Numerical Modelling and Simulation of Ship Hull Geometries

Michael Hopfensitz, Juan C. Matutat, and Karsten Urban

Abstract We consider the numerical optimization of the geometry of ship hulls with the Voith-Schneider® Propeller (VSP), an efficient propulsion and steering device. The presence of one or more VSP sets some requirements to the ship hull. We describe modeling, simulation and optimization of such geometries as a 4 step chain. First, we introduce a geometrical model of the ship hull plus attachment parts. Second, this model is transformed into a computational mesh which is used in Computational Fluid Dynamics (CFD) in the third step. The numerical optimization is the fourth part. Our method is fully automatic. We show some numerical results.

1 Introduction

The Voith-Schneider® Propeller (VSP) is an efficient propulsion and steering device, [1]. It is mainly used for ships requiring high steering control such as water tractors and ferries. In the past, the efficiency of VSP has been significantly improved by using numerical optimization, [4, 5, 8]. As a next step, one is interested in the optimization of the complete configuration consisting of ship hull and VSP. This allows an efficient ship design following the requirements of a particular client. The goal of the present paper is to describe a mathematical approach for this complex optimization. It consists of four main parts, namely
1. A geometrical model of the potential ship hull.
2. An automatic mesh generator.
3. A numerical scheme for simulating the flow around ship and VSP.
4. A numerical optimization.
Let us describe these four parts in a little more detail. On one hand, the geometrical model has to be flexible such that many different shapes of ship hulls can be represented. On the other hand, however, such a parametric model has to be as simple

Institute of Numerical Mathematics, University of Ulm, Germany
e-mail: {michael.hopfensitz, juan.matutat, karsten.urban}@uni-ulm.de

as possible since the number of parameters determines the dimension of the optimization problem. We introduce one such model in Section 2. In order to simulate the flow around such a geometry, we need a computational grid. Since this has to be done within a numerical optimization, the meshing has to be performed automatically, see Section 3. The simulation and optimization is briefly described in Section 4. As already mentioned, these four steps have to be performed automatically. Thus, the coupling of the modules in one chain is an important step which is realized by a Perl script allowing fully automatic computations. Finally, we describe some numerical results in Section 5. This paper is partly based upon [3, 6].

2 Geometric Ship Model

In order to determine some 'optimal' ship geometry, we obviously first need to setup a parametric model for a ship hull. As already mentioned, such a model needs to be both as sophisticated and as simple as possible. Sophisticated in the sense, that the model should allow to represent various kinds of geometries including in particular existing ships. Moreover, the more flexibility such a model allows, the more one can hope to determine an optimal ship. Within an optimization one hence seeks to determine specific values for the parameters in the model in order to optimize the ship w.r.t. a given target, e.g. efficiency. This means, that the number of parameters coincides with the dimension of the optimization problem. Hence, one would like to have as few parameters as possible which explains why the model should be as simple as possible. We use techniques from CAGD (Computer-Aided Geometric Design) in order to generate a smooth and watertight parametrical model. The generation of the model must be stable, robust and efficiently computable. Moreover, the model generation has to be included into an automatic optimization.
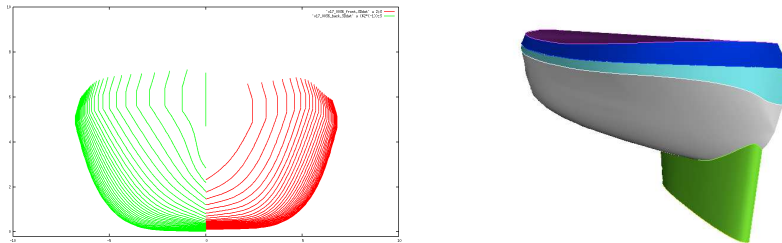

Fig. 1 Ribs in a CAD model (left) and CAD model of VSP Water Tractor composed by different surfaces and an attachment.

In order to understand the specific requirements of a *ship* model, let us consider a CAD model of the VSP Water Tractor in Figure 1. We see that the ship hull consists of different surfaces and at least one attaching part like the fin or a protection plate. Thus, we have to be able to model a hull with different attachments. Moreover,

both in CAD and true building, a ship is made of different ribs. These rips are combined horizontally by so called design curves, such as the water or keel line. The surfaces are determined by the points of the the design curves on the ribs. Both for construction and numerical purposes, it is necessary that the grid lines for the CFD code follow these design curves. Hence, our geometrical model must provide this. We start by describing a model for one particular rib. Due to symmetry, we can reduce ourselves to one half of it, see Figure 2.

Fig. 2 Model of a single rib. The curve is divided into subcurves with different regularity. The water line is indicated by 'wl'. The keel line is located at point A.
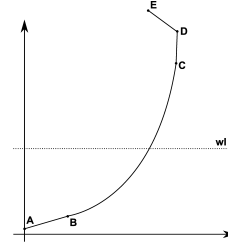
We subdivide each rib into four parts representing different surface parts of the ship hull. We assume that the end points of each part are located on one design curve, so that the collection of corresponding parts of different ribs in horizontal direction describe one surface. This surface is parameterized as

$$S : D := [0,1]^2 \rightarrow \mathbb{R}^3, \quad (u,v) \mapsto \big(P(u), s(v, Q(u))\big). \tag{1}$$

Here, $P : [0,1] \rightarrow \mathbb{R}$ describes the location of a rib, i.e., the scaling in x-direction. With the parameter space $\Omega \subset \mathbb{R}^r$, $r \in \mathbb{N}$, the function $s : [0,1] \times \Omega \rightarrow \mathbb{R}^2$ models the surface. We choose s as a NUBS (Non uniform B-spline, [7]) of the form

$$s(v,\omega) := \sum_{n=1}^{b} \mu_n \mathbf{d}_n(\omega) N_{n,k}(v), \tag{2}$$

where $\mathbf{d}_n$ are the control points depending on the parameters $\omega \in \Omega$ (by solving a linear system of equations), $N_{n,k}$ form a B-Spline basis of order k on a non-uniform grid Δ on $[0,1] \subset \mathbb{R}$ and μ_n are the weights. Finally, $Q : [0,1] \rightarrow \Omega$ describes the design curves. Thus, we obtain a closed surface description. Inserting the NUBS-representation (2) into the geometric model equation (1) shows that the surface is parameterized in terms of the mapping Q for given mappings P and s. Each part on a rib is hence described by r parameters contained in $\omega \in \Omega$. These parameters can describe e.g. the angle, the starting points, the area or the centroid of the particular curve. For an appropriate number of such conditions the NUBS-curve is uniquely described. The parameters used for the description of each block curve have been chosen in close corporation with the engineers from Voith in order to assign physical interpretation to the parameters.

When putting the model for each rib together for describing the surface, we perform a reduction of parameters since there is obviously redundancy. This is done

by numerical optimization w.r.t. a catalogue of existing ship models. For building up a model based on more surfaces the model can be extended in the following way. We define a vector $\mathbf{S} = (S_i)_{i=1,\dots,p} \in \mathbb{R}^p$ whose components S_i are surfaces as modeled in (1). In order to enforce continuity of the complete surface, we enforce $S_i(\cdot, 1) = S_{i+1}(\cdot, 0)$ for $i = 1, \dots, p-1$. Hence, the whole ship hull geometry is parametrized and described by the vector $\mathbf{S}$.

Finally, attachment parts like a fin or a protection plate are modeled in the same rib-wise manner and these parts are attached to the ship hull by computing the intersection curves. This can be done quite efficiently by using the NUBS-representation of the surfaces.

Before the described geometry can be used as an input for our mesh generator, we need to prepare the geometry by determining some 'pre-discretized' surface description. This has been done by using the STL data format. We obtain a smooth triangulation since our geometry description allows for a piecewise smooth representation.

3 Mesh Generation

Based upon the STL-representation, the next step is to generate a mesh for the numerical simulation. Such a mesh needs to fulfill some requirements, namely

1. The mesh has to be generated fast, robust and automatically without manual corrections;
2. It has to give a good representation of the geometry;
3. The mesh has to be of good quality for numerical simulations.

In particular 2. and 3. need to be balanced. For a flexible representation of the geometry, a mesh consisting of tetrahedra is desirable, see also Figure 3. A good mesh quality (e.g. in terms of angles or aspect ratios) can be better realized by hexahedra. This is the reason, why we follow a compromise by creating 3 layers, the inner and outer ones formed by hexahedra, the inner layer by tetrahedra. The inner layer close to the ship surface consists of possibly orthogonal hexahedra in order to optimize numerical precision. This is of particular importance since many optimization criteria (e.g. efficiency) involve physical quantities like the pressure on the surface. The outer layer is formed by hexahedra in order to reduce computational complexity. Finally, the connection between these two hexahedra layers is done by tetrahedra in the inner layer. Tetrahedra allow for more flexibility here. The interface between the layers is done by introducing pyramids in order to maintain mesh quality. Details on our meshing algorithms and tools can be found in [3]. For the generation of the surface mesh and the tetrahedron mesh we use CUBIT from Sandia National Laboratory.

4 CFD Computations and Numerical Optimization

The CFD computation is based on a cell centered finite volume method using the CFD code COMET from CD-Adapco, [2]. The governing equation are the Reynolds averaged-Navier-Stokes equations for incompressible fluids with a k-ε turbulence model. This is appropriate since we have a mach number below 0.3. The equations are well-known read $\mathbf{u}_t - Re^{-1}\Delta\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{g}$, $\mathrm{div}(\mathbf{u}) = 0$, where $\mathbf{u}$ denotes the unknown velocity field, p the unknown pressure, $\mathbf{g}$ a given exterior force and Re denotes the Reynolds number. In addition, appropriate initial and boundary conditions have to be imposed. COMET produces accurate and reliable results, that have been validated both in model experiments and real scale measurements. The simulation of the full model takes in the order of ten hours on a PC cluster with 12 processors when using a mesh with about one million cells.

So far, we use a direct optimization scheme (Hooke-Jeeves) since the gradient of the target is not available. However, first results using Automatic Differentiation are quite promising and we will follow this approach in future work.

5 Numerical Results

Now, we present some of our numerical results. In Figure 3 (left), we show one example of a mesh of a ship geometry. We obtain a smooth mesh following the design curves without any hanging nodes. Of course, we have tested our tool for a variety of geometries. The results are equally satisfying.

In Figure 3 (right) the results of a CFD simulation are shown. The force in x-direction is plotted for each time step of the simulation. We compare different meshes. The red line corresponds to a mesh created and frequently used by Voith. This is our reference using 3.2 million cells. The other curves correspond to different meshes created by our mesh generator. The pink curve corresponds to a mesh without inner layer. Even though the mesh consists of 1.3 million cells, the result is far away from the reference curve. This indicates the importance of the inner layer. The blue and green curve correspond to meshes with different number of cells. The green one matches the reference result quite good, keeping in mind that the mesh has only about 1/3 number of cells then the reference mesh. Increasing the number of cells even more results in a perfect matching of the curves. This clearly indicates the good performance of our mesh generator.

Finally, in Figure 4, we show the results of a numerical optimization w.r.t. the force in x-direction. The figure on the left shows the optimization steps of a direct method including the search steps. The result is shown on the left. We obtain a significant reduction of the target value.
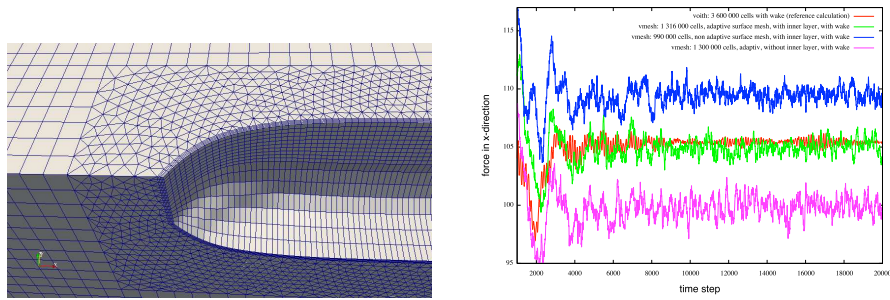
Fig. 3 Mesh of a ship hull geometry (left). Results of the numerical simulation (right). Force in x-direction over the time step (in N scaled on model size).
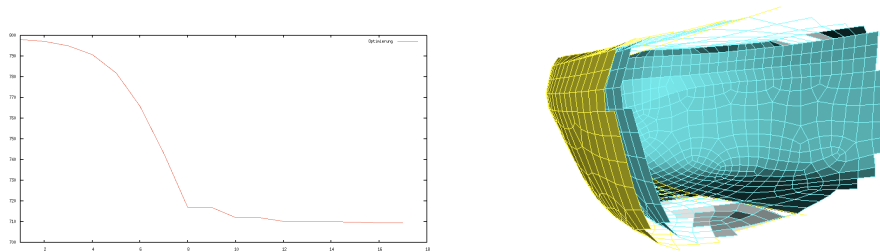

Fig. 4 Numerical optimization w.r.t. force in *x*-direction. Iteration history (left) and resulting ship geometry (right).

References

1. Bartels, J.E., Jürgens, D.: The Voith Schneider Propeller: Current applications and new developments. Tech. rep., Voith Turbo Marine GmbH (2006)
2. Ferziger, J., Peric, M.: Numerische Strömungsmechanik. Springer Verlag (2008)
3. Hopfensitz, M.: Numerische Optimierung von Schiffen mit VSP: Automatische Gittergenerierung. Ph.D. thesis, University of Ulm (2010)
4. Jürgens, D.: Theoretische und experimentelle Untersuchungen instationärer Tragflügelumströmungen und Entwicklung eines Berechnungsverfahrens für Vertikalachsmotoren. Ph.D. thesis, University of Rostock (1994)
5. Jürgens, D., Palm, M., Singer, S., Urban, K.: Numerical optimization of the Voith-Schneider Propeller. ZAMM **87**(10), 698–710 (2007)
6. Matutat, J.C.: Numerische Modellierung und Simulation von Schiffen mit VSP: Geometrie-Modellierung und Optimierung. Ph.D. thesis, University of Ulm (2010)
7. Piegel, L., Tiller, W.: The NURBS book. Springer (1997)
8. Singer, S.F.: Numerische Optimierung der hydromechanischen Parameter des Voith-Schneider-Propellers. Master's thesis, University of Ulm (2003)