Case Study

Flow Visualization for Turbomachinery Design

Martin Roth and Ronald Peikert

Swiss Center for Scientific Computing, ETH Zürich, Switzerland
roth@scsc.ethz.ch, peikert@scsc.ethz.ch

Abstract

Visualization of CFD data for turbomachinery design poses some special requirements which are often not addressed by standard flow visualization systems. We discuss the issues involved with this particular application and its requirements with respect to flow visualization.

Aiming at a feature-based visualization for this task, we will examine various existing techniques to locate vortices. The specific flow conditions for turbomachines demonstrate limitations of current methods. Visualization of turbomachinery flow thus raises some challenges and research topics, particularly regarding feature extraction.

1 Introduction

Water turbines used in hydroelectric power plants are large machines (typical runner diameter 3-7 m, Fig. 1) manufactured and designed individually for the specific conditions of the installation and requirements of the customer. Today, efficiencies exceed the level of 93%. A thorough introduction to hydraulic machines can be found in [1], a brief overview in [2].

During the design process, Computational Fluid Dynamics (CFD) is routinely used for optimization and comparison [3]. To investigate details in the flow and analyze its response to small changes of the machine geometry, visualization of the 3D flow in its proper spatial relationship with the channel geometry is crucial. A representative flow is shown in Fig. 2.

2 Issues of Turbomachinery Flows

Industrial turbomachinery flow simulations typically provide a *steady* (time-averaged) solution of either Euler (inviscid) or Navier-Stokes (viscous) equations. For the rotationally symmetric parts, only one *channel* (the space between two *blades*) is simulated; rotating parts are calculated in a rotating frame. Grids used for turbine flow simulations are 3D, 3-space, irregular structured meshes fitted to the geometry of the passage. A typical mesh for a Francis runner is depicted in Fig. 3.

Although only the flow in one channel is modeled, there is a strong demand for visualizing the flow in the whole machine. Before entering the space between two blades, a large part of the flow crosses channel boundaries (as visible in Fig. 2). Displaying streamlines only inside one channel is not sufficient since it does not provide an informative picture of the flow around the blades.

Figure 1: Simulation model of a Kaplan turbine runner, a type of water turbine often used in river power plants. This case study describes issues of visualizing results of CFD simulations for the design of such water turbines.

When flow exits the computational domain at places neighboring another channel, streamlines or particle paths have to be continued by entering the domain again at the opposite side. Except for Visual3, none of the major flow visualization packages offers this feature.

To study the interaction between different elements of a turbine, such as runner and draft tube, it is essential to connect two individually calculated parts for visualization. Particle traces then have to continue in the mesh of the draft tube after exiting the rotating mesh of the runner, for example. Displaying flow in multiple blocks moving relative to each other is another feature not supported in most visualization systems.

Due to the limited support in existing flow visualization systems for these critical issues, software for turbomachinery flows

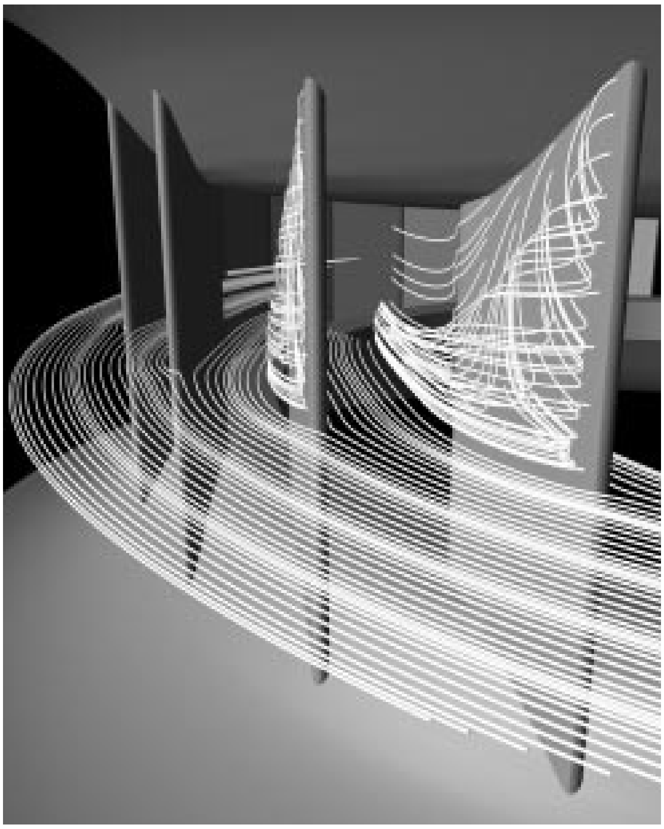

Figure 2: Example for turbomachinery flow visualization: Detaching flow at the stay vane blades

was developed at ETH Zürich, in collaboration with the turbine manufacturer Sulzer Hydro, Ltd. This program is used during the design phase for explorative flow visualization by the engineers at Sulzer Hydro as well as to produce high-quality pictures and video animations for showcasing to customers results and design decisions. An overview of this software and some selected topics such as streamline integration can be found in [2].

While streamlines and animated particles can be effective, their usability highly depends on carefully selected starting points. To improve this situation, current research (e.g. [4], [5]) has started to focus on automatic extraction of individual flow features.

3 Extraction of Flow Features: Vortices

In our application, the most important flow structures are vortices, which are to be avoided or minimized during design. A *vortex* is a region of swirling flow; unfortunately, there is no agreement on any more formal definition. It is important, however, to realize that not all regions of curving flows are considered to be vortices. If the flow is smoothly bent by an array of blades but does not detach, no vortices are present in the flow.

Currently available techniques to detect vortices are summarized below.

Vorticity magnitude

The simplest idea to find vortices automatically is to look for regions of high *vorticity magnitude. Vorticity* is defined as the curl of the flow vector field. However, the occurrence of curl in a flow does not necessarily imply any "swirling motion". An obvious example is sheared flow of constant direction but varying flow speed. Although all streamlines are straight lines, a curl perpendicular to the flow results.

Helicity

Helicity is defined as the projection of the curl onto the flow vector, thus remaining zero in regions of straight sheared flow. If the sheared flow is not straight, however, as is usually the case for turbomachinery, the results still turn out be of no practical use. Sheared and bent flow, or flow of uneven acceleration, influences the curl in a way that does not allow to visually associate isosurfaces of helicity with apparent vortices.

Minima of pressure

A technique which is frequently used is looking for regions of low pressure. It is argued that to keep the flow in a circular motion, a pressure gradient with low pressure in the center of a vortex is needed to generate a centripetal acceleration. This definition appears to work nicely for many applications with free-flowing fluids. However, by looking at the pressure distribution in turbomachines, which are characterized by large pressure gradients and strongly guided flows, it is obvious that a pressure minimum does not appear in the vortex centers. We attribute this to strong guidance of the flow by bent channels or widening tubes, causing pressure variations orders of magnitude greater than those caused by the small slow-rotating vortices.

λ_2-Method

A more sophisticated method is suggested in [6], which also lists some other definitions of a vortex and extensively discusses counterexamples where those methods fail, including vorticity and minima of pressure. The method defines vortices as regions where two of the three (real) eigenvalues of the symmetric matrix $S^2+\Omega^2$ are negative. S and Ω are the symmetric and antisymmectric part of the Jacobian of the vector field, respectively. However, we found this condition to hold for almost all the space in the turbine channel. Hence, individual vortices within the channel are not resolved very well.

Minor variations of methods mentioned above are listed in [6] and [7]; since they use similar paradigms, they share the same weaknesses when applied to our data.

Searching for vortex cores

Looking for *regions* containing vortices did not prove helpful for our application. More promising are methods designed specifically to find the core lines of vortices. Since a vortex is usually imagined as a line segment around which the flow swirls, it seems useful to look for these vortex center lines, which are called the *core* or *skeleton* of a vortex.

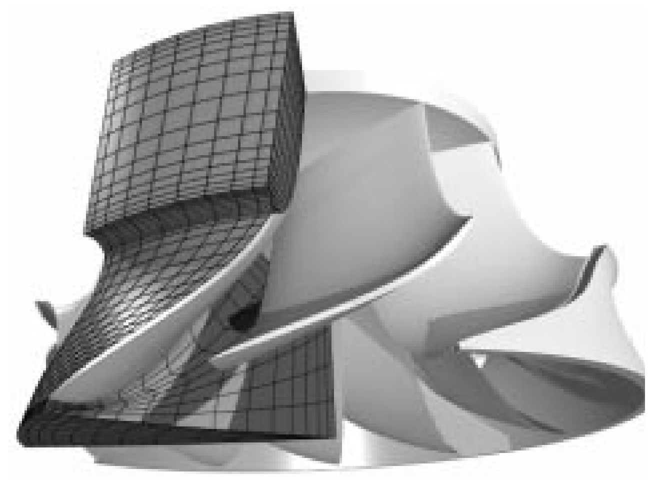

Figure 3: Typical mesh fitted into a channel of a runner

Streamlines from 3D critical points

Few publications deal explicitly with finding vortex core lines. [8] claims that vortex cores can be found by looking for *3D critical points* (isolated points in the flow where the velocity is zero) and integrating *streamlines* from there starting in the direction of eigenvectors if the critical point is of a spiraling type. Although reported successful for other application areas, this does not seem to be the case for our turbomachinery data. The problem starts with the fact that hardly any 3D critical points occur within the flow. It is possible to imagine a flow in a tube which has a constant forward component but is straight at first and starts a swirling motion later. This flow clearly contains a vortex in the second part, but obviously does not contain any critical point where the center could be traced from. In addition, critical points associated with vortices could lie outside the computational domain. Due to the numerical instability of streamline integration near the vortex center, it is difficult to track the center even if a suitable starting point is present.

Method of Singer and Banks

A method presented in [7] suggests integrating *streamlines of the vorticity field*. Addressing the issue of very unstable integration near the vortex center, they propose a prediction-correction algorithm that corrects each step along the vortex core in direction of the flow vorticity by moving to a local *minimum in pressure* in a plane perpendicular to the current vorticity. This method is the first to combine information from the vector field with the scalar field of pressure. However, we already established that for turbomachinery flows, pressure distribution is governed mainly by the guidance of the flow and not by vortices. Therefore, the correction step would fail here usually by moving toward the boundary of the computational domain.

Curl parallel to flow

The two contradictory suggestions that vortex cores are streamline segments of the vorticity field (e.g. Singer/Banks), or streamlines of the vector field itself (e.g. Globus et al.), fuels the idea they are neither and to define the vortex core line as the place where *vorticity (curl) is parallel to the vector.* After implementing this, we found the resulting line segments sometimes form coherent structures acceptable as vortex cores. Most fields, however, did not exhibit the expected features.

Real valued eigenvector parallel to flow

Since all methods discussed above (except the ones using pressure) are based on some elements of the Jacobian of the vector field but not on higher order derivatives, a locally linear model for the flow field is always assumed at each point. Close inspection of linear flow fields containing a swirling motion reveals that the direction of the center line corresponds not to the curl, but rather to the only *real valued eigenvector* of the Jacobian. This leads to a method of looking for places where the Jacobian has only one real valued eigenvector, and where this is *parallel to the flow.* The same idea was also suggested independently by Haimes and Sujudi [9], who formulated that the local velocity minus its component parallel to this eigenvector must be zero.

For a linear vector field, this appears to be the correct solution. Unfortunately, the line segments produced by this method still did not show any relation to observed vortices for our turbomachinery data.

Curved line vortex: a mathematical model

To study this problem and discard any effect of the underlying discrete grid as well as numerical problems, we defined mathematically a flow field consisting of a single vortex. When the vortex center is chosen as a straight line, all streamlines are helical except for the vortex core, which is straight. In this case, all methods involving curl or the eigenvector mentioned above work out perfectly; the center can be found by looking for locations where the flow is parallel to the curl.

After bending this helical flow into a circle of radius R and restoring the solenoidal property, we get a mathematical model for a curved vortex. The resulting bent helical flow field is shown in Fig. 4 and defined as

$$\vec{v} = \left\langle -\frac{\gamma y}{r} - \frac{\omega z x}{r^2},\ \frac{\gamma x}{r} - \frac{\omega z y}{r^2},\ \omega\left(1 - \frac{R}{r}\right)\right\rangle$$

$$\text{where} \qquad r = \sqrt{x^2 + y^2}$$

Parameters of this field are the rotational component ω, the forward motion γ along the vortex core, and the radius R to which the vortex core is bent.

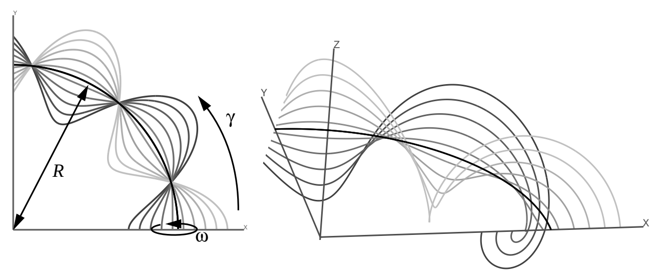

Figure 4: Two views of a model for a bent line vortex. Each streamline is a bent helix. The vortex core, itself a streamline, is the black circle.

For this bent helical flow, we expect to find simply the circle of radius R (dark line in Fig. 4) as the core line, since all remaining streamlines swirl around that curve. For very large R, the vortex is nearly straight and most methods work fine. Also, even for small R, things work as long as ω is large enough. However, if a slowly swirling vortex is bent to a large curvature (small R), then curl and eigenvector show an additional component indicating the rotation around the Z axis introduced by the bending. This causes locations where the curl or eigenvector is parallel to the flow to move inward to a radius smaller than R, as shown in Fig. 5.

4 Conclusions and Future Work

As the overview in this paper indicates, none of the commonly used vortex finders seems to be successful for typical turbomachinery data.

A major difference with most other areas of CFD applications is that the flow is confined to a channel, which is often strongly bent and changes its cross-section. Therefore, pressure distribution and directional changes of the flow are governed mainly by channel geometry rather than by the presence of vortices.

Furthermore, design of hydraulic machines has reached a level where vortices can be suppressed to a relatively high degree. As a consequence, vortex finding algorithms are confronted with weak structures in comparison with other fields of CFD. Streamlines thus often do less than a full revolution around the vortex core.

Given the limitations of current techniques, our goal is to develop a refined method to identify vortices in practical turbomachinery flows. We expect this new algorithm to find the apparent vortex core in the bent helix model introduced above.

In the projected feature-based visualization system, such a method will serve as a low-level feature extraction. We assume any such method will produce artifacts due to the discretization or noise in the flow data. Also, some very small features might be considered unimportant by the users. Therefore, we plan processing the output of the raw feature detector by high-level methods of computer vision and pattern recognition.

Using these building blocks, we intend to assemble a functional feature-based visualization system for turbine design. By automating the tedious search for important flow structures, the visualization cycle can be shortened, which is highly desirable in the context of industrial simulations.

Acknowledgments

This project is supported by Sulzer Hydro, Ltd., Zürich (formerly Escher Wyss), and Swiss KWF Grant No. 2974.1.

Figure 5: The apparent vortex core is depicted by a circular stream surface around it. The place where real valued eigenvector is parallel to the flow is marked by the black line.

References

[1] Krivchenko, G.: *Hydraulic machines: turbines and pumps*. 2nd ed., ISBN 1-56670-001-9, CRC Press, 1994.

[2] Roth, M.: *Visualization of Turbomachinery Flow,* Technical Report, SCSC, ETH Zürich, 1996, available from http://www.scsc.ethz.ch/SV/turbo.

[3] Göde, E. and R. Cuénod: *Numerical Simulation of Flow in a Hydraulic Turbine*, Sulzer Tech. Review 4/89.

[4] Helman, J. L. and L. Hesselink: “Visualizing vector field topology in fluid flows”. *IEEE Computer Graphics and Applications*, 11(3):36--46, May 1991.

[5] van Walsum, T. and F. Post: “Selective Visualization of Vector Fields”. In *Tutorial on Visualization and Topology of Vector and Tensor Fields,* Visualization ‘95, October 1995.

[6] Jeong, J. and F. Hussain: “On the identification of a vortex”. *Journal of Fluid Mech.*, 285:69--94, 1995.

[7] Singer, B. and D. Banks: *A predictor-corrector scheme for vortex identification*. ICASE Report No. 94-11, NASA Langeley, Hampton VA, March 1994.

[8] Globus, A., C. Levit, and T. Lasinski: “A tool for visualizing the topology of 3D vector fields”. In *Proceedings of Visualization ‘91*, pages 33--39, San Diego, CA, October 1991.

[9] Haimes, R. and D. Sujudi: *Identification of swirling flow in 3D vector fields*. Tech. Report, Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA, 1995.

More information and example images are available on the conference video tape and on the World Wide Web at `http://www.scsc.ethz.ch/SV/turbo`.