Parallel Computation of Stator-Rotor Interaction in an Axial Turbine

Ruprecht, A., Bauer, C., Gentner, C., Lein, G.

Institute for Fluid Mechanics and Hydraulic Machinery
University of Stuttgart
Pfaffenwaldring 10, 70550 Stuttgart, Germany
Tel.:+49-711/685-3256 Fax:+49-711/685-3255
e-mail: ruprecht@ihs.uni-stuttgart.de

ABSTRACT
The unsteady flow caused by the interaction of guide vanes and runner is investigated for an axial turbine. The calculation is performed in parallel. The solution algorithm and the parallelization procedures are described. The parallelization is obtained by domain decomposition with overlapping meshes. The interaction of rotor and stator is handled by a sliding interface where downstream the node values are interpolated and upstream the flux integrals are exchanged. The simulation results are compared with measurements.

NOMENCLATURE
G Production term in k- and $\varepsilon$-equations
P mean pressure
$U_i$ mean velocity components
$c_1, c_2, c_3$ coefficients in the k- and $\varepsilon$-equations
$k$ turbulent kinetic energy
t time
$x_i$ cartesian coordinates
$\Omega_i$ rotation vector
$\varepsilon$ dissipation rate
$\varepsilon_{ijk}$ cross-product tensor
$\nu$ kinematic viscosity
$\nu_t$ turbulent viscosity
$\rho$ density
$\sigma_k, \sigma_\varepsilon$ Prandtl/Smith number for k- and $\varepsilon$-equations

1 INTRODUCTION
The design of hydraulic turbines has been strongly supported by numerical flow simulations for years, e.g. [1,2]. It is fairly standard to simulate the different components separately. Since there are strong interactions between the components, especially between stator and rotor, many attempts have been made lately to introduce this interaction into the computation by different coupling procedures. In order to save computer power these simulations are performed steady state by applying an averaging procedure in the circumferential direction. Consequently only one channel of the runner has to be considered, e.g. [3,4].

However the flow is unsteady because of the stator - rotor interaction. In order to avoid resonance phenomena the number of channels of stator and rotor is usually unequal, consequently it is not sufficient to calculate the flow only in one channel and to apply periodic boundary conditions. Instead so many channels have to be considered until periodicity is given simultaneously at the rotor and at the stator. If there is no common division the entire stator and rotor have to be modeled. This requires great resources in computer power. To obtain a reasonable response time these simulations have to be performed in parallel.

In this paper the interaction between stator and rotor is analyzed for an axial type hydraulic turbine (AXENT), see figure 1. This turbine was developed for pressure recovery in piping systems such as water supply systems or chemical plants. The advantage of this turbine is its flow characteristic. The discharge is nearly independent of the turbine speed. This means that a water hammer does not occur in case of disconnection of the generator from the electrical network and therefore no security devices are necessary. For details of the turbine the reader is referred to [5,6].
The stator and rotor part of the turbine is shown in figure 2. The stator consists of 12 guide vanes and the rotor of 15 blades. Therefore periodicity can be assumed for 4 stator and 5 rotor channels.

![Figure 1: AXENT, axial pressure recovery turbine](image1)

**Figure 1:** AXENT, axial pressure recovery turbine

Since the geometry of the different channels is equal the computational grids for the different domains can be obtained without great additional generation effort. A parallel computation of each channel is therefore easily applicable. The load distribution is schematically shown in figure 3.

![Figure 2: Geometry of stator and rotor](image2)

**Figure 2:** Geometry of stator and rotor

In addition to that the rotor parts are calculated in a rotating frame of reference, whereas the stator channels are treated in a fixed frame. This requires a transformation of the velocity in the sliding interface.

![Figure 3: Decomposition of the calculation domain](image3)

**Figure 3:** Decomposition of the calculation domain

2 BASIC EQUATIONS

The basic equations are the time dependent Reynolds-averaged Navier-Stokes equations for an incompressible fluid. The Reynolds stresses are expressed by the Bousinesq turbulent viscosity hypothesis.

As already mentioned for the stator geometry the equations are solved in a fixed frame of reference. The flow in the rotor is calculated in a rotating frame of reference. The momentum equations in the absolute frame of reference are given in tensor notation:

\[
\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} + \frac{1}{\rho} \frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_j} \left( \nu \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) = 0
\]

(1)

In the rotating frame of reference they can be expressed as
\[
\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} + \frac{1}{\rho} \frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_i} \left[ \left( \mathbf{v} + \mathbf{v}_i \right) \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] = 2\varepsilon_{ij} \Omega_j U_k + \Omega_i \Omega_j x_j - \Omega_j \Omega_k x_j
\]  
\tag{2}

(\Omega_i is rotation vector). The momentum equations have to be solved together with the continuity equation which is equal in both cases
\[
\frac{\partial U_i}{\partial x_i} = 0.
\]  
\tag{3}

The turbulent viscosity is calculated either by a zero-equation model of Prandtl mixing length type or by the so-called standard k-\(\varepsilon\) model. This model is given in the following. The turbulent viscosity can be obtained by
\[
\nu_t = C_{\mu} \frac{k^2}{\varepsilon}.
\]  
\tag{4}

The turbulent kinetic energy \(k\) and its dissipation rate are calculated from the transport equations
\[
\frac{\partial k}{\partial t} + U_j \frac{\partial k}{\partial x_j} - \frac{\partial}{\partial x_j} \left[ \left( \mathbf{v} + \mathbf{v}_i \right) \frac{\partial k}{\partial x_j} \right] = G - \varepsilon
\]  
\tag{5}

and
\[
\frac{\partial \varepsilon}{\partial t} + U_j \frac{\partial \varepsilon}{\partial x_j} - \frac{\partial}{\partial x_j} \left[ \left( \mathbf{v} + \mathbf{v}_i \right) \frac{\partial \varepsilon}{\partial x_j} \right] = c_1 \frac{\varepsilon}{k} G - c_2 \frac{\varepsilon^2}{k}
\]  
\tag{6}

with
\[
G = \nu_t \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \frac{\partial U_i}{\partial x_j}.
\]  
\tag{7}

The model constants are summarized in table 1.

<table>
<thead>
<tr>
<th>(C_{\mu})</th>
<th>(\sigma_k)</th>
<th>(\sigma_\varepsilon)</th>
<th>(C_1)</th>
<th>(C_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.09</td>
<td>1.0</td>
<td>1.3</td>
<td>1.44</td>
<td>1.92</td>
</tr>
</tbody>
</table>

Table 1: Model constants of k-\(\varepsilon\) model

3 NUMERICAL METHODS

The calculations are carried out using the program FENFLOSS which has been developed at the institute for more than a decade [7].

3.1 Discretisation

The partial differential equations are solved by a Galerkin Finite Element Method. The spatial discretization of the domain is performed by 4-node quadrilateral elements for two-dimensional problems and by 8-node brick elements in three-dimensional cases. For the velocity components and the turbulence quantities a bi-linear respectively tri-linear approximation is applied. The pressure is assumed to be constant within the element. For advection dominated flow a Petrov-Galerkin formulation with skewed upwind orientated weighting functions is applied.

The time discretization is done by a three-level fully implicit finite difference approximation of 2\(^{nd}\) order.
\[
\frac{\partial \phi}{\partial t} \approx \frac{1.5\phi^{n+1} - 2\phi^n + 0.5\phi^{n-1}}{\Delta t} + O(\Delta t^2)
\]  
\tag{8}

3.2 Solution Procedure

For the solution of the momentum and continuity equation a segregated solution algorithm is used. Each momentum equation is handled independently. The momentum equations are linearized by successive substitution. The linear systems are solved by the BICGSTAB2 algorithm [8] with an incomplete LU decomposition (ILU) for preconditioning. The pressure is treated by a modified Uzawa type pressure correction scheme [9]. The pressure correction is carried out in a local iteration loop without reassembling the system matrices until the continuity error is reduced by a given order (usually 6-10 iterations needed).

After the solution of the momentum and continuity equations the turbulence quantities are calculated and a new turbulence viscosity is obtained. The k- and \(\varepsilon\)-equations are also linearized by successive substitution and the linear systems are solved by the BICGSTAB2 algorithm with ILU preconditioning.

The whole procedure is carried out in a global iteration until convergence is obtained. For unsteady simulations the global iteration has to be carried out in each time step. The iteration procedure is summarized in figure 5.
3.3 Parallel Computation

As already described the flow in the stator domain and in the rotor domain are both calculated in parallel. The parallelization is obtained by a domain decomposition. The divided grids have an overlapping range on both sides. This is schematically shown in figure 6. There the additional elements and nodes are shown in red for one domain and in blue for the other domain. The node values of the black nodes are the unknowns which are included in the system matrices. Due to the additional elements the assembled equations for the unknowns are correct. The influence of the additional nodes are introduced by data exchange between the domains in the matrix-vector multiplication in the BICGSTAB2 solver.

The preconditioning is carried out locally on each domain. In the program the data exchange is organized using MPI (Message Passing Interface).

3.4 Sliding Interface

At the interface from the stator to the rotor non-matching grids have to be allowed. Otherwise the grid generation would be very complicated and the time steps would be severely restricted. Overlapping meshes are used, schematically shown in figure 7.

The rotor and stator domains are calculated independently of each other. The information exchange from one domain to the other is organized in form of dynamical boundary conditions, which are updated in the global iteration, figure 5. The exchange is made by interpolation of the node values from the stator to the rotor, see figure 8. First the element is evaluated where the node in question is located. In the element the interpolation is carried out using the bi-linear respectively tri-linear shape functions. The interpolated node values are specified as Dirichlet boundary conditions and the values are updated in each global iteration.

From the downstream rotor to the upstream stator the fluxes are prescribed. These values are introduced to the surface integrals resulting from the weak Galerkin formulation (applying Green-Gauss theorem to the diffusion terms and pressure gradient of the momentum and the turbulence equations). The surface integrals are shown in figure 9.

**Figure 6: Parallel computing**

**Figure 5: Flow chart of the solution procedure**
Due to the rotation of the runner the grids move relatively to each other. Consequently the location of the interpolation nodes changes from time step to time step. The location can move from one processor to another. Therefore the overlapping elements and nodes are exchanged globally to each processor. With that the necessary information is available on each processor and the interpolation can be performed locally.

As already mentioned the flow in the stator domain is calculated in a fixed frame of reference whereas the flow in the runner is expressed in a rotating frame of reference. Consequently the node values of the velocity, which will be exchanged from one domain to the other, have to be transformed from one coordinate system to the other, see figure 10. In the stator the absolute velocity is determined and in the rotor the relative velocity (vector sum of absolute velocity and circumferential velocity of the runner).

4 APPLICATION

4.1 Measurements

The unsteady velocity distribution in the stator and in the rotor of a prototype turbine are measured with Laser-Doppler-Velocimetry. Pressure distributions were measured by pressure taps at the wall. Details of the measurements can be found in [10,11].
4.2 Boundary conditions

The simulation is carried out for the guide vanes, the runner and part of the draft tube, this means that two sliding interfaces have to be applied. For the simulation the boundary conditions shown in figure 11 are applied. At the inlet a uniform velocity distribution is prescribed. As already mentioned the simulation can be restricted to 4 stator channels and 5 rotor channels when assuming periodicity. In this case at the left and right end periodic boundary conditions are applied. However, if a non-symmetric inflow or outflow is applied, the complete rotor and stator have to be considered. At the blade surfaces as well as at the hub and shroud wall boundary conditions (logarithmic wall functions) are specified. At the outlet free outflow is assumed.

![Figure 11: Specified boundary conditions](image)

The distribution to the processors is shown schematically in figure 12. Each part has nearly the same number of nodes. Since the grid in the draft tube is coarser, the draft tube blocks are larger.

For the entire geometry 32 domains were used (12 guide vanes, 15 runner and 5 draft tube domains). The calculation is carried out partly on a Cray T3E-512 and partly on a Hitachi SR 2201.

4.3 Grids

The grids for the different rotor and stator channels are equal. Therefore only the grid of one channel is presented. In figure 13 the mesh of a guide vane channel is shown. It consists of approximately 40000 elements. Figure 14 shows the mesh of a runner channel. It also consists of approximately 40000 elements. In total together with the draft tube the grid consists of approximately 1.3 Mio nodes.

4.4 Results

For the simulation a time step was chosen which corresponds to a rotation of the runner of 1°.

In figure 15 the pressure distribution in the middle plane (a cylindrical plane on half the height of the blades projected to a flat surface) is shown for different time steps. The rotation of the rotor between the different plots is 5°.

It can be seen, that the pressure distribution is smooth across the sliding interface. The flow is accelerated in the guide vanes resulting in a pressure drop. It can be observed, that the pressure drop on the suction side near the trailing edge of the guide vanes is strongly affected by the position of the runner.

In the runner a further pressure reduction can be observed. On the suction side in the last part of the blades the lowest pressure occurs. This is the region where during the measurements inception of cavitation could be
observed, when the pressure level on the test rig was reduced.

**Figure 13**: Mesh of a guide vane channel

**Figure 14**: Mesh of a runner channel

The low pressure region on the suction side as well as the high pressure region on the pressure side of the blades clearly show the influence of the guide vane wakes. Shortly after the outlet of the runner, however, the pressure is quite uniform, although the calculation area is much longer than shown in the figure 15.

In figure 16 the relative velocity in a part of the runner is shown for a certain time step. Clearly the stagnation points can be seen at the runner inlet. The positions of the stagnation points are different for the blades shown, because of their different position relatively to the guide vanes. On the suction side of the blades the velocity maximum can be observed in the same position, where in figure 15 the minimum pressure could be detected.

Behind the runner the wakes of the blades can be seen clearly. The non-uniform structure of the wakes results from the wakes of the stator blades.

**Figure 15**: Pressure distribution in the middle plane for different time steps
A comparison of the simulation results with the measurements is shown in figure 17 and 18. The first figure shows the measured velocity in the runner transformed to the rotating coordinate system. In figure 16 the simulation results are shown. The velocity vectors are interpolated to the measuring points.

It can be seen that the simulation agrees quite well with the measurements. However, the flow angle at the runner outlet differs in some extent. In the simulation the flow follows the blade shape quite well, whereas in the measurements the blade angle and the flow angle are different.

The reason is presumed to be caused by the logarithmic wall functions, which tends to follow too much the geometrical shape.

In figure 19 the measured and calculated pressure distributions in different positions are plotted versus the rotation angle of the runner. Position 1 is located in front of the guide vanes. It can clearly be seen, that the fluctuation caused by the runner blades can be observed in front of the stator. Position 2 is located near the trailing edge of the guide vanes. Here, as expected, the pressure fluctuations are much stronger. In both positions the agreement between measurements and simulation is quite well. Position 3 is located at the wall at the end of the runner.
5 CONCLUSIONS

An algorithm has been presented for the parallel simulation of the unsteady flow in a hydraulic turbine. The parallelization is introduced by domain decomposition with overlapping grids. The stator-rotor interaction is handled in a sliding interface where node values are interpolated and exchanged in downstream direction and fluxes are transmitted in upstream direction.

The algorithm has been applied to an axial turbine. The simulation results agree quite well with the measurement data. The prediction of the flow angle at the end of the blades, however, shows some discrepancies, presumably due to the modeling of the near wall flow by logarithmic wall functions. An improvement will certainly be obtained by using finer grids and resolving the flow along the wall.

6 REFERENCES


