1$ particles;
- **PSO-hcf**, with the swarm particles initially distributed at the centre of the hypercube faces and center point, requires instead only $2 \cdot N\nu + 1$ particles.

The interesting quantity to be monitored are hence both the resistance reduction rate and the number of function evaluations N_f, being related to the overall computational effort.
An overall comment is that SQP and PSO-sobol show comparable performances, PSO-hcv is computationally slow to converge. PSO-hcf is as efficient as SQP in terms of N_f : although capable of reducing the objective function significantly, it doesn't reach however the C_W % reduction found by the other methods.
By looking at Table 4 it can be clearly observed that all the three approaches show better performances when the number of variables increases (see also Figure 6). The SQP method demonstrates a fast convergence for all cases, requiring a relatively small value of N_f . To explain this result, a further analysis has to be carried out, by looking at the shape of OPT1 and OPT2 feasible domains (Figure 7 and 8: the feasible set for OPT3, OPT4 and OPT5 cannot be easily shown, being domains in $\Re^4, \Re^5, \Re^6$, respectively).
The plot of the function iso-contours and of the (volume) constraint clearly shows that the domain is convex and hence, with a unique global optimum. This condition ensures that even a local optimization algorithm can find the global optimum independently from the initial guess. This is an ideal condition for the fast convergence of any gradient-based approach. The good performances of SQP also for OPT3, OPT4 and OPT5, can be seen as an indirect confirmation that the basic features of the set of

[1] The *quasirandom* Sobol sequence (Sobol [27]) is a useful tool in the approximation of integrals in higher dimensions and in global optimization. The sequence is easily obtained: choose a base (say *2*), and with *i* = *1, 2, 3, ...* , write *i* in base *2*, then reverse the digits, including the decimal sign, and convert back to base *10*.

optimization problems we built for this test are relatively simple.

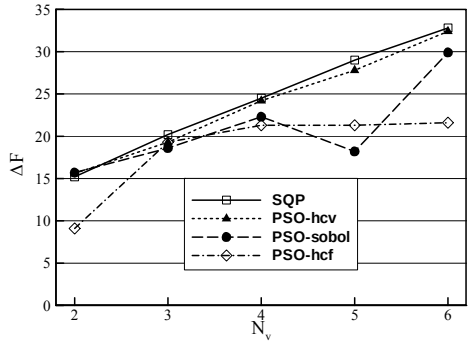


Figure 6 C_W % reduction (ΔF) as a function of the number of variables N_V. SQP (□) and PSO-hcv (▲) increase linearly their performance with N_V. PSO-sobol (●) shows roughly the same trend but N_V =5. PSO-hcf (◊) shows approximately a square-root trend.

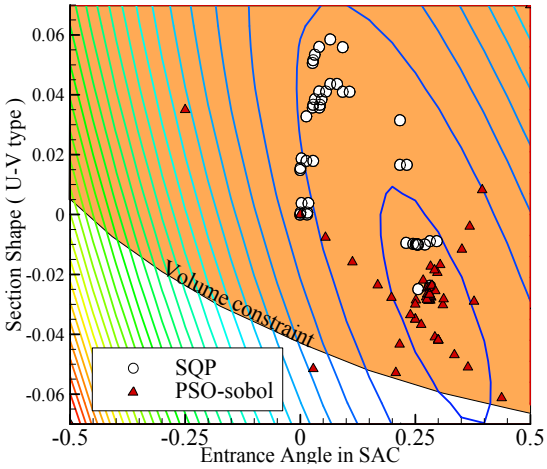


Figure 7 C_w isocontours as a function of the design variables for OPT 1. SQP and PSO-sobol convergence path is also reported.

PSO-hcv shows almost identical C_w reductions with respect to SQP, but with a much higher number of function evaluations N_f (up to one order of magnitude). This is not unexpected: PSO is a derivative-free, global optimization algorithm, and the lack of knowledge of gradient information requires more computational effort. On the other hand, it is expected to work well in cases where any local method would fail. The different behavior between the two methods can be observed in Figure 7 and 8 where the path of the SQP and of the PSO-sobol are compared.

An interesting result is that both PSO-sobol and PSO-hcf show a much reduced number of function evaluations with respect to PSO-hcv and very close to the SQP values (in two cases even less than SQP). These two approaches show however different performances with respect to the C_w reduction: the optimum found by PSO-sobol are very close to the SQP and PSO-hcv best results.

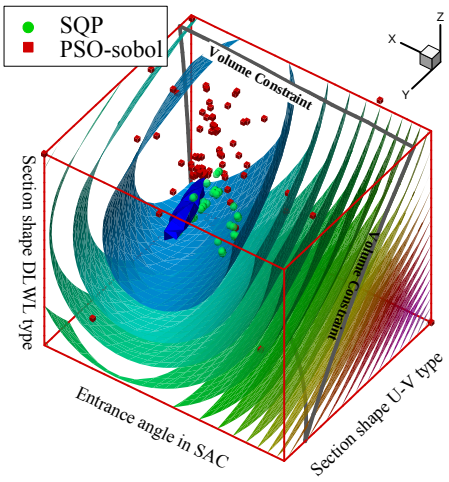


Figure 8 C_w isocontours as a function of the design variables for OPT 2. SQP and PSO-sobol convergence path is also reported.

In conclusion, PSO-sobol shows very interesting features with good efficiency in terms of reduced number of evaluations, a characteristic that is particularly attracting when more complex problem have to be solved (e.g. with more complex constraints).

5.2 Multi-objective test #1

For the multi-objective test #1, the C_w at two speeds (F_N = 0.24 and 0.26) are the two objectives for KCS. Four design variables were used in this test: the entrance angle in SAC, the section shape (U-V type & DLWL type) and the bulb area. The geometrical constraint is the same with that of the single objective problem. This problem is evidently a multi-point optimization problem more than a truly multi-objective, but, as stated before, we are interested in assessing the performances of the optimizers/parametric approach coupling rather than in solving a complicated problem.
The numerical solver is a standard free surface potential flow solver with nonlinear free surface conditions. For the computations, the hull and

the near free-surface are discredited with 1848 and 2370 panels, respectively.
The numerical results for the multi-objective test are summarized in Table 5. SQP, as all the gradient-based approaches, can only deal with single objective problems. To deal with a multi-objective one SQP has to use an aggregated approach, i.e. the problem has to be transformed into a single objective one by making a linear combination, with some user defined weights, of all the objective functions. In this way however, the true nature of the multi-objective problem (i.e. the concept of Pareto front, see as an example in Miettinen [29]) is lost. This is an intrinsic limit of gradient-based methods. Here, the SQP is used to solve the separated problems at the two speeds.

Table 5 The performances of the three selected Pareto front solutions (PSO_ship1, 2 and 3) are compared with the original ones. Original C_W value is 0.29757×10^{-3} (at F_N =0.24) and 0.55547 $\times10^{-3}\times10^{-3}$ (at F_N =0.26).

	Obj function #1 %	Obj function #2 %
Original	100.0	100.0
PSO-KCS1	92.2	93.3
PSO-KCS2	92.4	92.5
PSO-KCS3	93.0	91.5
SQP-KCS1	91.6	94.6
SQP-KCS2	93.7	91.4

Table
6. KVLCC main dimensions and parameters

Parameter	Ship	Model
Scale ratio λ	58.0	
Length LBP (m)	320.0	5.5172
Breadth B (m)	58.0	1.0000
Draft t (m)	20.8	0.3586
Block Coef. C_B	0.8098	0.8098
Reynolds No. R_N		4.6 x 106

The swarm particles are initially distributed according the PSO-sobol approach in the multi-objective problem.
The solutions in the function space with the populations generated in the course of optimization cycles are plotted in Figure 9 in terms of objective function 1 and 2, and the symbol delta denotes the original hull, the symbol diamond the Pareto set after 5 generation and the symbol square the Pareto optimal set.

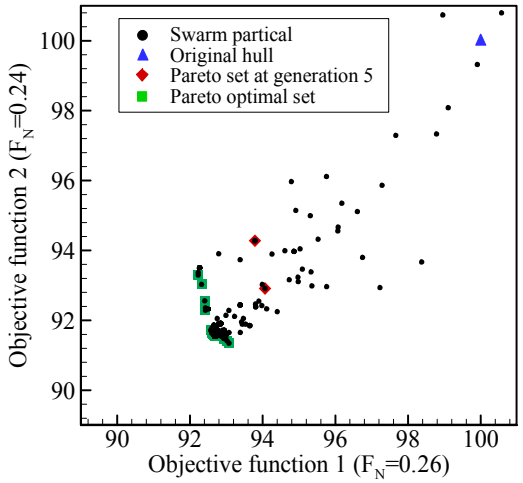


Figure 9 Pareto set after 5 generation and Pareto optimal set

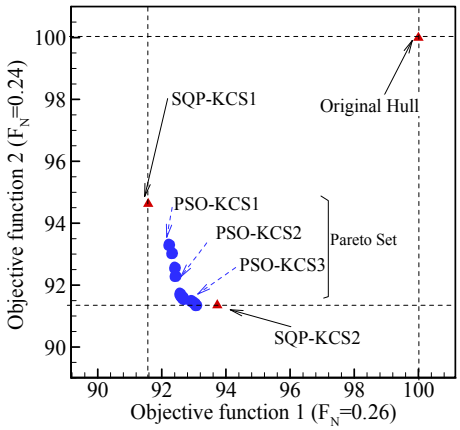


Figure 10 The Pareto front is reported in the function space. SQP-KCS1 and 2 represent the SQP solution of the two single objective problems obtained for the two different speeds. PSO-KCS1, 2 and 3 are three Pareto solutions obtained with the PSO-sobol approach

It can be seen that the PSO optimization algorithm can find the Pareto optimal set where the values of all the objective functions are decreased. Figure 10 shows three PSO solutions selected in Pareto optimal set and SQP solution of the two single objective problems obtained for the two different speeds, whereas their relative performances are reported in Table 5.
In Figure 11-12 the body lines and buttock lines of three PSO solutions selected as a sample are plotted, whereas their relative performances are reported in Figure 9 and Table 5. PSO-KCS1 has better wave resistance performance in higher design speed and PSO-KCS3 has better wave resistance performance in lower design speed within Pareto optimal set. PSO-KCS2 is in the middle of PSO-KCS1 and PSO-KCS3.

5.3 Multi-objective test #2

This second multi-objective test is more complex and computational expensive. The code adopted for the analysis is the RANS solver WAVIS (version 1.3. A description of the code is given in Van et al. [30]).
Being the goal of this example to show the possibility of linking the parameterization technique with volume solvers and to test the effectiveness of the optimizers in this cases, only double body computations have been performed during this test (at model scale). The volume grid adopted in the analysis has $173 \times 41 \times 41$ cells. Each computation required about 25 minutes on an Intel Core 2 - 6600 - 2.4 GHz.

The problem is to optimize the shape of the KRISO VLCC ship (KVLCC, see Table 6 for main dimensions) given two objective functions to be minimized: (i) the form resistance (Hino [31]):

$$1+k = (C_T)_{double\ model} / C_{FO} \qquad (13)$$

where,

$C_{FO} = \dfrac{0.075}{(\log_{10} \mathrm{Re} - 2)^2}$ (ITTC 1957 frictional coefficient)

and (ii) the mean longitudinal velocity V_X at the propeller plane (we take $-V_X$ as objective function to be minimized).
Three design variables have been used in this test: (i) ν_1 = the run angle in SAC (stern region), (ii) ν_2 = the U-V type section shape and (iii) ν_3 = the DLWL type section shape. The adopted box constraints are: on ν_1: 0.3 of the station; on ν_2: 7% and ν_3: 5% of half the breadth. Only the aft part of the hull can be modified (from stations 0 to 8).

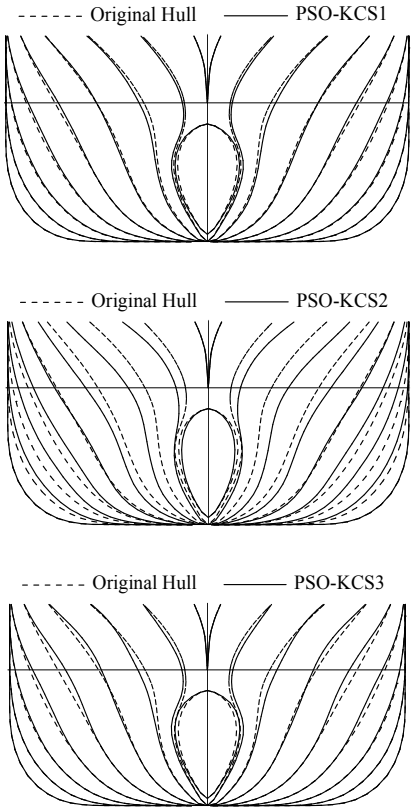


Figure 11 The bodylines of three Pareto front solutions (PSO-KCS1, 2 and 3) are compared with the original ones.

Table 7. Multi-objective test case #2.

	Orig.	PSO-KVLCC1	PSO-KVLCC2	PSO-KVLCC3
$\Delta(m^3)$	312450	311348	310970	310008
CF *1000 1975 ITTC	3.450	3.450	3.450	3.450
CF * 1000	3.374	3.387	3.374	3.400
CVP * 1000	0.913	0.817 (89.5%)	0.856 (93.8%)	1.056 (115.7%)
$1+k$	1.243	1.213 (97.6%)	1.226 (98.6%)	1.292 (103.9%)
V_x	0.32	0.266 (83.1%)	0.360 (112.5%)	0.547 (170.9%)

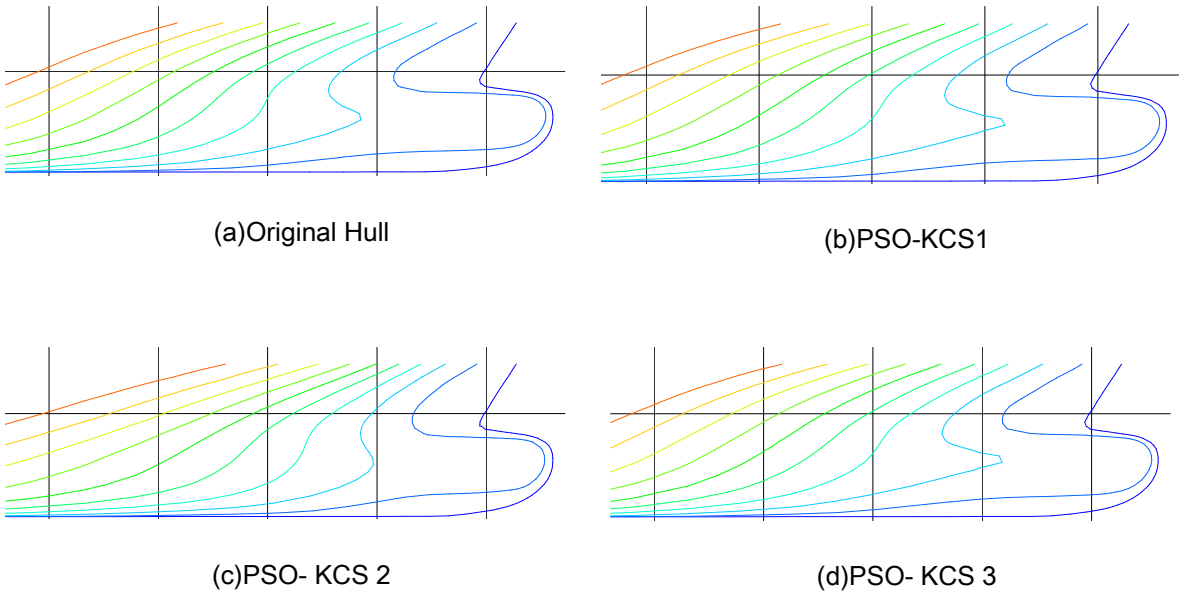


Figure 12 Comparison with the original and Optimum hull shapes

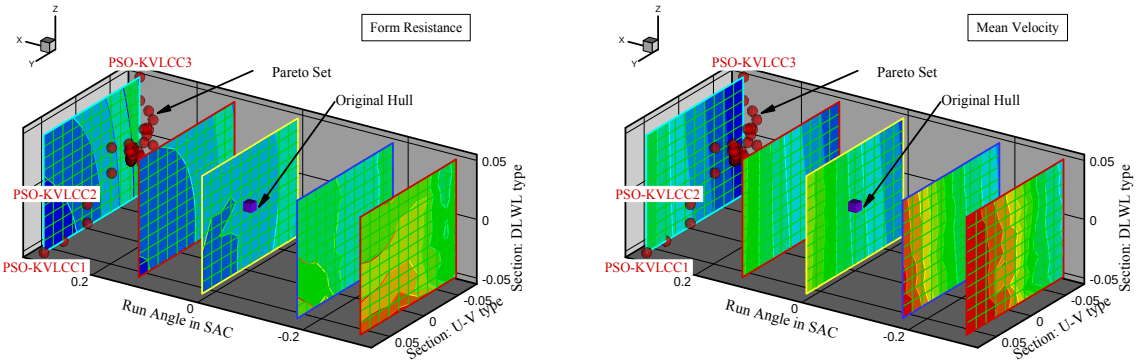


Figure 13 Form resistance (top) and mean velocity (bottom) iso-contours as a function of the 3 design variables. The red dots are Pareto solution.

A regular sampling of the two objective functions was performed before the test started, so that it has been possible to plot their gross structure and to follow the evolution of the optimization procedure (Figure 13). For positive values of v_1 (run angle in SAC) both the objective functions show reduced values with respect to the original hull (the point at $v_1 = v_2 = v_3 = 0$). As to v_2 (the U-V type section shape), the two functions show opposite trends. Minor changes are produced by v_3 (DLWL type section shape).

The discrete approximation of the Pareto front eventually found by the algorithm is also reported in Figure 13. The solutions are close to the box constraint on v_1. Among these Pareto-optimal solutions, The three hulls have been selected as a sample: PSO-KVLCC1, PSO-KVLCC2 and PSO-KVLCC3. Their bodylines, compared with the original ones, are reported in Figure 13.

The geometrical features of these three solutions are easily identifiable. PSO-KVLCC1 has the

highest admissible value of v_2, and hence shows a V-type stern, whereas PSO-KVLCC3 has the lowest admissible value of v_2, hence showing a U type shape. All the three ships display a larger SAC angle (positive v_1). The individual performances of the three ships are reported in Figure 14 and in Table 7, whereas the distribution of the axial velocity at the propeller disk is plotted in Figure 16.

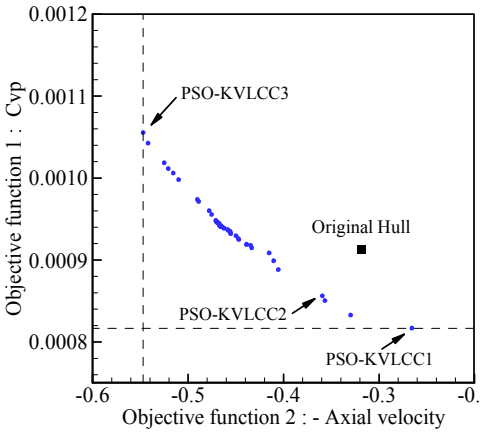


Figure 14 The performances of the Pareto solutions for the multi-objective test case.

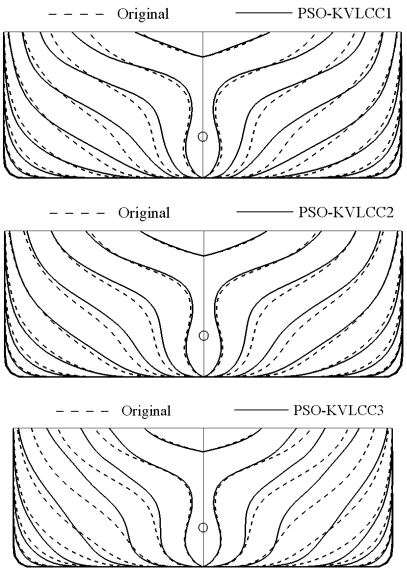


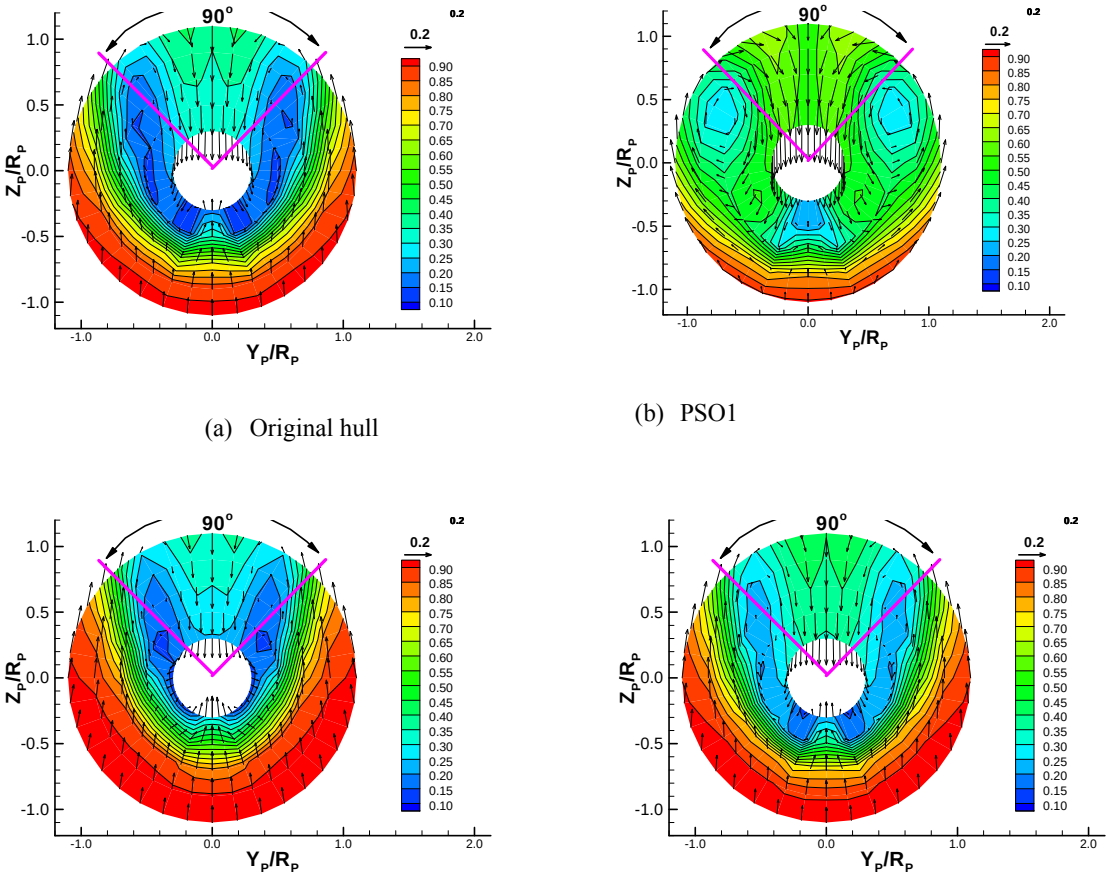
Figure 15 Body lines for three different solutions of the multi-objective test case #2

6. CONCLUSIONS

A designer friendly hull form parameterization and its coupling with a local and a global optimization algorithm have been developed, and applied to several tests of the design of hull form. As optimization algorithms, we choose to implement the Sequential Quadratic Programming and the Partial Swarm Optimization as representative of algorithms with rather opposite characteristics, and they relative performances are discussed.

Following a well known naval architect's design practice, a parametric modification tool is developed and adopted in the present study to produce modified hull forms during optimization cycles. In this approach, the original geometry can be easily deformed by direct selection of some standard design parameters and useful information about the effect of the changing in the parameters are immediately obtained and visualized. At the same time, design parameters are assumed as design variables in the formulation of the optimization problem. This study has also shown that the present parameterization method demonstrated robustness and flexibility in the use.

As an application, a container ship is taken as an initial hull, and local and global optimizer are used to determine the optimal hull forms with respect to wave resistance coefficient at two speeds. A wave resistance reduction at two speeds in the range of 7 to 8 % is achieved. The second multi-objective optimization shows the possibility of linking the parameterization technique with volume solvers and test the effectiveness of the optimizers. The KVLCC is taken as an initial hull and the form resistance and the mean longitudinal velocity at the propeller plane are minimized.



(a) Original hull

(b) PSO1

(c) PSO2

(c) PSO39

Figure 16 Axial velocity distribution at the propeller disk: (a) original hull, (b) PSO1, (c) PSO2, (d) PSO39

ACKNOWLEDGMENTS

This work has been supported partially by the Advanced Ship Engineering Research Center (ASERC) of the Korea Science and Engineering Foundation and partially by Italian Ministry of Transport in the framework of the INSEAN research program 2007-2009.

REFERENCES

[1] Pinto A, Peri D, Campana EF (2007) Multiobjective optimization of a containership using deterministic particle swarm optimization. Journal of Ship Research 51:217-228

[2] Tahara Y, Peri D, Campana EF, Stern F (2008) Computational fluid dynamics-based multiobjective optimization of a surface combatant using a global optimization method. Journal of Marine Science Technology 13:95-116

[3] Peri D, Campana EF (2005) High-fidelity models and multiobjective global optimization algorithms in simulation-based design. Journal of Ship Research 49:159-175

[4] Hirata N (2004) Comparison of genetic algorithm and gradient-based method applied to ship shape optimization. J.Kansai Soc.N.A. 241:59-65 (in Japanese)

[5] Peri D, Campana E F (2003) Multidisciplinary design optimization of a naval surface combatant. Journal of Ship Research 47:1-12

[6] Duvigneau R, Visonneau M, Deng GB (2003) On the pole played by turbulence closures in hull shape optimization at model and full scale. Journal of Marine Science and Technology 8:11-25

[7] Brizzolara S (2004) Parametric optimization of SWAT-hull forms by a viscous-inviscid free

surface method driven by a differential evolution algorithm. 25th Symposium on Naval Hydrodynamics, Newfoundland and Labrador, Canada

[8] Tahara Y, Stern F, Himeno Y (2004) Computational fluid dynamics-based optimization of surface combatant. Journal of Ship Research 48:273-287

[9] Harries S (1998) Parametric design and hydrodynamic optimization of ship hull forms. Ph.D. Thesis, Technical University of Berlin, Germany

[10] Masuda S, Suzuki K (2001) Experimental verification of optimized hull form based on rankine source method. J.Kansai Soc.N.A. 236:27-33 (in Japanese)

[11] Ragab SA (2001) An adjoint formulation for shape optimization in free-surface potential flow. Journal of Ship Research 45:269-278

[12] Dejhalla R, Mrsa Z, Vukokic S (2002) A genetic algorithm aApproach to the problem of minimum ship wave resistance. Marine Technology 39:187-195

[13] Saha GK, Suzuki K, Kai H (2004) Hydrodynamic optimization of ship hull forms in shallow water. Journal of Marine Science and Technology 9:51–62

[14] Choi HJ, Kim MC, Chun HH (2005) Development of an optimal hull form with minimum resistance in still water. Journal of Ship and Ocean Technology 9:1-13

[15] Suzuki K, Kai H, Kashiwababa S (2005) Studies on the optimization of stern hull form based on a potential flow solver. Journal of Marine Science and Technology 10:32-40

[16] Yang C, Lohner R (2004) H_2O: Hierarchical hydrodynamic optimization. 25th Symposium on Naval Hydrodynamics, Newfoundland and Labrador, Canada

[17] Chen PF, Huang CH, Fang HC, Chou JH (2006) An inverse design approach in determining the optimal shape of bulbous bow with experimental verification. Journal of Ship Research 50:1-14

[18] Markov NE, Suzuki K (2001) Hull form optimization by shift and deformation of ship sections. Journal of Ship Research 45:197-204

[19] Harries S, Abt C, Hochkirch K (2004) Modeling meets simulation-process integration to improve design. Sonderkolloquium zu Ehren der Professoren Hagen, Schlüter und Thiel 23. July 2004

[20] Nowaki H (1993) Hull form variation and evaluation. J. Kansai Soc. N.A. 219:173-184

[21] Lowe TW, Steel J (2003) Conceptual hull design using a genetic algorithm. Journal of Ship Research 47:222-236

[22] Softley J, Schiller TR (2002) An approach to advanced ship design using parametric design templates. Transactions - Society of Naval Architects and Marine Engineers 110:245~258

[23] Lackenby, H.; "On the Systematic GeometricalVariation of Ship Forms," RINA Vol. 92, 1950.

[24] Rao, S.S., "Engineering Optimization: Theory and Practice", Third Edition, Wiley-Interscience, 1999

[25] Campana, E.F., Fasano, G., Pinto, A., "Dynamic System Analysis for the Selection of Parameters and Initial Population in Particle Swarm Optimization" (INSEAN Tech Rep. 2006-56) submitted to Journal of Global Optimization.

[26] Choi, H.J., Chun, H.H., "Potential Flow Analysis for a Hull with the Transom Stern", J. of Ocean Engineering and Technology, Vol. 15, No. 1, 2001, pp.1~6 (in Korean).

[27] Raven, H.C., "A Solution Method for the Nonlinear Ship Wave Resistance Problem," Doctor's Thesis, Delft Univ. Techn., Delft, Netherlands, 1996.

[28] Sobol, I., "Uniformly Distributed Sequences with an Additional Uniform Property", USSR Computational Mathematics and Mathematical Physics,Volume 16, 1977, pp. 236-242.

[29] Miettinen, K.M., "Nonlinear Multiobjective Optimization", Kluwer Academic Publisher, 1999.

[30] Van, S.H., Kim, W.J., Kim, D.H., Yim, G.T., "Development of Computational System 'WAVIS' for Hull Form Evaluation", Proceedings of the Annual Spring Meeting, SNAK 1998, pp. 199-202.

[31] Hino, T., Proc. of CFD Workshop Tokyo 2005, Tokyo, Japan.