Algebraic method for efficient adaption of structured grids to fluctuating geometries

D E Bohn* and **N Moritz**
Institute of Steam and Gas Turbines, RWTH Aachen University, Aachen, Germany

The manuscript was received on 8 June 2004 and was accepted after revision for publication on 13 January 2005.

DOI: 10.1243/095765005X7619

Abstract: An efficient method for adaption of a structured grid to fluctuating turbine blade geometry is presented based on an algebraic algorithm. The objective of the application of this method is to analyse the aerodynamic, thermal and rotational load of rotating and cooled blades with a conjugate approach. The grid adaption method is validated with two test cases by using a simple deformation model considering the blade as a torsion spring. This model ensures a strong coupling between aerodynamic load and deformation of the blades. Thus, the stability of the numerical code can be analysed. The calculations show that convergence for the blade deformation is reached very soon. Even for great blade deformation the algebraic grid adaption method generates no negative cell volumes although this cannot be guaranteed by an algebraic algorithm.

Keywords: turbine flow, blade untwist, grid adaption, stuctured grid

1 INTRODUCTION

Increasing the efficiency of modern gas and steam turbines is still an important subject of scientific research. In recent years, the application of modern numerical tools has allowed the reduction in the necessity for costly experimental investigations in the design process of new turbine blades. The optimizing process taking place requires the occurring physical phenomena to be captured by suitable models with sufficient accuracy and as completely as possible.

One objective of numerical analysis is the prediction of the aerodynamic, thermal and rotational load of twisted, cooled rotating blades. Here, the goal is to develop a high-quality numerical tool to analyse all the physical phenomena using a conjugate approach, that is, using one numerical code for the coupled calculation of a blade's load.

Owing to the aerodynamic, thermal and rotational forces occurring during operation the geometric shape of the blade deviates from the originally manufactured shape. A first approach to describe theoretically the untwisting of a twisted rotating blade was presented by Ohtsuka [**1**]. This method is based on the solution of Herrmann [**2**] to predict the torsional warping of linear and axial blade axes. A validation of the analytic results for a discretized blade was performed by comparison with experimental results. However, this technique neglected the interaction between material and fluid flow. Thus, the calculations are valid only for blades rotating in vacuum.

Liu [**3**] developed a technique that allows the consideration of the influence of fluid flow on the blade untwist by application of the variational principle. Pressure and strain distribution in the blade material and the ambient flow field can be predicted by a FEM analysis. This method, valid first for isotropic materials, was improved for consideration of anisotropic materials by Liu [**4**].

Besides the investigation of the static load of a turbine blade, the analysis of the dynamic interaction between fluid and blade is an object of scientific research too. Qualitative predictions on the dynamic load of airfoils and propeller blades have been possible for several years. The theory of aeroelastics and the finite element method were connected by Nellessen [**5**] so that the calculation of lift distribution and of dynamic overload in the case of airfoil

*Corresponding author: Institute of Steam and Gas Turbines, RWTH Aachen University, Tempergraben 55, D-52056 Aachen, Germany.

blast interaction is possible for airfoils with high aspect ratios. Based on the before mentioned publications, Liu [**4**] suggested a technique based on the variational principle to predict the blade flutter of rotating blades.

For both static and dynamic analysis of blade untwist and vibration it is necessary to modify the computational grid during the runtime of a CFD calculation. One possibility is to calculate a new grid for the new blade geometry by integration of a grid generation tool in the numerical fluid flow analysis code. However, this method would be very time-consuming. Thus, a more efficient method is given by adaption of the primary grid once defined by the grid generator to the fluctuating blade geometry.

2 CONJUGATE FLUID FLOW AND HEAT TRANSFER SOLVER (CHTflow)

2.1 Governing equations

The governing equations for the conservative variables in arbitrary, body-fitted coordinates ξ, η, ζ with the fluxes in normal directions to the cell faces, for which ξ, η, ζ are constant, read for the fluid flow

$$\mathbf{U}_t + \mathbf{E}_\xi + \mathbf{F}_\eta + \mathbf{G}_\zeta = 0 \tag{1}$$

with

$$\mathbf{U} = \mathrm{J} \cdot (\rho, \rho u, \rho v, \rho w, \rho e_t)^T \tag{2}$$

$$\mathbf{E} = \mathrm{J} \cdot \left(\tilde{\mathbf{E}} \cdot \xi_x + \tilde{\mathbf{F}} \cdot \xi_y + \tilde{\mathbf{G}} \cdot \xi_z\right) \tag{3}$$

$$\tilde{\mathbf{E}} = \begin{pmatrix} \rho u \\ \rho u^2 - \tau_{xx} \\ \rho uv - \tau_{xy} \\ \rho uw - \tau_{xz} \\ (\rho e - \tau_{xx})u - \tau_{xy}v - \tau_{xz}w + q_x \end{pmatrix} \tag{4}$$

The fluxes **F** and **G** are obtained analogously. **U** is the vector of the conservative variables, $\tau_{xx}, \ldots$ etc. are the components of the Reynolds stress tensor, **q** is the heat flux vector. J is the cell volume and the expressions $\xi_x, \ldots$ are the metrics, which result from the coordinate transformation. A more detailed description of the conjugate calculation method and its validation can be found in Bohn *et al.* [**6**, **7**].

2.2 Computational model

The numerical procedure for the present investigation performs a conjugate flow and heat transfer calculation. It works on the basis of an implicit finite volume method combined with a multiblock technique. The physical domain is divided into separate blocks and the full, compressible, three-dimensional Navier–Stokes equations are solved in the fluid blocks. Making use of a Newton method leads to the following approximation of the conservation equations in implicit form.

$$\frac{\Delta U}{\Delta t} + (\underline{\mathbf{A}}^n \cdot \Delta\mathbf{U})_\xi + (\underline{\mathbf{B}}^n \cdot \Delta\mathbf{U})_\eta + (\underline{\mathbf{C}}^n \cdot \Delta\mathbf{U})_\zeta = -\left(\mathbf{E}^n_\xi + \mathbf{F}^n_\eta + \mathbf{G}^n_\zeta\right) = \mathrm{RHS} \tag{5}$$

where ***A***, ***B***, ***C*** are the Jacobians of the fluxes **E**, **F** and **G**. The solution vector $\mathbf{U}^{n+1}$ at the new time level can be obtained by adding $\Delta\mathbf{U}$ to the old solution $\mathbf{U}^n$.

For the inviscid fluxes an upwind discretization is used. A locally one-dimensional Riemann problem is solved on each cell face [**8**]. Thus, the transport of information is modelled in a physically correct manner. This complex calculation preserves high accuracy even with transonic flows. With respect to numerical diffusion a Godunov type flux-differencing is employed [**9**]. In order to achieve a third-order accuracy, van Leer's MUSCL-technique is used [**10**]. Since the Godunov flux is not sufficiently diffusive to guarantee stability in regions with complex flow phenomena, it is combined with a modified Steger–Warming flux (flux vector splitting) [**11**].

The viscous fluxes are approximated using central differences. The solution of the resulting system of linear equations is solved by a Gauss–Seidel point iteration scheme. The closure of the conservation equations is provided by the algebraic eddy-viscosity turbulence model by Baldwin and Lomax [**12**].

3 GRID ADAPTION ALGORITHM

3.1 Laplacian grid adaption

The calculation of the deformation of turbine bladings requires the possibility to adjust the numerical domain to the changing geometric conditions within the run of a fluid flow solver. One option to solve this problem is to generate a new grid during the runtime, which, however, involves a high amount of computational time. For complex geometries, the calculation of a computational grid can last several hours.

A faster method is given by adaption of the grid only in the near surface part, that is, that the original grid representing the primary blade geometry is modified. Here it is necessary to ensure that the modified grid includes no cells with a negative volume. Hence, any intersection of grid lines has to be prevented.

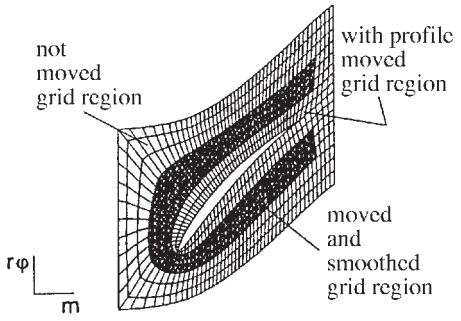

Fig. 1 Grid adaption method by Peitsch [**14**]

The unsteady interaction of oscillating blades and cascade flows was investigated by He [**13**]. The FEM Euler equation analysis was combined with a zonal grid treatment so that for each time step the grid could be adapted to the actual deformation of the blade. The grid adaption was performed with a first-order arithmetic progression to fit both the fixed region and the moving blade boundaries.

For calculation of fluttering compressor bladings, Peitsch [**14**] suggested a method to calculate a new geometry matching grid based on the solution of the Laplacian differential equations. The computational grid is divided into three regions (Fig. 1). In a region close to the blade surface the grid points are moved corresponding to the movement of the blade surface (with profile moved grid region). In the so-called 'moved and smoothed grid region' the Laplacian equations are solved to reduce the distortion of the grid cells. The outer region of the grid is not changed in this method.

The major advantage of the grid adaption method by Peitsch is that the solution of the Laplacian equations ensures by mathematical reasons the suppression of negative cell volumes. However, its applicability is given in a small grid region due to loss of orthogonality of the grid cells. Furthermore, the use of the Laplacian equations in the smoothing region yields a more homogenized grid point distribution. This result is not convenient for turbulent turbine flow calculation due to the necessity of a high grid line density in the range of the boundary layer. Moreover, solving the differential equations is quite time-consuming.

3.2 Algebraic grid adaption

In order to reduce the computational effort, a new grid adaption method for application to turbine flow calculations was developed on the basis of an algebraic algorithm. Corresponding to the Peitsch method the computational grid is divided into three regions as they are sketched in Fig. 2 for one of the tested grids.

The region closest to the blade surface is the moving grid region limited by the blade surface grid points and the grid points on the last moving grid line (white grid line with $\zeta = \zeta_{\max,\mathrm{move}}$). All grid points that are on the same $\xi = \text{const.}$ grid coordinate (e.g., reference grid line) and that are within these boundaries are moved by the same vector Δ as the surface grid point of this grid coordinate.

In the fixed grid region that is limited by the outer boundary of the grid and the first fixed grid line ($\zeta = \zeta_{\min,\mathrm{fixed}}$) all grid points always have the same position as in the primary grid.

In the adapted grid region, that is, located between the moving grid region and the fixed grid region, the algebraic grid adaption algorithm is used to determine the movement vector of each grid point. The first step in this algorithm is to calculate the length l_{local} of the actual grid line $\xi = \text{const.}$ between the grid lines $\zeta = \zeta_{\max,\mathrm{move}}$ and $\zeta = \zeta_{\min,\mathrm{fixed}}$. This length

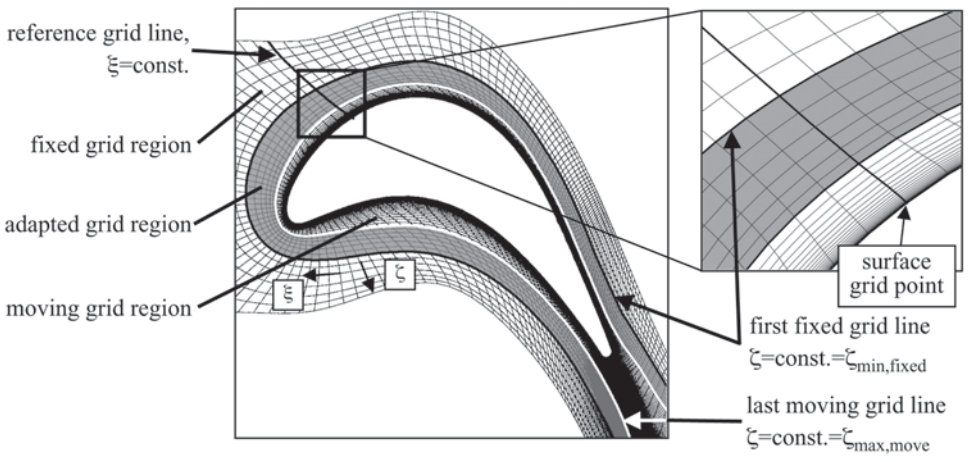

Fig. 2 Algebraic grid adaption method

is used to determine the non-dimensional distance $\tilde{l}$ of each grid point in the adapted grid region to grid line $\zeta = \zeta_{\max,\text{move}}$. With using a weighting function $w(\tilde{l})$ the local movement vector $\delta(\xi,\zeta)$ is calculated.

$$\delta(\xi, \zeta) = w(\tilde{l}) \cdot \Delta \tag{6}$$

The weighting function for the algebraic grid adaption that is a function of the non-dimensional length $\tilde{l}$ has to fulfil several boundary conditions. For the non-dimensional length $\tilde{l} = 0$ the value for the weighting function has to be $w = 1$ so that the position of the grid points near to the moving grid region is changed nearly by the same vector as the points in the moving grid region. The value of the weighting function for $\tilde{l} = 1$ has to be $w = 0$ so that the grid points near to the grid line $\zeta = \zeta_{\min,\text{fixed}}$ are nearly not moved. A recommended additional condition is that the derivative of the weighting function for $\tilde{l} = 0$ and $\tilde{l} = 1$ is $w' = 0$ to achieve a small change in grid line curvature at the boundaries of the adapted grid region.

These boundary conditions can be captured with trigonometric functions (e.g., cosine function) or with fourth-order polynomial functions. They were defined as the following.

$$w_{\cos}\left(\tilde{l}\right) = 0.5 \cdot \left[\cos\left(\pi \cdot \tilde{l}\right) + 1\right] \tag{7}$$

and

$$w_{\text{pol}}\left(\tilde{l}\right) = (-0.5 \cdot P + 3) \cdot \tilde{l}^4 + (P - 4) \cdot \tilde{l}^3 - 0.5 \cdot P \cdot \tilde{l}^2 + 1 \tag{8}$$

with parameter P to change the shape of the weighting function by modification of P from 0 to 10. In Fig. 3 the cosine function and two polynomial function ($P = 0$ and $P = 10$) are shown. If the parameter is set to values beyond these boundaries the weighting function will have a local extremum between $\tilde{l} = 0$ and $\tilde{l} = 1$.

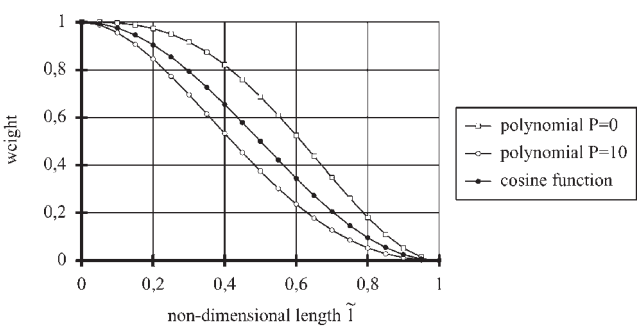

Fig. 3 Weighting functions for algebraic grid adaption

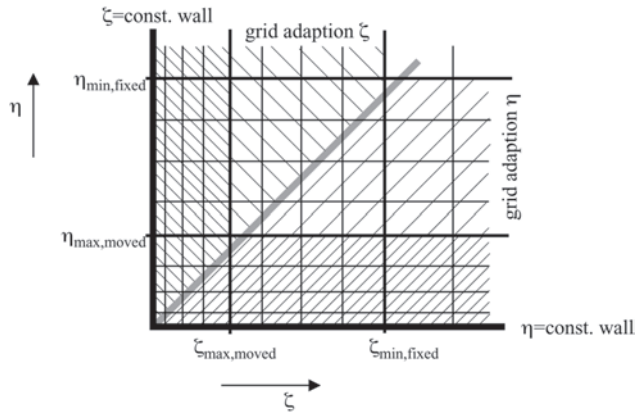

Fig. 4 Three-dimensional grid adaption near two walls

After the determination of the adapted numerical grid the corresponding Jacobian matrix for the coordinate transformation from Cartesian coordinates to body-fitted coordinate is calculated. If negative values occur the calculation is stopped since this indicates that the grid adaption has led to negative cell volumes. In this case the polynomial weighting function can be used to avoid the appearance of negative volumes. However, even for great geometric changes of the blade surface the grid adaption method has not generated negative cell volumes.

The described grid adaption algorithm is applicable to three-dimensional grids, too. Considering the blade height to coincide with the coordinate η, the grid adaption region can be defined as mentioned above. The blade deformation is determined for all surface points in the ξ–η-plane so that for each $\eta = \text{const.}$ plane the grid adaption algorithm can be performed.

Concerning the capturing of two fluctuating walls, the algorithm has to be improved by an additional adaption zone in, for example, the η direction. Near the two walls, for each grid point the adaption method determines whether it is closer to the η-wall or to the ζ-wall (Fig. 4). According to the wall distances, the grid adaption is performed with the deformation vector Δ from the corresponding point on the wall.

4 TEST CASES

The algebraic grid adaption method was tested with two turbine blades with subsonic flow conditions. The mesh for test case 1 is a C-type grid; test case 2 has an H-type grid in the main flow. The primary grids are displayed in Fig. 5.

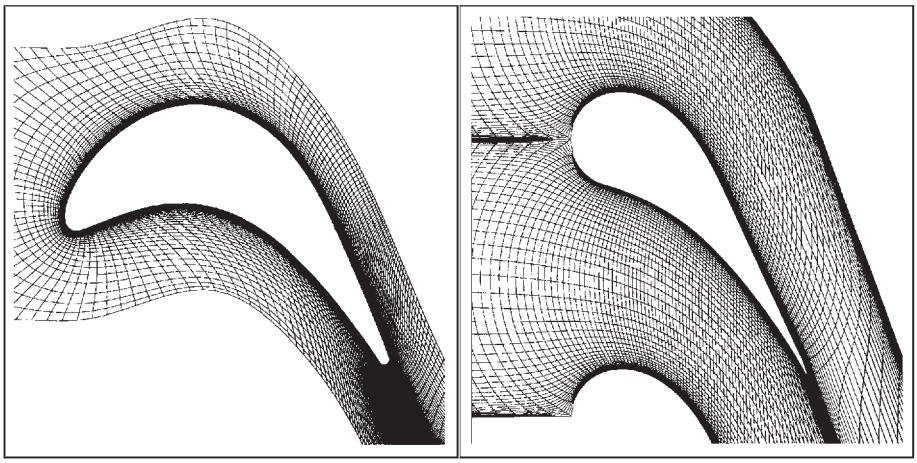

Fig. 5 Primary numerical grids

Figures 6 and 7 show the cell skewness and the cell aspect ratio. For cell skewness the cells are categorized by the minimum cell angle. It can be seen that the orthogonality of test case 1 is higher than for test case 2 due to the C-type grid. This type of grid allows a higher grid resolution in the near wall regions so that a higher percentage of the cells has an aspect ratio lower than 0.1 (see right-hand side of Figs 6 and 7).

A simple model was implemented to determine the deformation of the blades. The main target of the model was not to determine the blade deformation as precisely as possible, but to prove that the grid adaption method can generate useful grids and that a converged solution is reached. Thus, the blades were considered as torsion springs with a non-changing profile.

The torsional momentum M_t for the area centroid (Fig. 8) is determined by integration of the forces on the blade surface resulting from the pressure distribution.

$$M_t = \int_A \mathbf{r} \times (-p\mathbf{n})\, \mathrm{d}A \tag{9}$$

The deformation angle α is calculated from the moment of inertia I_p, the shear modulus G and the blade height H, which is derived from the axial chord length, to be

$$\alpha = \frac{M_t \cdot H}{G \cdot I_t} \tag{10}$$

By this means, there is a strong coupling between the flow field calculation and the changing blade surface outline. Thus, the stability of the numerical solver in the case of fluctuating geometries can be analysed by

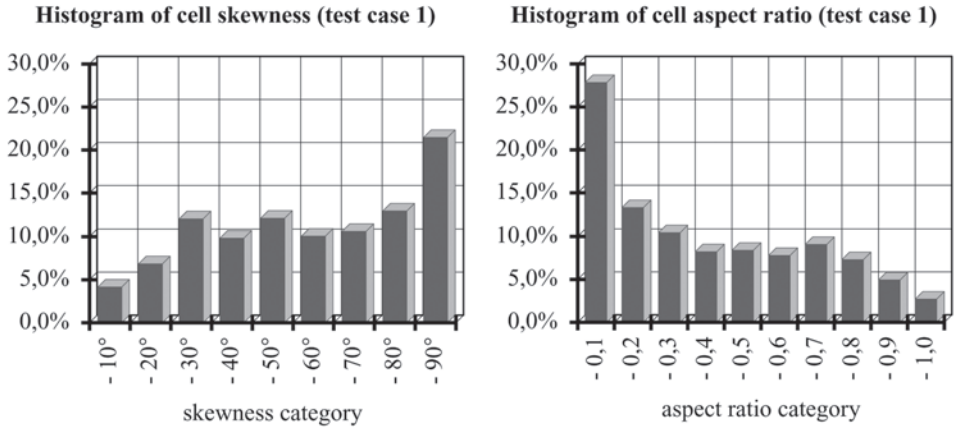

Fig. 6 Cell skewness and cell aspect ratio of test case 1 (C-type)

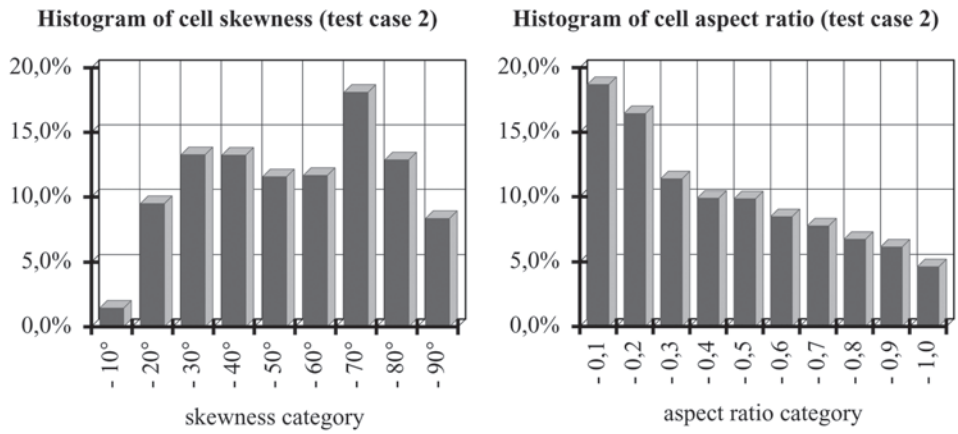

Fig. 7 Cell skewness and cell aspect ratio of test case 2 (H-type)

the development of the torsion of the blades depending on the time steps. Using common values for the shear modulus (i.e., $G = 72.3 \times 10^3\ \mathrm{N/mm^2}$ for case 1 and $G = 80.0 \times 10^3\ \mathrm{N/mm^2}$ for case 2) the deformation angle yields values of $\alpha \approx 0.4 \times 10^{-3}$. These values are too small to prove the stability of the flow solver in combination with grid adaption method. Therefore, the values of the shear modulus were reduced by three orders to achieve more significant modifications of the surface.

First, calculations without deformation of the blades were performed to create the initial solution for the grid adaption run. Test case 1 is a gas turbine blade designed for experimental and numerical investigations. Test case 2 is a down-scaled steam turbine profile that was used in an air test turbine. The inlet and outlet boundary conditions for the computation are given in Table 1.

5 RESULTS

5.1 Flow field

The results of the initial computation that was performed without deformation of the blades are shown in Fig. 9 for test case 1 and in Fig. 10 for test case 2. On the left-hand side, the Mach number distribution is displayed. The right-hand side shows the pressure distribution and the flow vectors. Owing to the subsonic character of the flow for both test cases there are no shocks on the suction side of the blades. The maximum Ma-number for test case 1 is $Ma = 0.4$, for test case 2 it is $Ma = 0.4126$. The calculations were performed with the design boundary conditions, so that no separations occur in the flow field.

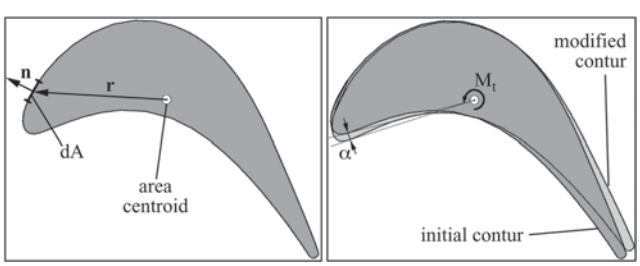

Fig. 8 Vector definition for calculation of torsional momentum and blade deformation

5.2 Influence of deformation model

Starting with the initial solution, the deformation model and grid adaption algorithm were activated to predict a torsion angle both for test case 1 and test case 2. Due to considering only aerodynamic forces on the blade surface, but neglecting for example centrifugal forces, the deformations of the blades did not influence the flow field significantly. Therefore, a comparison of the Mach number distribution and pressure distribution in the flow channel between blades is not performed here.

Regarding the pressure distribution on the blades' surface, it can be found that even the predicted small torsion angles have some influence on the flow field. Figure 11 shows the surface pressure distribution for test case 1 and test case 2 depending on the axial

Table 1 Boundary conditions

	Test case 1	Test case 2
p_{t1}	99 453 Pa	1538 926 Pa
T_{t1}	293.773 K	371.934 K
α_1	50°	0°
p_2	93 088 Pa	142 4783 Pa

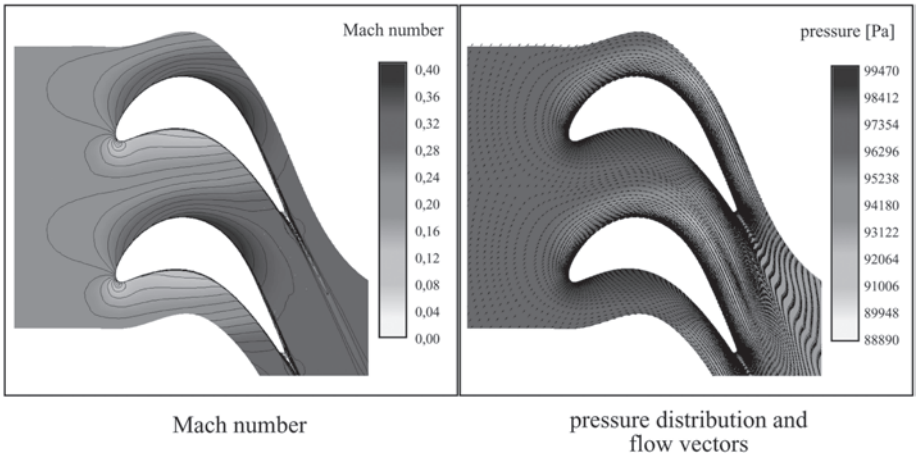

Fig. 9 Flow field results without grid adaption (test case 1)

chord length. For test case 1 the main influence can be found on the suction side of the blade. In the region from the axial chord length $x/L = 0.05$ to $x/L = 0.7$ the surface pressure is lower with consideration of the deformation model. The pressure distribution for test case 2 is significantly influenced on the pressure side in the region of the axial chord length $x/L = 0.2$ to $x/L = 0.3$. Here, a small flow separation can be found if the deformation model is taken into account.

In Fig. 12, the development of the torsion angle α depending on the numerical time steps is shown. Positive values stand for a counter-clockwise rotation of the blade. The charts illustrate that nearly a constant deformation of the blades is achieved rapidly. For test case 1 the value of the torsion angle α fluctuates after 20 000 time steps in the range of ± 1 per cent of the mean value. Owing to a broad low velocity area on the suction side near the trailing edge the torsion angle fluctuation is higher for test case 1 than for test case 2. Here, only small deviations from the mean value can be found after 30 000 time steps. Owing to the lack of experimental data, a comparison to measurement results cannot be presented here. However, a good agreement to measured data should not be expected for the simple deformation model used for this investigation.

Nevertheless, the analysis of the progression of the blade's torsion angle shows that the numerical method with consideration of a deformation model is very stable in the case of application of the algebraic grid adaption model. Usage of the cosine function or usage of the polynomial function did not result in negative cell volumes in the numerical domain.

5.3 Verification of grid adaption algorithm

In this section the quality of the grid adaption method is analysed. Owing to the small torsion

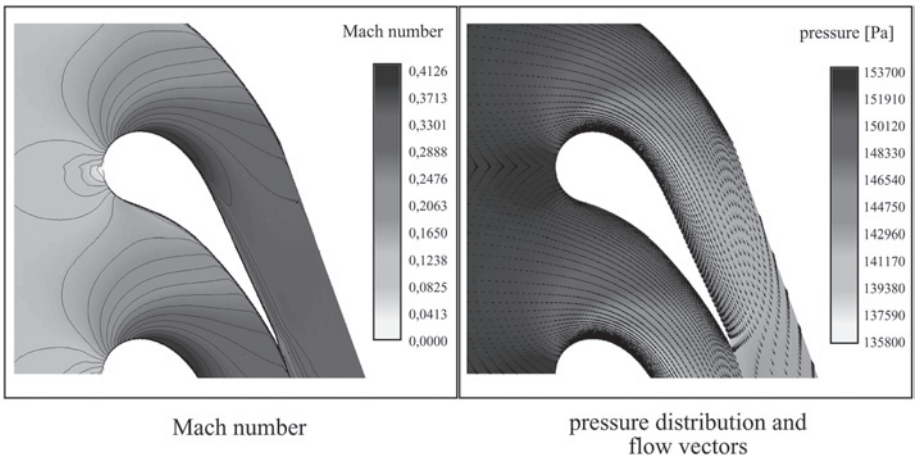

Fig. 10 Flow field results without grid adaption (test case 2)

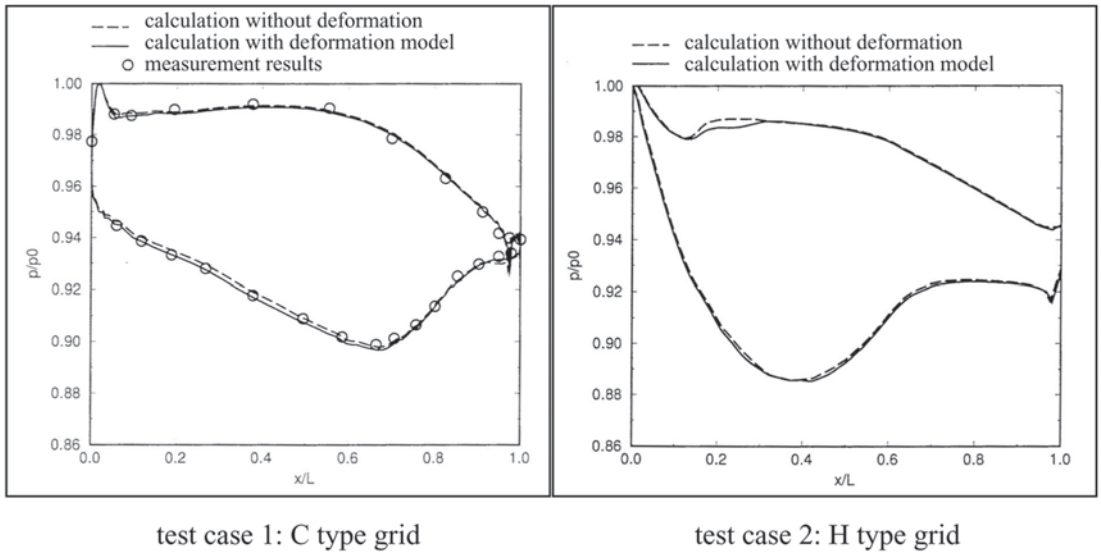

Fig. 11 Pressure distribution on blade surface

angles that were predicted by the deformation model, the verification of the grid adaption method was performed for a fixed torsion angle of $\alpha = 3^\circ$. Even for this deformation of the blade surfaces no negative cell volumes were detected. The following figures show on the left-hand side the primary and thus not deformed grid and on the right-hand side the adapted grid for the prescribed torsion angle. As the weighting function that is used in the adapted grid region the polynomial function with the parameter $P = 4$ was chosen.

5.3.1 *Test case 1*

In Figs 13 to 15 the grids for test case 1 are depicted. Figure 14 gives a detailed view of the leading edge region and Fig. 15 of the trailing edge region of the grid. It can be seen that the orthogonality of the cells near the blade surface is preserved very well. In the leading edge area the quality of the grid is barely affected by the prescribed deformation. Near the trailing edge, the sheared cells already in the primary grid are more skewed due to the grid adaption. However, this is not a considerable decline of grid quality.

5.3.2 *Test case 2*

The adapted grid for test case 2 is shown in Figs 16 to 18. Owing to high grid refinement in the region of the boundary layer a significant influence of the grid adaption method is to be found on the suction side near the trailing edge. Even in this region the orthogonality of the grid cells near the blade surface is preserved.

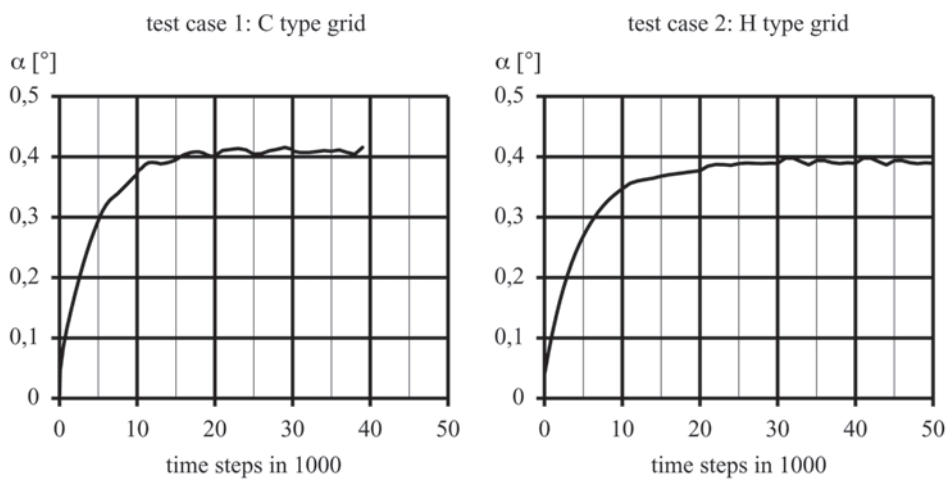

Fig. 12 Progression of deformation

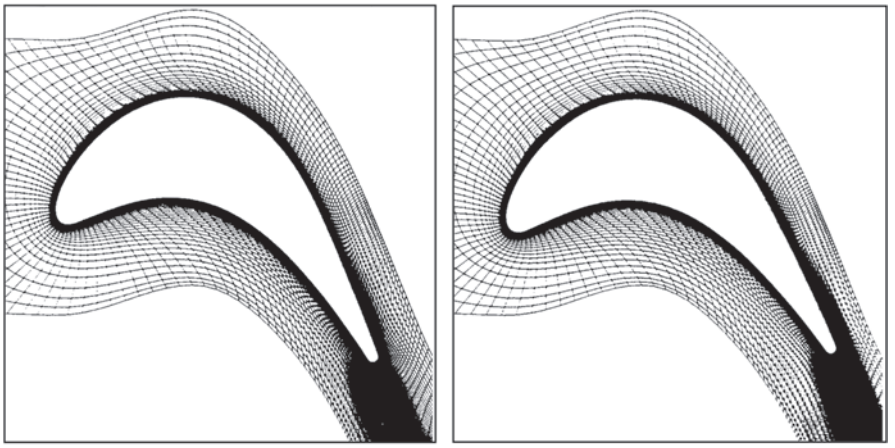

Fig. 13 Test case 1: quality of adapted grid

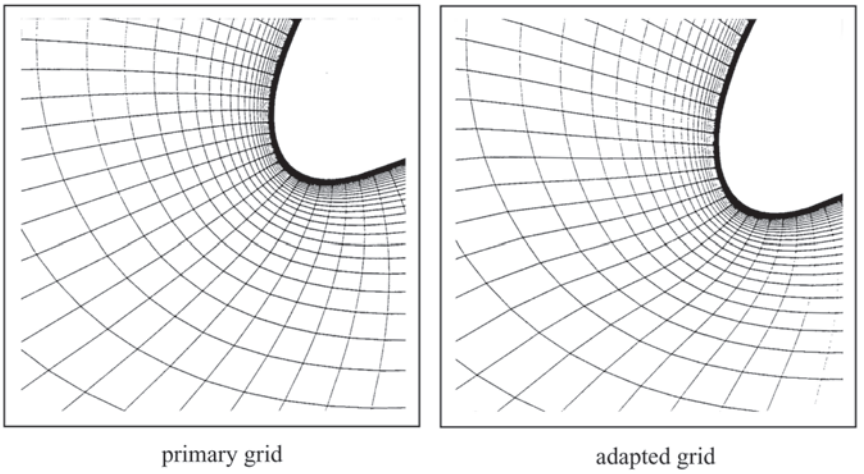

Fig. 14 Test case 1: quality of adapted grid (detail leading edge)

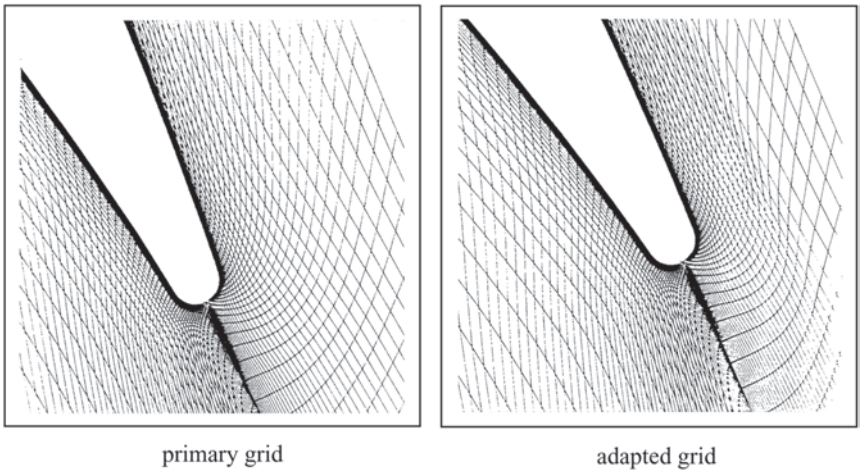

Fig. 15 Test case 1: quality of adapted grid (detail trailing egde)

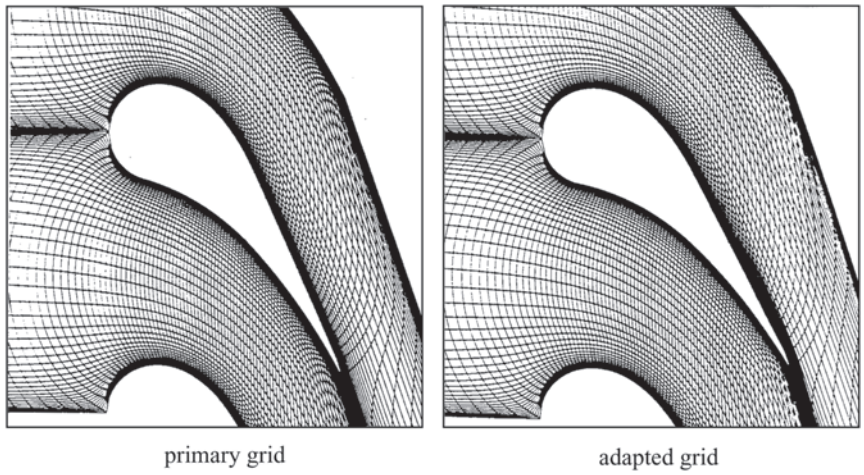

Fig. 16 Test case 2: quality of adapted grid

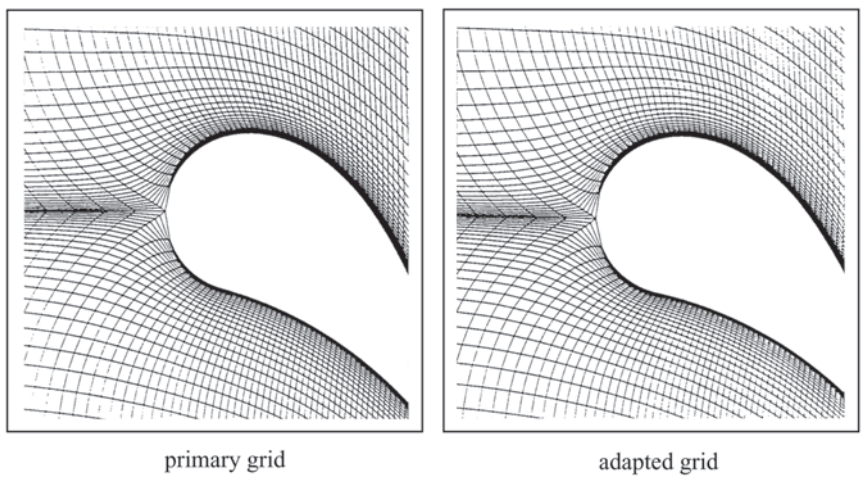

Fig. 17 Test case 2: quality of adapted grid (detail leading edge)

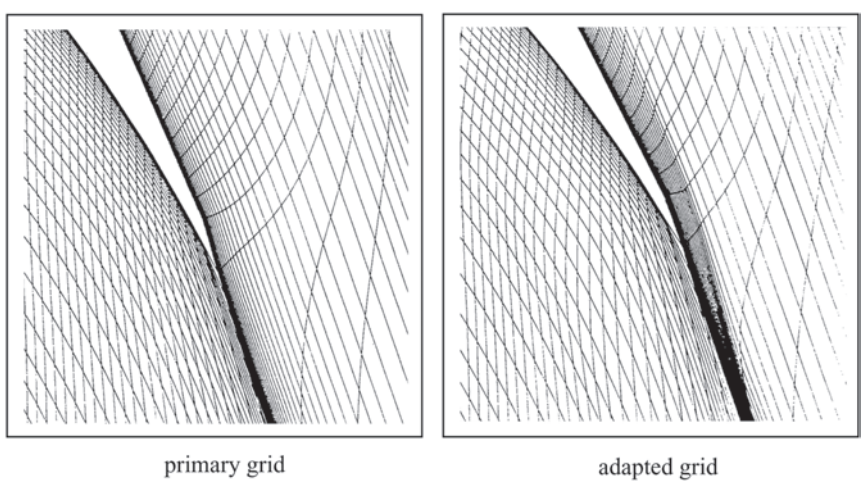

Fig. 18 Test case 2: quality of adapted grid (detail trailing edge)

6 CONCLUSIONS

In this publication a grid adaption method based on an algebraic algorithm has been presented. It has been tested with two test cases, which show that both *C*-type and *H*-type grids can be adapted to fluctuating geometries with this method.

Application of a simple blade deformation method shows that the stability of the fluid flow solver CHTflow is not influenced by the grid adaption method. In fact, the predicted blade deformation has a good convergency behaviour.

Algebraic grid adaption methods cannot prevent the occurrence of negative cell volumes. However, in all tests of the suggested method, even for changes of the blade geometry with a torsion angle $\alpha = 3^\circ$, no negative volumes have been detected.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG). The responsibility for the content of this publication lies with the authors.

REFERENCES

1 **Ohtsuka, M.** Untwist of rotating blades. ASME paper 74-GT-2, 1974.
2 **Herrmann, L. R.** Elastic torsional analysis of irregular shapes. *Proc. Am. Soc. Civ. Eng. J. Eng. Mech. Div.* 1965, **91**(EM 6), 11–19.
3 **Liu, G. L.** Untwist of Rotating Blades as an Aeroelasticity problem: A Unified Variational Theory. Memorial Tribute Volume to Prof. Chung-Hua Wu, 1993 (Machinery Press, Beijing, China).
4 **Liu, G. L.** The generalized untwist problem of rotating blades: a coupled aeroelastic formulation. *Int. J. Turbo and Jet Engines*, 1995, **12**, 109–117.
5 **Nellessen, D. M.** Schallnahe Strömungen um elastische Tragflügel. PhD thesis, RWTH Aachen University, Aaden, Germany.
6 **Bohn, D., Bonhoff, B., Schönenborn, H.,** and **Wilhelmi, H.** Validation of a numerical model for the coupled simulation of fluid flow and diabatic walls with application to film-cooled turbine blades. VDI-Berichte 1186, 1995.
7 **Bohn, D., Schönenborn, H., Bonhoff, B.,** and **Wilhelmi, H.** Prediction of the Film-Cooling Effectiveness in Gas Turbine Blades Using a Numerical Model for the Coupled Simulation of Fluid Flow and Diabatic Walls, 1995, pp. 1150–1159 (ISABE 95-7105).
8 **Schmatz, M. A.** Three-Dimensional Viscous Flow Simulations Using an Implicit Relaxation Scheme. Notes on Numerical Fluid-Mechanics (NNFM), Vol. 22, pp. 226–242 (Vieweg, Braunschweig).
9 **Godunov, S. K.** A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mathematicheskii Sbornik*, 1959, **47**, 271–290.
10 **Anderson, W. K., Thomas, J. L.,** and **van Leer, B.** A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations. *AIAA 85-0122*, 1985.
11 **Eberle, A., Schmatz, M. A.,** and **Bissinger, N.** Generalized flux vectors for hypersonic shock-capturing. *AIAA 90-0390*, 1990.
12 **Baldwin, B. S.** and **Lomax, H.** Thin layer approximation and algebraic model for separated turbulent flows. *AIAA 78-257*, 1978.
13 **He, L.** An Euler solution for unsteady flows around oscillating blades. ASME paper 89-GT-279, 1989.
14 **Peitsch, D.** Berechnung der dreidimensionalen, instationären Druchströmung schwingender Turbomaschinengitter mit einem expliziten Zeitschrittverfahren. PhD thesis, RWTH Aachen University; Aachen, Germany.

APPENDIX

Notation

A	area, m^2
$\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$	Jacobian matrices
e	specific internal energy, m^2/s
$\mathbf{E}$, $\mathbf{F}$, $\mathbf{G}$	fluxes
G	shear modulus, N/m^2
H	blade height, m
I	moment of inertia, m^4
J	Jacobian of transformation
l	length, m
L	axial chord length, m
M_t	torsional momentum, Nm
Ma	Mach number
$\mathbf{n}$	vector perpendicular to surface, m
p	pressure, Pa
P	weighting function parameter
$\mathbf{q}$	heat flux vector kg/s^2
$\mathbf{r}$	vector, m
t	time, s
T	temperature, K
$\mathbf{U}$	vector of conservative variables
u, v, w	cartesian velocity components, m/s
w	weighting function
x, y, z	cartesian coordinates, m
α	torsion angle, degree
Δ	movement vector on surface, m
δ	local movement vector, m
ξ, η, ζ	arbitrary coordinates, m
ρ	density, kg/m^3
τ	components of Reynolds stresses, kg/m^2

Subscripts

fixed	fixed grid region
local	local grid position in adapted grid region
max	maximum
min	minimum
move	moving grid region
p	polar
t	total
0	reference value
1	at inlet
2	at exit

Superscripts

T	transposed
n	time level
$\sim$	non-dimensional