

# Optimization of Hydraulic Machinery Bladings by Multilevel CFD Techniques

**Susanne Thum**

*Institute of Hydraulic Machinery and Plants, University of Technology Munich (TUM), 85748 Garching, Munich, Germany  
Email: thum@lhm.mw.tu-muenchen.de*

**Rudolf Schilling**

*Institute of Hydraulic Machinery and Plants, University of Technology Munich (TUM), 85748 Garching, Munich, Germany  
Email: schilling@lhm.mw.tu-muenchen.de*

Received 25 June 2002

The numerical design optimization for complex hydraulic machinery bladings requires a high number of design parameters and the use of a precise CFD solver yielding high computational costs. To reduce the CPU time needed, a multilevel CFD method has been developed. First of all, the 3D blade geometry is parametrized by means of a geometric design tool to reduce the number of design parameters. To keep geometric accuracy, a special B-spline modification technique has been developed. On the first optimization level, a quasi-3D Euler code (EQ3D) is applied. To guarantee a sufficiently accurate result, the code is calibrated by a Navier-Stokes recalculation of the initial design and can be recalibrated after a number of optimization steps by another Navier-Stokes computation. After having got a convergent solution, the optimization process is repeated on the second level using a full 3D Euler code yielding a more accurate flow prediction. Finally, a 3D Navier-Stokes code is applied on the third level to search for the optimum optimum by means of a fine-tuning of the geometrical parameters. To show the potential of the developed optimization system, the runner blading of a water turbine having a specific speed  $n_q = 41$  1/min was optimized applying the multilevel approach.

**Keywords and phrases:** optimization, CFD, multilevel, hydraulic machinery.

## 1. INTRODUCTION

The design optimization of hydraulic machinery can be performed by a manual modification of the design parameters and an evaluation of the results by a full flow field solution. After a number of systematic design alterations, further improvement cannot be achieved.

Thus, only the integration of a numerical optimization algorithm in the design process has proven to be a more efficient method in searching a final optimum solution. But due to the big number of design parameters and the required high accuracy of the Navier-Stokes code used for the flow computation, the numerical optimization results in a huge amount of computational costs.

To reduce CPU time, two different approaches may be used. The first alternative is to decrease the number of free design parameters by a selective parametrization of the complex geometry. The second is to save CPU time by using faster CFD tools yielding sufficiently accurate results.

To reduce the numerical effort, a new optimization concept based on a multilevel CFD technique has been developed. The main goal of this new idea is to start the optimization process on a low CFD level using a calibrated and very fast computing quasi-3D Euler code (EQ3D) being 100 times faster than a 3D Navier-Stokes code (NS3D). Thus, a very fast and relatively accurate flow prediction is obtained on the first optimization level.

Then, the numerical optimization is carried out on the second level applying a 3D Euler code (E3D) using unstructured meshes which yields the 3D structure of the flow through a blading. The E3D solutions obtained on a very coarse computational mesh are very close to those generated by an NS3D code, but the E3D computations are about 10 times faster compared to the NS3D solutions which need a much more refined mesh to predict the losses accurately.

On the third level of optimization, the most accurate but very time consuming tool, a Navier-Stokes code, is applied to optimize the shape of a blading with respect to minimum losses. Since the optimization process usually reaches a very high maturity after having passed the first and second level of optimization, only a few NS3D computations are needed for the fine-tuning of the blade shape.

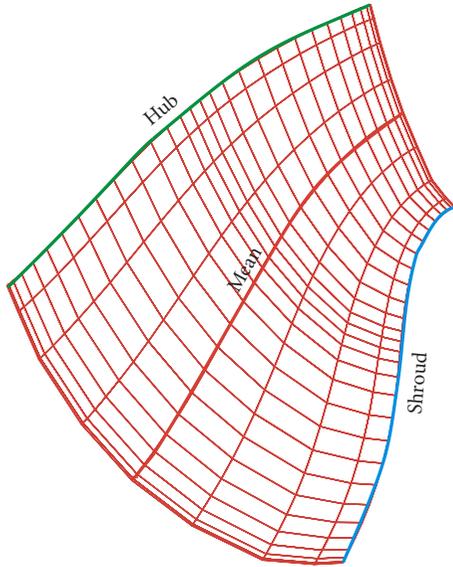


FIGURE 1: B-spline representation of a camber surface.

This multilevel CFD technique is integrated in a fully automatic design optimization process. Depending on the nature of the problem, different optimization algorithms can be applied to solve the optimization problem with the objective functions and constraints, which are defined by mathematical functions.

## 2. PARAMETRIZATION OF THE GEOMETRY

The 3D geometry of rotodynamic pumps and water turbines is described by superimposing a camber surface, see Figure 1, with the thickness distributions defined on a given number of geometrical surfaces of revolution, see Figure 2, yielding much more than 100 parameters.

The camber surface, the thickness distribution, and the meridional contour, see Figure 3, are approximated by means of B-splines in which the coordinates of the describers are the design parameters of the blading. With augmenting level of optimization, the number of design parameter is increased.

The modification of the camber surface is performed in the conformal mapping representation on a given number of stream surfaces. Using this 2D representation of the blade camber surface instead of a 3D representation, the number of design parameters is reduced remarkably.

Since the CPU time for an optimization run increases dramatically with the number of free design parameters, a minimum number of describers are used to approximate the 3D blade geometry.

## 3. NUMERICAL TOOLS

### 3.1. CFD codes

For the analysis of the flow field and the evaluation of the total losses or the hydraulic efficiency, respectively, three levels of CFD codes are used.

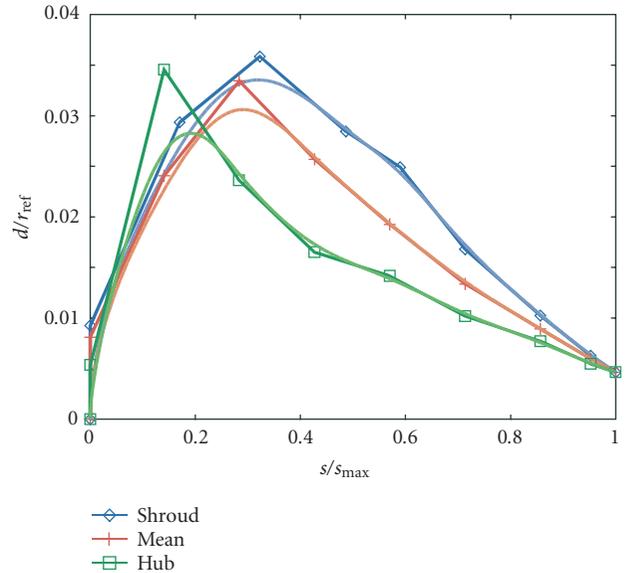


FIGURE 2: Thickness distribution along three surfaces of revolution.

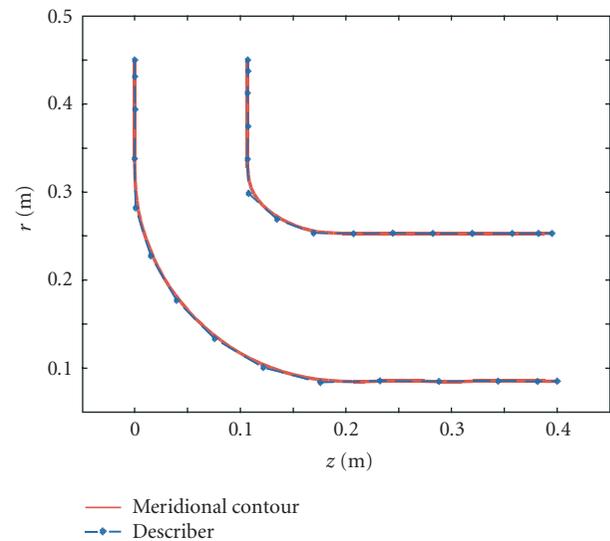


FIGURE 3: Meridional contour.

### Quasi-3D Euler code (EQ3D)

Applying the theory of Wu [1], the 3D inviscid but rotational flow through the bladings of hydraulic machinery may be approximated by iterating between a number  $N1$  of blade-to-blade flows on  $S1_i$ -surfaces, a number  $N2$  of hub-to-shroud flows on  $S2_j$ -surfaces, and a number  $N3$  of secondary flows on  $S3_k$ -surfaces. Since the numerical effort to solve this system of equations is very big, a simplified but very efficient model is used by superimposing a number  $N1$  of flows on  $S1_i$ -surfaces being surfaces of revolution and the flow on a representative  $S2_m$ -surface, see Schilling [2].

This EQ3D code yields the velocity and pressure distributions along hub and shroud as well as along the pressure and suction side of the blades which are used as input for an analytical loss prediction module, see Schilling [3].

#### Full-3D Euler code (E3D)

In order to describe the 3D character of the flow through the bladings especially the secondary flow induced by the blade forces more precisely a fast computing E3D code has been developed using unstructured meshes and a multigrid scheme to reach a speedup of the convergence behaviour. The basic equations to be solved are the continuity equation and the momentum equations including the viscous terms needed for the stabilization of the numerical solution procedure. The transport equations are solved by means of a second-order finite-volume scheme using a higher-order upwind scheme for the approximation of the convective terms and a second-order approximation for the viscous terms. To satisfy the conservation law of mass in differential form, the SIMPLE algorithm is applied successfully. To check the influence of the viscous effect on the numerical solution, the runner Reynolds-number  $Re$ , defined by the tip circumferential speed, the diameter, and the kinematic viscosity, has been modified within a wide range. It has been found that beyond  $Re \cong 500$  the integral quantities, for example, the energy coefficient, and the flow field quantities, for example, the velocity and pressure distribution, are almost independent of the  $Re$ -number.

Thus, the Navier-Stokes solution for the big  $Re$ -numbers may be interpreted as an approximate Euler solution supposing Euler boundary conditions along the walls.

This type of Euler solution yields in the same way as described above the velocity and pressure distributions being used for an analytical loss analysis.

#### 3D Navier Stokes code

For the viscous analysis of the 3D turbulent flow through hydraulic machinery, a second-order and block-structured Navier-Stokes code using structured meshes has been developed based on an existing cell-centered finite-volume code of Ritzinger [9]. Furthermore, a higher-resolution upwind scheme and an efficient parallelization using the standard software MPI have been implemented. Mass conservation is achieved by using a SIMPLE pressure correction scheme [8]. A standard  $k$ - $\epsilon$ -model with log-law wall functions is used to calculate the turbulent viscosity  $\nu_t$ .

Both the convective and the viscous fluxes are approximated using a deferred correction scheme. The system of equation is solved by means of the SIP of Stone. For the calculation of stages of hydraulic machinery, a coupling algorithm is used being both momentum and mass conservative. This code has been validated by comparing the predicted flow field data and performance curves with measurements carried out at the TU Darmstadt, see [10, 11, 12]. The numerical and experimental results coincide fairly well. Thus, the real flow through hydraulic machinery bladings may be simulated with a high accuracy and reliability.

Referring the numerical effort of the NS3D and E3D codes to the effort of the EQ3D code needed for one flow computation, the CPU times show the relation 100 : 10 : 1 approximately.

### 3.2. Optimization algorithms

The general model of nonlinear programming is to minimize an objective function  $F(\vec{x})$  subject to  $m$  nonlinear equality and  $p$  inequality constraints, see Eschenauer et al. [4]. Therefore, the following mathematical nonlinear programming problem is defined as follows:

$$\begin{aligned} \min F(\vec{x}), \quad \vec{x} \in \mathfrak{R}^n, \\ g_i(\vec{x}) \leq 0, \quad i = 1, \dots, m, \quad \vec{x} \in \mathfrak{R}^n, \\ h_j(\vec{x}) = 0, \quad j = 1, \dots, p, \quad \vec{x} \in \mathfrak{R}^n. \end{aligned} \quad (1)$$

The numerical optimization of a hydraulic machinery blading is carried out by algorithms for constrained nonlinear optimization. The objective functions are evaluated from the results of a numerical flow computation. Because these results are generated numerically, rough objective functions are obtained. This fact has to be taken into account selecting the suitable optimization algorithm.

Three different algorithms have been implemented in the developed optimization system.

The Hooke-Jeeves algorithm uses a deterministic pattern search method which is widely used for nonsmooth objective functions due to its simplicity and robustness. The Hooke-Jeeves strategy consists of two basic moves, the exploratory and the pattern move using constant step sizes, see [5].

The DONLP2 works with a sequential equality constrained quadratic programming method with an improved Armijo-type stepsize algorithm [6]. To evaluate the first partial derivatives of the objective function, finite differences are used.

The FSQP algorithm is based on sequential quadratic programming modified so as to generate feasible iterates that satisfy all constraints. The objective function can be minimized by an Armijo-type arc search or with a nonmonotone line search, see [7]. The gradients of the objective functions may be calculated by forward finite differences.

A comparison between the pattern search (PS) algorithm of Hooke and Jeeves and the Gradient-based FSQP algorithm applied to find the minimum minimum of the 2D Rosenbrock function with the two optimization parameters  $x_1$  and  $x_2$ ,

$$F(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2, \quad (2)$$

clearly shows that the FSQP is much more efficient than the PS, see Figure 4. The Hooke-Jeeves algorithm needs 451 iterations while the FSQP is needing only 92 steps.

## 4. MLCFD STRATEGY

The basic idea of the multilevel CFD strategy is to approach the global optimum of the problem to be solved on

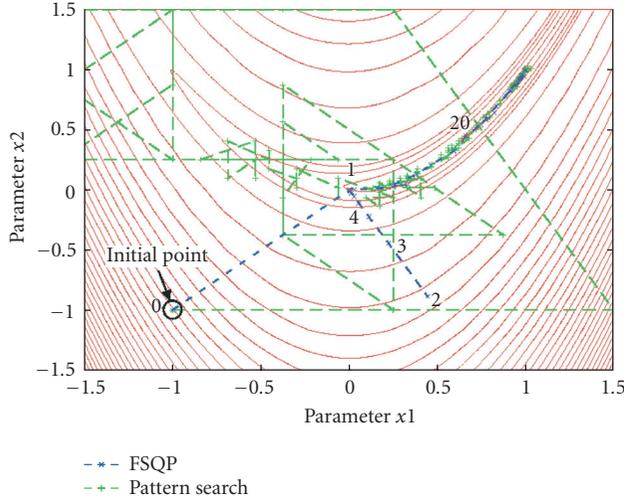


FIGURE 4: Comparison of the FSQP and the Hooke-Jeeves optimization algorithm applied to the 2D Rosenbrock function.

different levels of CFD techniques, that is, levels of accuracy reached and CPU time needed. With augmenting optimization level the deviation of the computed compared to the real fluid flow becomes smaller and the numerical effort bigger.

Thus, most of the optimization work can be done on the first level applying the very fast computing EQ3D code and reaching a relatively good approach. The solution obtained is improved on the second level by using the E3D code. Here, a much less number of iterations are carried out yielding a geometry which is relatively close to the final one. Therefore, only a very few Navier-Stokes iterations are needed on the third and final level to find the minimum minimum, that is, the optimum geometry and the minimum losses within the given constraints.

## 5. DEFINITION OF THE OBJECTIVE FUNCTIONS

Optimizing the runner of a water turbine, a weighted summation formulation of objective functions like head, efficiency, deviation from prescribed velocity and pressure distributions, and smooth pressure distribution and cavitation criterion may be defined.

The vector of the objective functions is converted into a scalar function  $F_p$  using a linear combination of the individual objective functions  $F_i$ :

$$F_p(F(\vec{x})) = \sum_{i=1}^m \sigma_i F_i(\vec{x}), \quad \vec{x} \in \mathfrak{R}^n. \quad (3)$$

Three definitions of individual objective functions are listed below.

(1) *Runner head.* The first optimization task is to minimize the deviation of the calculated runner head  $H_{RC}$  from

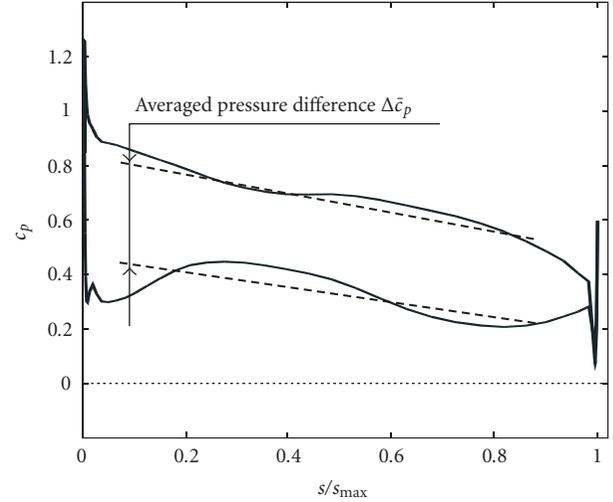


FIGURE 5: Smooth pressure distribution.

the design runner head  $H_R$ :

$$F_H(\vec{x}) = \frac{|H_{RC} - H_R|}{H_R}. \quad (4)$$

(2) *Deviation from a prescribed velocity distribution.* The second objective function evaluates the deviation of the calculated velocity profile from the prescribed one:

$$F_{c_i}(\vec{x}) = \frac{\sqrt{(1/N) \sum_{j=1}^N (c_{i_{req},j} - c_{i,j})^2}}{(1/N) \sum_{j=1}^N c_{i_{req},j}}. \quad (5)$$

(3) *Smooth pressure distribution.* For a constant loading of the blade, the optimization task is the minimization of the deviations of the local and the averaged pressure differences along the surfaces of revolution as it is shown in Figure 5.

With the pressure difference of pressure and suction side

$$\Delta c_{p,i} = c_{p,PS,i} - c_{p,SS,i}, \quad (6)$$

an averaged pressure difference is evaluated as

$$\Delta \bar{c}_p = \frac{1}{i_{\max}} \sum_{i=1}^{i_{\max}} \Delta c_{p,i} \quad (7)$$

and with

$$\Delta s_i = s_i - s_{i-1}, \quad (8)$$

the objective function is defined as

$$F_{c_p}(\vec{x}) = \frac{1}{N} \sum_{j=1}^N \sqrt{\frac{\sum_{i=1}^{i_{\max}} \Delta s_i \cdot (\Delta \bar{c}_{p,j} - \Delta c_{p,i,j})^2}{s_{\max,j}}}. \quad (9)$$

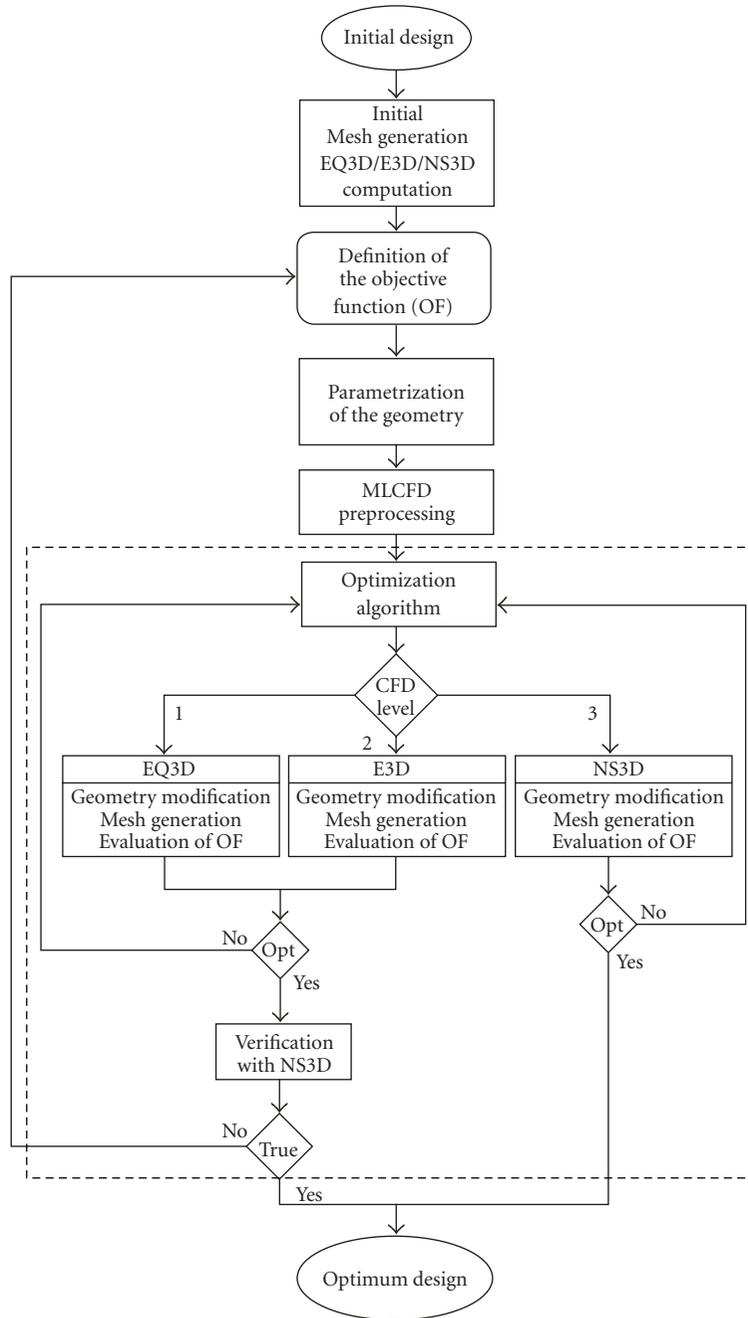


FIGURE 6: Optimization system.

### 6. OPTIMIZATION SYSTEM

The structure of the developed optimization system is shown in Figure 6. After performing the flow field solutions of the initial design on all three CFD levels, the user has to set up his optimization problem by defining the objective functions and constraints. The objective functions are directly coupled with the parametrization of the geometry, yielding the free

design parameters, and the CFD level on which the optimization is realized. The multilevel CFD preprocessing tool calibrates the EQ3D and E3D codes based on the initial NS3D calculation.

Then, the optimization loop is entered. First, the optimization algorithm calculates the vector of new design parameters and the geometry module generates the new meridional geometry, the corresponding new blading, and the new

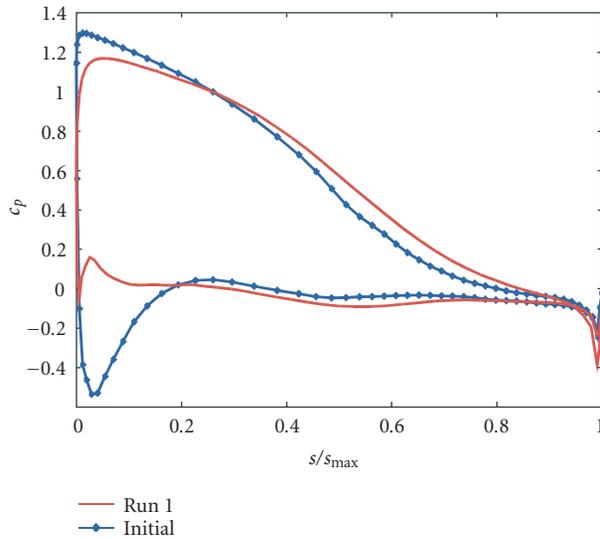


FIGURE 7: Pressure distribution at the shroud of the runner.

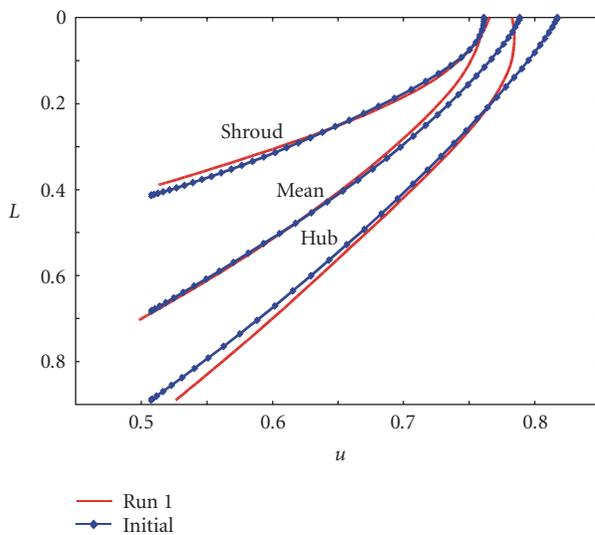


FIGURE 8: Conformal mapping representation of the runner blade for the initial design and the first optimization run.

numerical mesh. Then, the flow simulation is performed and the objective functions are evaluated by using the results of the evaluation. In the next step, the optimization algorithm calculates the new vector of design parameters depending on the latest values of the objective function. This procedure is repeated until a convergence criterion is satisfied. One of the most critical parts of the optimization system is the mesh generation for the CFD analysis. Thus, it has to be ensured that a numerical mesh with a required minimum quality for any geometry can be generated during the optimization process.

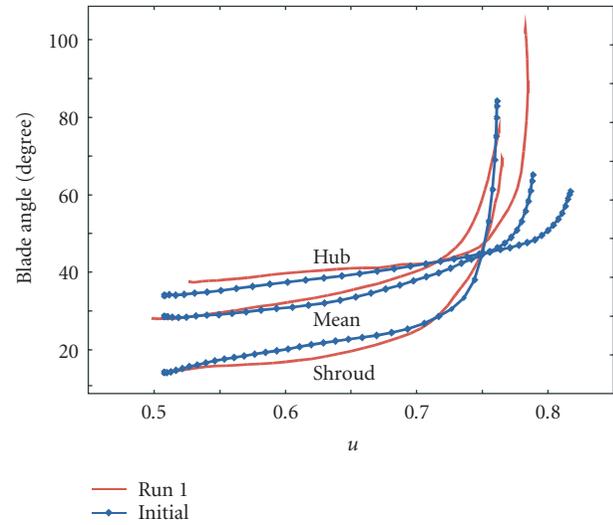


FIGURE 9: Blade angle distribution of the runner blade for the initial design and the first optimization run.

After having passed the optimization runs only on the first and second levels, a verification of the final design has to be carried out by an NS3D recalculation to guarantee that the obtained solution satisfies all requirements. If yes, the next optimization level may be chosen.

Otherwise, the optimization process has to be repeated by defining a modified objective function and a new set of parameters.

## 7. RESULTS

The described optimization system has been applied to optimize a Francis turbine runner having a specific speed  $n_q = 41$  1/min using the MLCFD technique.

Figures 7, 8, and 9 show the results of the first optimization run with respect to a prescribed pressure distribution, along the shroud of the Francis runner, the conformal mapping representation and the blade angle distribution. However, the computed runner head  $H_{RC} = 86.5$  m exceeds the design head  $H_R = 81.8$  m remarkably. Regardless, the hydraulic efficiency of the runner could be improved from the initial value  $\eta_{h0} = 95.6\%$  about  $\Delta\eta = 0.8\%$ .

In a second run, the runner geometry has been optimized to reach the design head. Figure 10 shows that the outlet circumferential velocities could be adapted much better to the required distribution yielding a computed head  $H_{RC} = 82.5$  m being much closer to the required value. In Figures 11 and 12, the conformal mapping representation and the blade angle distribution of the first and second optimization runs are shown. The second run yields an additional improvement of the hydraulic efficiency of  $\Delta\eta = 0.3\%$ , so that an optimum value  $\eta_h = 96.7\%$  is reached. Since NS3D codes are underpredicting the turbine efficiencies the final optimization result is well promising.

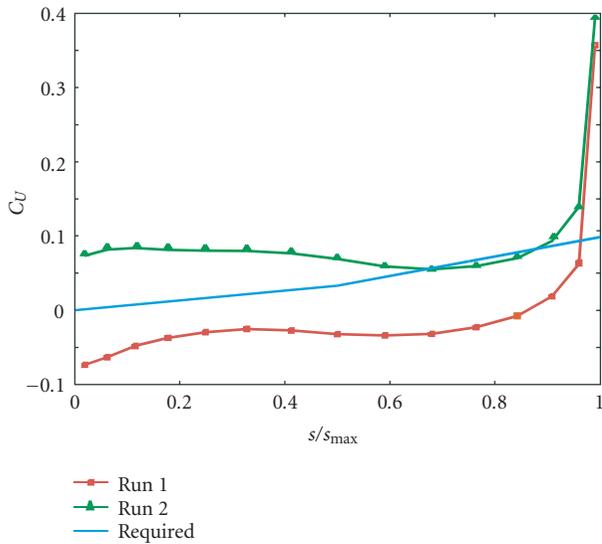


FIGURE 10: Outlet circumferential velocity distribution for the first and second optimization runs compared with the required distribution.

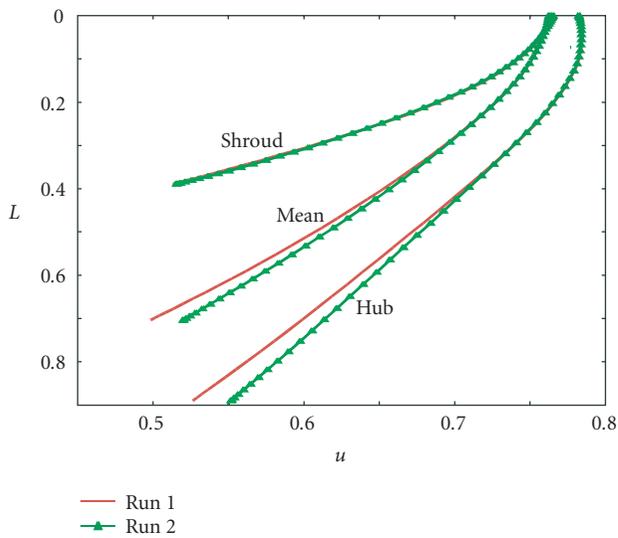


FIGURE 11: Conformal mapping representation of the runner blade for the first and second optimization runs.

### 8. CONCLUSIONS

A Multi Level CFD technique has been developed and integrated in a fully automatic optimization process.

The optimization was performed on the lowest CFD level, with the EQ3D code, which yields a blade geometry with much better flow behaviour. Only on the final optimization level the most powerful but very time-consuming NS3D code is applied for the fine-tuning of the blade shape with respect to minimum losses.

Thus, the multilevel CFD technique has proven to be an efficient engineering tool for the optimum design of hydraulic machinery bladings.

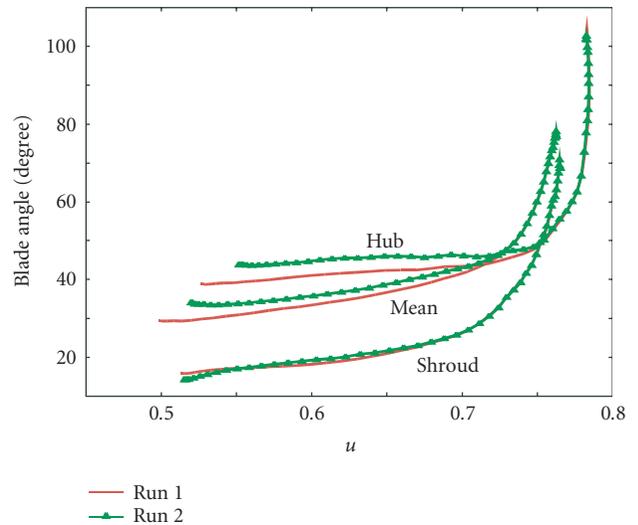


FIGURE 12: Blade angle distribution of the runner blade for the first and the second optimization runs.

### REFERENCES

- [1] C.-H. Wu, A general theory of three-dimensional flow in subsonic and supersonic turbomachines of axial, radial and mixed-flow types, NACA, TN 2604, 1952.
- [2] R. Schilling, "Numerical calculation of the Q3D incompressible, inviscid flow in turbomachines," in *11th IAHR Symposium Amsterdam*, Amsterdam, the Netherlands, September 1982.
- [3] R. Schilling, Modellierung der Reibungsverluste in Beschauelungen hydraulischer Strömungsmaschinen, LHMinterner Bericht, 2001.
- [4] H. Eschenauer, J. Koski, and A. Osyczka, *Multicriteria Design Optimization, Procedures and Applications*, Springer, Berlin, Germany, 1990.
- [5] R. Hooke and T. A. Jeeves, "Direct search solution of numerical and statistical problems," *Journal of the ACM*, vol. 8, no. 2, pp. 212–229, 1961.
- [6] P. Spellucci, "An SQP method for general nonlinear programs using only equality constrained subproblems," *Mathematical Programming*, vol. 82, no. 3, pp. 413–448, 1998.
- [7] C. T. Lawrence, J. L. Zhou, and A. L. Tits, User's Guide for CF-SQP Version 2.3: A C Code for Solving Constrained Nonlinear Optimization Problems, Generating Iterates Satisfying all Inequality Constraints, Institute for Systems Research, College Park, Md, USA, 1995.
- [8] J. H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, Springer, Berlin, Germany, 1996.
- [9] S. Ritzinger, *Simulation realer Laufradströmungen*, Ph.D. Dissertation, Technische Universität München, München, Germany, 1996.
- [10] R. Schilling, S. Ritzinger, B. Stoffel, and J. Hambrecht, "Numerical and experimental flow analysis in centrifugal pump impellers," in *Proc. International Conference on Fluid Engineering (ICFE '97)*, vol. 1, pp. 207–211, Tokyo, Japan, July 1997.
- [11] S. Skoda, R. Richter, and R. Schilling, VDMA-Forschungsvorhaben: Entwicklung und Kalibrierung eines 3D Navier-Stokes Codes zur Nachrechnung von beliebigen Kreiselpumpenstufen; numerischer Teil, 2000.
- [12] W. Aysheshim and B. Stoffel, VDMA-Forschungsvorhaben: Entwicklung und Kalibrierung eines 3D Navier-Stokes Codes zur Nachrechnung von beliebigen Kreiselpumpenstufen; experimenteller Teil, 2000.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

