AUTOMATED CFD FLOW OPTIMISATION OF AN AXIAL EXPANSION TURBINE FOR ENERGY RECOVERY FROM WORKING FLUIDS

Andreas NEIPP*

Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart, Germany

Stefan RIEDELBAUCH

Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart, Germany

ABSTRACT

This paper deals with the blade design of an axial expansion turbine for energy recovery from working fluids using an automated optimisation process that applies full 3D CFD simulations. In order to reduce the total calculation time not only the optimisation process, but also the geometry generation, the mesh generation as well as the CFD processing operates in parallel. The different jobs are administrated by a Perl-interface which coordinates the work flow and processes the data. An in-house optimisation algorithm performs the optimisation procedure. A blade parameterisation is introduced and described in detail. As quality criterion the average efficiency is combined with the standard deviation of the efficiencies of three different operating points. Additionally different boundary values are introduced to investigate their influence on the blade design and the flow behaviour. The optimisation results show that the use of two combined boundary values improves the cavitation behaviour of the turbine significantly. The average efficiency of the axial expansion turbine could be increased without narrowing the operating range.

KEYWORDS

Axial expansion turbine, blade parameterisation, automated flow optimisation

1. INTRODUCTION

Many industrial processes need working fluids on high pressure levels. At the end of processes fluids are often released to ambient pressure via conventional throttles. The pressure energy dissipates and remains unused. An axial expansion turbine (AXENT) is one possibility to convert the pressure energy into electrical energy. The reuse of recovered energy increases the overall efficiency of systems.

The key requirement on recovering systems is that the energy recovery has no negative effect on the plant safety and the process stability. A significant advantage of the axial expansion turbine is the almost constant flow rate even in the case of power failure, when the turbine reaches its runaway speed. The constant flow rate prevents a hydraulic pressure surge, which could result in damages on valves, sealings and other structures of the system. The AXENT needs no additional safety device or auxiliary control equipment [1]. The future success of the AXENT depends to a large extent on the efficiency and the durability.

The turbine efficiency shall be increased without narrowing the operating range by having a negative influence on the cavitation behaviour. A previous manual optimisation proved to be extremely time-consuming, because even slight changes in blade shape have strong influence on the flow behaviour and operating point. The counteracting effect observed during the manual optimisation was that an increase in efficiency comes along with worse cavitation behaviour finally needing additional design time. In order to accelerate the design process an

* *Corresponding author:* Institute of Fluid Mechanics and Hydraulic Machinery, Pfaffenwaldring 10, 70550 Stuttgart, Germany, phone: +49(0)711/685-60231, email: neipp@ihs.uni-stuttgart.de

optimiser based on a genetic algorithm is used [2]. The automated CFD flow optimisation is performed on the basis of a manual optimised reference design. The

2. OPTIMISATION PROCESS

The whole optimisation process can be divided into three processing levels. The top level is the optimiser calculating the parameters of each individual in a generation. The next level is a Perl-interface coordinating the work flow. The third level consists of three successively executed jobs, the geometry generation, the mesh generation and the CFD computation. The work flow of one generation can be divided into three major steps. In the first step the design is parametrized and the geometry generation is started. The second step consists of evaluating output data and distributing work. Therefore the geometries and the meshes are categorised into fit or failed designs. Depending on the categorisation the designs will be passed to the next job or will be separated from the regular work flow, as shown in figure 1. The third step is the solutions processing. One single quality criterion for each design is calculated and passed to the optimiser. Hereafter all output data of one generation are cleared.

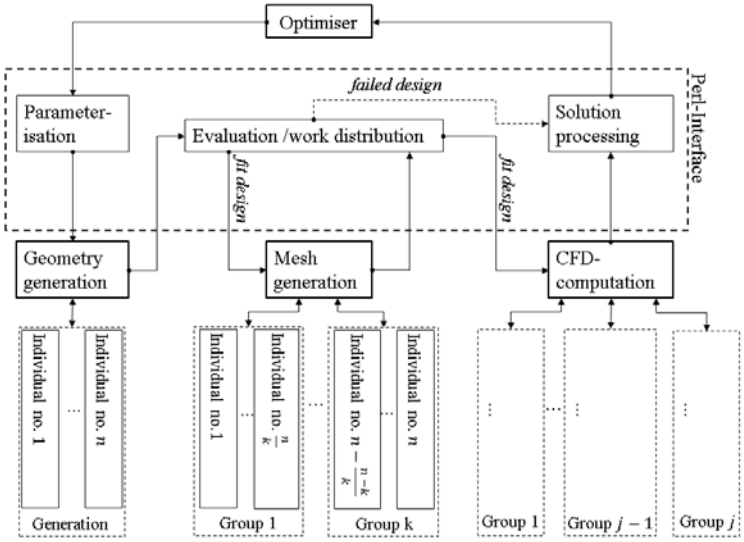

Fig.1 Optimisation procedure

Every individual of a generation has to be executed in the same way and follow the same workflow. The serial execution of the geometry generation, the mesh generation and the CFD-computation allows that the number of parallel executed individuals can differ between those three jobs. Depending on the computing capacity and the amount of licences available the number of individuals treated in parallel may be adapted. In case, for example, that the number of individuals in one generation is higher than the number of available licences the generation can be divided into groups being sequentially executed.

The parallel running processes are called child processes. By using the fork system call every child process is writing a solution with its PID number into a container file. For a robust optimisation procedure not every child process has to finish successfully. If one child process is terminated due to any computational or network error it returns a predefined trivial solution to the container file. The trivial solution is treated as a failed design. After all child processes have finished the container file is processed by the parent process and passes the data to the Perl-interface for evaluation or solution processing.

The optimisation algorithm

The used in-house optimisation algorithm is based on a genetic procedure, which uses a predefined number of individuals per generation [3], [4]. Each individual represents one

design with a given parameter setup. After the computation of one generation the solution process starts and calculates one single target value for each individual, e.g. the turbine efficiency. The optimiser retrieves the data to sort out all bad designs and calculates a new set of parameters for each individual of the next generation. This complies with the evolutionary principle of "surviving of the fittest". Individuals that are sorted out by the evaluation module because of a failed geometry generation or a bad mesh quality are transferred directly to the solution processing and will always be treated as failed designs.
The following section shortly outlines the work sequence of the optimiser. First the number of parameters, the lower and upper value of each parameter, the number of individuals n_{ind}, the number of surviving individuals n_{sur}, the maximum number of generations and a convergence criterion have to be defined. In the first step the optimiser randomly calculates a set of parameters in the allowed range of values for n_{ind} individuals of the first generation. After processing all individuals, the optimiser retrieves the solutions of each individuals and sorts out the best n_{sur} surviving individuals building the basis for the next generation with $n = n_{ind} - n_{sur}$ individuals. In the case that the defined convergence criteria is met the optimiser stops.
The optimiser is based on three different methods for setting up new parameters. The method of reproduction, two combined simplex algorithm and a random function are used. Each method provides one third of the new individuals. The method of reproduction and the simplex algorithm generate the new designs on basis of surviving individuals whereas the random function generates its designs totally stochastically and as a result reduces the risk of finding a local maximum.

Blade parameterisation

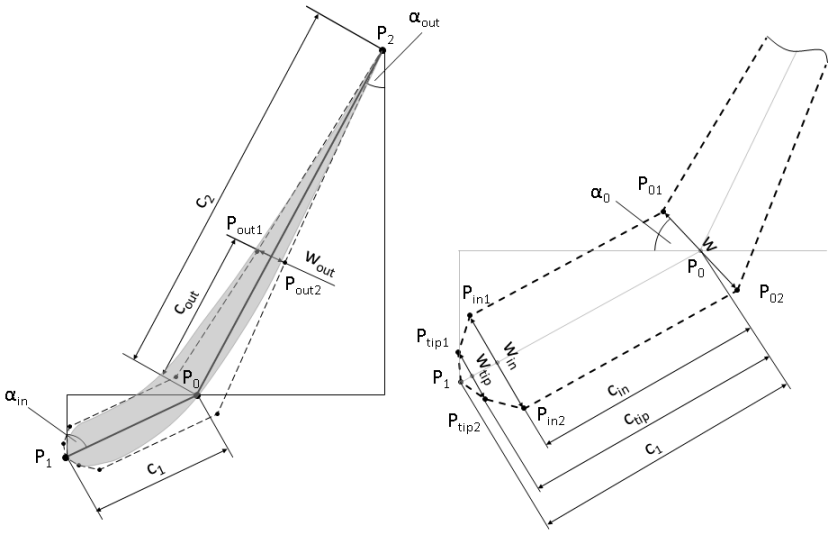

Fig.2 Blade parameterisation

The parameterisation of the blade shape is based on a polygon constructed from ten points (Fig. 2). The blade profile is modelled by a B-spline formulation of fourth order. The corners of the polygon are treated as control points. The length, the thickness distribution as well as the inlet and outlet angle of the blade is defined by the shape of the polygon.
The blade is described by splines in three radial cuts. The cuts are placed on the hub, in the middle of the blade channel and on the shroud. The blade surface is built by putting a lofted surface over the three splines. The blade is constructed out of 30 points and consequently the parameterisation of the blade has 60 degrees of freedom. In order to simplify the optimisation process, with regard to the geometry and the mesh generation, the degrees of freedom for one blade were reduced to a maximum of 33. The following section explains how the points were set into relationship to each other.

The construction of the polygon is initially based on the point of origin P_0 and two additional points P_1 and P_2. The coordinates of those two points are defined by two angles α_{in} and α_{out}, a length ratio x_{t0} and the combined length $c = c_1 + c_2$ of two straight lines connecting the three points (Fig. 2: left):

$$\begin{matrix} x_0 = 0 \\ y_0 = 0 \end{matrix}; \quad \begin{matrix} x_1 = -\sin(\alpha_{in}) \cdot x_{t0} \cdot c \\ y_1 = -\cos(\alpha_{in}) \cdot x_{t0} \cdot c \end{matrix}; \quad \begin{matrix} x_2 = \sin(\alpha_{out}) \cdot (1 - x_{t0}) \cdot c \\ y_2 = \cos(\alpha_{out}) \cdot (1 - x_{t0}) \cdot c \end{matrix} \tag{1}$$

The width w of the polygon at P_0 is defined by two points P_{01} and P_{02}. The coordinates are calculated with the length c, a length width ratio t and an angle α_0.

$$\begin{matrix} x_{01} = -\sin(\alpha_0) \cdot c \cdot \frac{t}{2} \\ y_{01} = \cos(\alpha_0) \cdot c \cdot \frac{t}{2} \end{matrix}; \quad \begin{matrix} x_{02} = \sin(\alpha_0) \cdot c \cdot \frac{t}{2} \\ y_{02} = -\cos(\alpha_0) \cdot c \cdot \frac{t}{2} \end{matrix}; \quad \alpha_0 = \frac{\alpha_{in} + \alpha_{out}}{2} \tag{2}$$

The two straight lines connecting P_1 and P_2 with P_0 form symmetry axes. The remaining six points of the polygon are symmetrically arranged in pairs. Figure 2 shows the position of those points being described by a distance c_i and a width w_i. The coordinates of each point are calculated with the length c, the two angles α_{in} and α_{out}, two length ratios x_{t0} and x_i and a length width ratio t_i.

$$\begin{aligned} x_{i1,i2} &= \left(-\sin(\alpha_{in}) \cdot x_{t0} \cdot (1 - x_i) \mp \cos(\alpha_{in}) \cdot \frac{t_i}{2} \right) \cdot c \\ y_{i1,i2} &= \left(-\cos(\alpha_{in}) \cdot x_{t0} \cdot (1 - x_i) \pm \sin(\alpha_{in}) \cdot \frac{t_i}{2} \right) \cdot c \\ x_{out1,out2} &= \left(\sin(\alpha_{out}) \cdot (1 - x_{t0}) \cdot x_{out} \mp \cos(\alpha_{out}) \cdot \frac{t_{out}}{2} \right) \cdot c \\ y_{out1,out2} &= \left(\cos(\alpha_{out}) \cdot (1 - x_{t0}) \cdot x_{out} \pm \sin(\alpha_{out}) \cdot \frac{t_{out}}{2} \right) \cdot c \end{aligned} \tag{3}$$

The blade tip is very important for the flow behaviour in the blade channel. The flow very sensitively reacts on small changes in their shaping. For that reason the five points defining the shape of the blade tip are put close together. The remaining points at the end of the blade have a greater distance to generate a uniform outflow.

Due to the spline definition both angles α_{in} and α_{out} do not represent the inlet and outlet angle of the blade. Except for P2 which is the starting and ending point of the spline all other control points are not located on the spline. The blade shape is exactly defined by the 33 parameters defining the coordinates of the polygon corners. However, this current geometry definition does not allow to directly predefine an actual value for the chord length as well as the in- and outlet angle of the blade. Those geometry values are indirectly defined through the final spline generation.

The parameterisation and the actual geometry generation are done with an in-house code [5]. For this purpose the equations for the parameterisation as well as the B-splines and the corresponding surfaces have to be predefined in XML files and referred to a part of the geometry e.g. the hub or the blade. At the end a name is assigned to each part and the geometry information is exported as STEP (STP) file.

Mesh generation

The most important requirement for a mesh generated during an automated blade optimisation is an insensitive reaction of the mesh quality to changes in blade geometry. A further requirement is that the generated mesh has a sufficient quality and the smallest possible

number of nodes. For this reason the domain consists of one blade channel with a periodical 1:1 mesh coupling and a simple block structured mesh.
Generally two different methods can be distinguished for blocking one single blade channel. The first method is the inline blocking, where the blade is surrounded by blocks. Periodic interfaces that cover the whole length of the blade channel have to be defined. The second method is the blade-to-blade blocking. The blocks are located between two blades in the channel. The channel partition runs through the blades and hence the periodic surfaces for the flow are much smaller. Previous investigations showed that smaller periodic interfaces lead to a greater flexibility in placing the blocks. Therefore a blade-to-blade mesh is more insensitive to geometry changes than an inline mesh. For that reason a blade-to-blade mesh is used.
The automated mesh generation is done with a replay script executed in ICEM CFD. In the first step the geometry is imported from several STP files each containing the geometry information of one part. In the next step the blocking is generated and all associations are set. Afterwards the mesh is loaded from the blocking and a mesh report is written. This report enables to evaluate the mesh quality. In the last step the mesh is exported.

Evaluation and work distribution
The evaluation of the geometry and the mesh generation is an essential requirement for a robust optimisation process. At first the module checks if all STP files of a design were generated. In case of a positive evaluation the design is rated fit and passed on to the mesh generation. If the geometry generation failed for any reason, a trivial solution is passed on to the solution processing and the individual is treated as failed design (Fig. 1).
After all fit designs have run through the mesh generation the second evaluation run starts. First of all the existence of the mesh file and the mesh report file of all executed individuals is checked. If both files exist the evaluation module reads out the mesh quality from the mesh report file. The mesh quality has to be equal to or greater than the predefined minimum mesh quality. The individuals fulfilling the defined requirement are passed to the CFD computing. The others are treated as failed designs (Fig. 1).

CFD computation
The CFD computation can be divided into three basic operations the pre-processing, the solver run and the post-processing. To guarantee the comparability of the simulation results the actions performed during the CFX-Pre and the CFD-Post session were recorded with ANSYS CFX and saved in session files. A bash script is used to successively start the three operations in batch mode. Depending on the available licences and computing capacities a predefined number of designs can be treated parallel.
The pre-processing including the definition of the solver input file is very simple because the only thing that varies is the mesh. All relevant physics and material information are predefined in a CFX command language (CCL) file. The command line starting the CFX-Solver defines all settings concerning the solver run. During the post-processing three output values are exported to a text (txt) file, the turbine efficiency η, Eq.(5), the minimum static pressure $p_{st,min}$ Eq.(6) and the volume v_{cav} in which cavitation is likely to occur. The described CFD computation is simultaneously executed for three different turbine operating points, such as part load, best point and full load. The underlying idea is to optimise the flow behaviour of the turbine over the whole operating range.

Solution processing
The solution processing is the last decisive interface before the quality criterion of each individual in a generation is passed on to the optimiser. As mentioned before the trivial solutions sent from the evaluation module are treated as failed designs. The same applies to

designs where the CFD computation has terminated and no result file could be generated. The simulation results of all remaining individuals are processed. Based on predefined boundary values, such as minimum static pressure $p_{st,min}$ Eq.(6) and cavitation volume v_{cav}, the designs are again categorized. Fit designs that comply with the predefined boundary values are rated by an objective function f_t. For that reason the efficiency of the turbine is calculated by using Eq. (4) where M is the torque, ω the angular velocity, Δp the pressure difference between turbine in- and outlet and Q the discharge.

$$\eta = \frac{M \cdot \omega}{\Delta p \cdot Q} \ . \tag{4}$$

The target function is a combination of the average turbine efficiency η_{ave} and the standard deviation s_{dev} of the efficiencies in the three operating points. The optimisation goal is to increase the efficiency in all three operating points and to reduce the deviation leading to a flat characteristic curve on a higher efficiency level. Because the optimiser searches for the global minimum the objective function is defined as $f_t = -(\eta_{ave} - s_{dev})$ with

$$\eta_{ave} = \frac{\eta_{op1} + \eta_{op2} + \eta_{op3}}{3}$$
$$s_{dev} = \sqrt{\frac{1}{2}\left((\eta_{op1} - \eta_{ave})^2 + (\eta_{op2} - \eta_{ave})^2 + (\eta_{op3} - \eta_{ave})^2\right)}. \tag{5}$$

3. THE OPTIMISATION

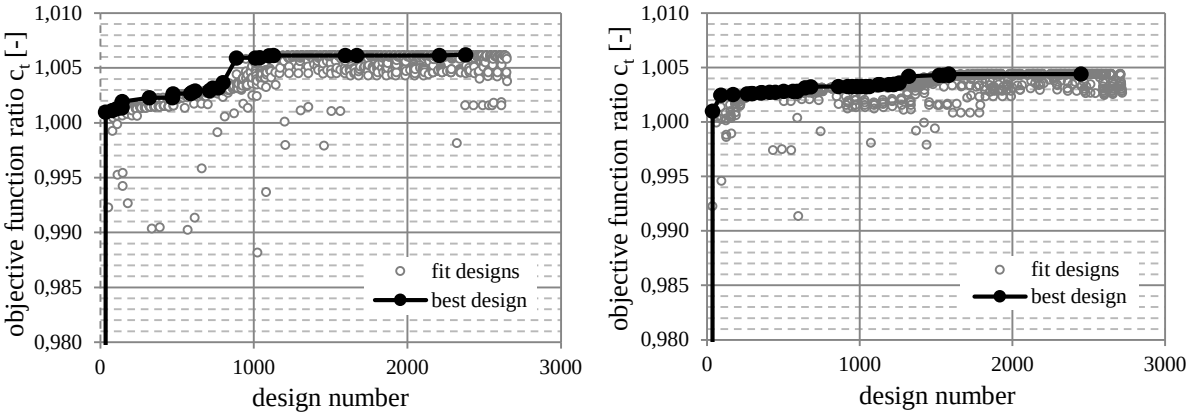

Fig.3: Objective function ratio over design number for run 1 (left graph) and run 2 (right graph)

In this paper two optimisation runs are presented. The only difference between the two runs is the number of output values used for categorisation during solution processing. The two optional boundary values are the volume v_{cav} and the minimum static pressure $p_{st,min}$. Figure 3 shows the objective function ratio $c_t = f_t / f_{t,ref}$ of the optimised design to the reference design as function of the design number for both runs. The first optimisation run uses the volume v_{cav} as boundary value. The second run has the minimum static pressure $p_{st,min}$ as additional second boundary value. Because of time restrictions both runs were terminated before reaching the predefined convergence criterion. With the help of the graphs it can be assumed that no further increase of c_t will happen for either of the runs.

In run one (left graph) a total of 2658 individuals were generated. About 58 %, i.e.1545 individuals were categorized as failed designs with a value of $f_t = 0$ either by the evaluation module or during the solution process. Even a physically correct design can be marked as failed design if the simulation results do not comply with the defined limits for the minimum static pressure or the cavitation volume. For that reason boundary values have a great influence on the number of surviving individuals and the convergence behaviour. By

comparing the two graphs it can be seen, that an optimisation run with more boundary values has a smaller number of surviving individuals. As a result the optimiser has to generate more individuals to find the best design.
In order to examine the possible cavitation behaviour the minimal static pressure $p_{st,min}$ in the flow channel is evaluated. A minimum static pressure

$$p_{st,\min} < p_{at} - p_v \tag{6}$$

where p_{at} = 101300 *bar* is the atmospheric pressure and p_v = 2340 *Pa* a chosen vapour pressure, indicates that cavitation can occur in the specific operating point. The cavitation volume v_{cav} is defined as the volume where Eq. (6) is satisfied. For part load and at the best point no cavitation is allowed at all. The maximum cavitation volume is set to $v_{cav,max}$ = 0 *mm³*. For full load the cavitation is restricted to $v_{cav,max}$ = 50 *mm³*, which corresponds to approximately 0.8 ‰ of the rotor blade channel. In the first optimisation run the minimum static pressure is not limited.

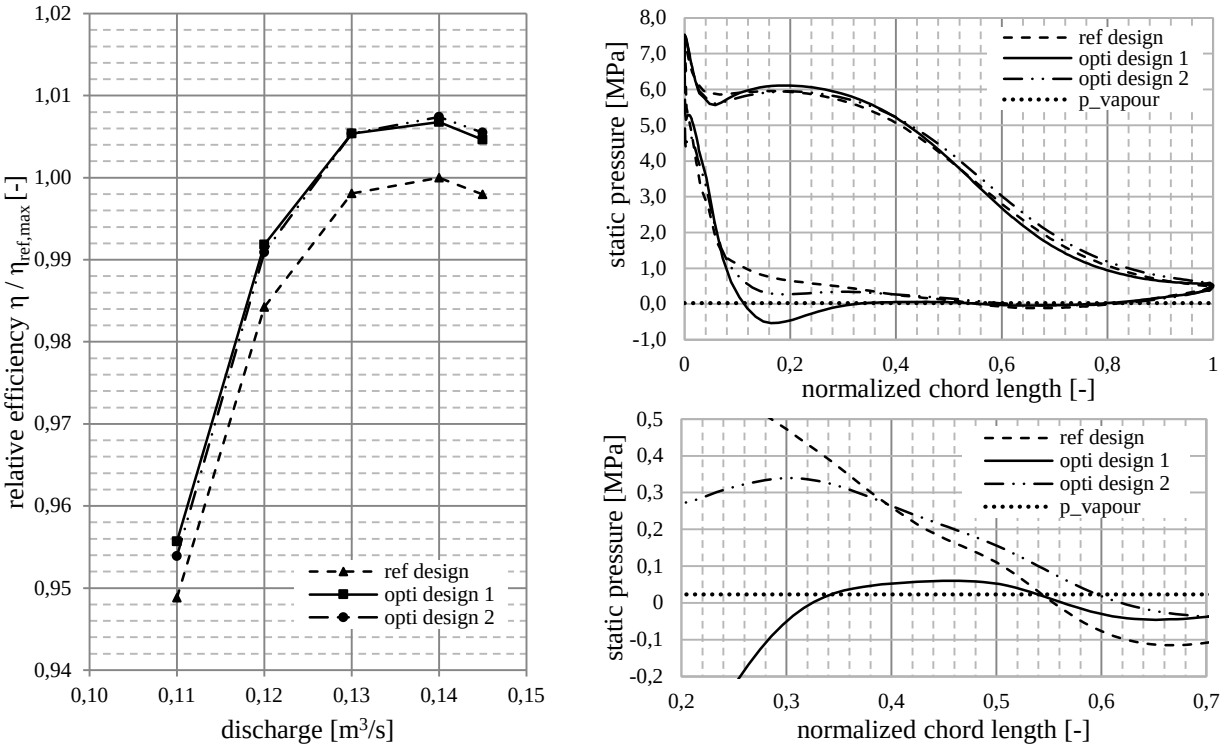

Fig.4 Characteristic curves (left graph). Static pressure over normalized chord length (right graphs).

The left graph in figure 4 shows that in terms of efficiency increase both optimisation runs were successful. The average efficiency could be increased by approximately 0.7 %. For both optimised designs the size of the cavitation volume v_{cav} stayed nearly the same compared to the reference design (ref design). The two graphs on the right show the static pressure along the three different blade designs. The first optimised design (opti design 1) cavitation occurs in two different blade areas. Compared to the reference design a new cavitation area at the front region of the blade appears. The intermediate pressure rise above the vapour pressure p_v between the two cavitation areas (between 0.34 and 0.54) significantly increases the risk of cavitation erosion on the optimised blade.
As a reaction on the bad cavitation behaviour of the first optimised design, the minimum static pressure is set to $p_{st,lim}$ = -11000 *Pa* and used as bounded output value in the second optimisation run (opti design 2). The limitation of the minimum static pressure prevents the pressure drop at the front area of the blade and leads to a pressure distribution along the blade

similar to the reference design (Fig. 4 right). It can be assumed that the cavitation behaviour of the second optimised design and the reference design are nearly identical. The blade design resulting from the second optimisation run increases the turbine efficiency without narrowing the operating range.

4. CONCLUSION

The first conclusion drawn by comparing the two different optimisations runs is that the setup of the boundary values has a significant influence on their respective progress. A higher number of bounded output values leads to a slower convergence and as consequence to higher affordable computing time. Due to usually limited computer capacities the number of boundary values should be kept reasonable small.
On the other hand the CFD results show that the introduction of additional bounded output values can have great impact on the optimised design. By adding the minimum static pressure as a second boundary value, the cavitation behaviour of the turbine could be improved significantly. The new optimised blade design leads to an expansion turbine with 0.7 % higher average efficiency while maintaining the same wide operating range as the reference turbine.

5. ACKNOWLEDGEMENTS

ANSYS Germany GmbH is gratefully acknowledged for support of this research.

6. REFERENCES

[1] Neipp, A. and Riedelbauch, S. (2014), Modular Axial Expansion Turbine for Energy Recovery from Working Fluids, *18th International Seminar on Hydropower Plants, Vienna.*

[2] Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley Publishing Company. 1989.

[3] Ruopp, A., Ruprecht, A. and Riedelbauch, S. (2011), Automatic Blade Optimisation of Tidal Current Turbines Using OpenFoam, *European Wave and Tidal Energy Congress (EWTEC), Southhampton.*

[4] Ruprecht, A., Ruopp, A. and Simader, J. (2012), CFD Based Mathematical Optimization of Hydro Turbine Components Using Cloud Computing, *Sym Hydro, Hydraulic modelling and uncertainty, Sophia Antipolis.*

[5] Tismer, A., Schlipf, M. and Riedelbauch, S. (2014), An Object-Oriented approach for a Highly Customizable Framework to Design Hydraulic Machines, *18th International Seminar on Hydropower Plants,Vienna.*

7. NOMENCLATURE

$c, c_1, c_2, c_{in}, c_{tip}, c_{out}$	(mm)	lengths	s_{dev}	(-)	standard deviation
f_t	(-)	target function	t, t_{in}, t_{tip}, t_{out}	(-)	length width ratios
M	(Nm)	torque	v_{cav}, $v_{cav,max}$	(mm^3)	cavitation volumes
n_{ind}	(-)	number of individuals	$w, w_{in}, w_{tip}, w_{out}$	(mm)	widths
n_{szr}	(-)	number of survivors	$x_{t0}, x_{in}, x_{tip}, x_{out}$	(-)	length ratios
p_{at}	(Pa)	atmospheric pressure	α_0, α_{in}, α_{out}	(°)	angles
$p_{st,min}$, $p_{st,lim}$	(Pa)	static pressures	Δp	(Pa)	pressure difference
p_v	(Pa)	vapour pressure	η ,η_{ave}	(%)	efficiencies
Q	(m^3 s^{-s})	flow rate	ω	(rad s^{-1})	angular velocity