A Predictor-Corrector Technique for Visualizing Unsteady Flow

David C. Banks*
Institute for Computer Applications in Science and Engineering
MS 132C, NASA Langley Research Center
Hampton, VA 23681-0001

Bart A. Singer**
High Technology Corporation
MS 156, NASA Langley Research Center,
Hampton, VA 23681-0001

Abstract

We present a method for visualizing unsteady flow by displaying its vortices. The vortices are identified by using a vorticity-predictor pressure-corrector scheme that follows vortex cores. The cross-sections of a vortex at each point along the core can be represented by a Fourier series. A vortex can be faithfully reconstructed from the series as a simple quadrilateral mesh, or its reconstruction can be enhanced to indicate helical motion. The mesh can reduce the representation of the flow features by a factor of one thousand or more compared with the volumetric dataset. With this amount of reduction it is possible to implement an interactive system on a graphics workstation to permit a viewer to examine, in three dimensions, the evolution of the vortical structures in a complex, unsteady flow.

*This research was supported by the National Aeronautics and Space Administration under NASA contract No. NAS1-19480 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA 23681-0001.

**Work supported by the Theoretical Flow Physics Branch at NASA Langley Research Center under contract NAS1-20059.

1 Introduction

In order to study the complex behavior of an unsteady (*i.e.*, time-varying) fluid flow, one might imagine being immersed within the flow but not disturbing it. One could then roam about the flow field, free to observe its development or to measure quantities of interest. This scenario is impossible in real life, of course. The physical presence of a human observer would change the very flow under inspection. However, a direct numerical simulation (DNS) of the flow produces all the relevant flow quantities that an appropriate visualization system would need in order to let a viewer navigate through the flow. In order to develop such an interactive system, one must (1) locate the salient structures within the three-dimensional flow data, (2) represent the structures geometrically, and (3) display them to the viewer, preferably at interactive frame rates of 20 updates per second or more.

What, exactly, are the important structures within an unsteady flow? Vortices are typically considered the most important structures in flow fields. Consider the effects of vortices over a range of spatial scales: large-scale vortices are responsible for hurricanes and tornadoes; medium-scale vortices affect the handling characteristics of an airplane; small-scale vortices are the fundamental building blocks of the structure of turbulent flow. Vortices control the dynamics of the flow in the sense that if they are removed the flow becomes quiescent. As an example, hairpin vortices are considered to be “a major sustaining flow structure involved in the perpetuation of turbulent boundary layers” [1]. Leonard [2] emphasizes that

> ...it is mathematically correct and often very convenient to consider inviscid fluid dynamics in terms of parcels of vorticity which induce motion on each other as an alternative to pressure-velocity considerations.

One would like, therefore, to visualize a flow by locating and displaying its vortices. This paper describes how a predictor-corrector technique can locate vortex structures in three-dimensional flow data [3] with enough data-reduction to store and animate them on a workstation.

The predictor-corrector technique is effective at locating vortices even in turbulent flow data. Simulating an unsteady flow may require hundreds or even thousands of time steps, each containing many megabytes of data. The vortices themselves may occupy significant subvolumes of the original volumetric data. A typical scientific workstation does not have adequate memory to store more than a few frames of the original data; data reduction is absolutely essential for the interactive display of time-varying vortices. The predictor-corrector scheme provides a terse, one-dimensional representation of vortex tubes, which offers significant reduction of the flow data. This benefit suggests the design of an interactive visualization system that can re-play the development of a computed flow while allowing a viewer to explore the vortex shapes with a graphics workstation.

The paper is organized as follows. Section 2 presents a survey of other techniques that attempt to identify vortices. Section 3 presents our predictor-corrector scheme and discusses some of the programming considerations that are necessary to make the scheme efficient. Section 4 describes how we calculate the cross-sections of the vortex tube and how we represent them in a compressed fashion using Fourier analysis. In section 5 we show how the vortex skeletons, together with an efficient representation of the cross-sections, offer substantial data-reduction in representing features of a flow. We describe the process of reconstructing the vortex tubes from the compressed format and report on the successful development of an interactive graphical system based on these techniques.

2 Survey of Identification Schemes

The term “vortex” connotes a similar concept in the minds of most fluid dynamicists: a helical pattern of flow in a localized region. There are mathematical definitions for “vorticity” and “helicity,” but vortical flow is not completely characterized by them. For example, a shear flow exhibits vorticity at every point even though there is no vortical motion. A precise definition for a vortex is difficult to obtain — a fact supported by the variety of efforts outlined below.

2.1 Ideal Definition

It is surprisingly difficult to establish a definition of a vortex that is robust enough to locate all the coherent structures that a flow physicist would consider to be vortices. Robinson [4] suggests the following working definition for a vortex.

> A vortex exists when instantaneous streamlines mapped onto a plane normal to the vortex core exhibit a roughly circular or spiral pattern, when viewed from a reference frame moving with the center of the vortex core.

Robinson [5] and Robinson, Kline, and Spalart [6] use the above definition to confirm that a particular structure is, in fact, a vortex. Unfortunately, this definition requires a knowledge of the vortex core before one can determine whether something is a vortex. The definition, therefore, does not lend itself to a convenient algorithm for detecting vortices.

2.2 Isovalue of a Scalar Field

Is there a scalar value that can be easily derived from flow quantities such that a single isovalue yields surfaces surrounding the vortical structures? One might imagine a scalar field that attains a non-negative value in the interior of the vortices but attains negative values elsewhere. The zero-valued isosurfaces would define the boundaries of the vortices. Several attempts have been made to locate vortices as isosurfaces of scalar quantities.

Low Pressure

Robinson and his colleagues find that elongated low-pressure regions in incompressible turbulent flows almost always indicate vortex cores. Isosurfaces of low pressure are usually effective at capturing the shape of an individual vortex (fig. 1a), especially if the flow field contains no solid bodies. Pressure surfaces become indistinct where vortices merge, however, and a high-quality image can easily require thousands of triangles to create the surface. The need to compress the representation becomes acute when visualizing time-varying data.

Eigenvalues of the Velocity Gradient

Chong, Perry, and Cantwell [7] define a vortex core as a region where the velocity-gradient tensor has complex eigenvalues. In such a region, the rotation tensor dominates over the rate-of-strain tensor. Soria and Cantwell [8] use this approach to study vortical structures in free-shear flows. At points of large vorticity, the eigenvalues of the velocity-gradient matrix are determined: a complex eigenvalue suggests the presence of a vortex. This method correctly identifies the large vortical structures in the flow. However, the method also captures many smaller structures without providing a way to link the smaller vortical volumes with the larger coherent vortices of which they might be a part (fig. 1b).

2.3 Geometry of the Vortex Core

Instead of defining the bounding surface of a vortex, some researchers have sought ways to locate the one-dimensional core through the vortex center. Various schemes for determining the geometry of a vortex core are described below.

Vorticity Lines

Vorticity is a vector quantity proportional to the angular velocity of a fluid particle. It is defined as

$$\omega = \nabla \times \mathbf{u}$$

where **u** is the velocity at a given point. Vorticity lines are integral curves of vorticity. Moin and Kim [9] [10] use vorticity lines to visualize vortical structures in turbulent channel flow. The resulting curves are extremely sensitive to the choice of initial location $\mathbf{x}_0$ for the integration. As Moin and Kim point out [9],

> If we choose $\mathbf{x}_0$ arbitrarily, the resulting vortex line is likely to wander over the whole flow field like a badly tangled fishing line, and it would be very difficult to identify the organized structures (if any) through which the line may have passed.

Fig. 6 illustrates the potential for vortex lines to create a tangle [10]. To avoid such a confusing jumble, they carefully select the initial points. However, Robinson [5] shows that even experienced researchers can be surprisingly misled by ordinary vorticity lines. The problem with vorticity lines in a shear flow is not just that numerical techniques of integration propagate error. Even an errorless analytic integration fails to follow a vortex core that is not aligned in the direction of mean shear. Jiminez points out that a vortex tube "does not have vorticity perfectly aligned along its axis [core], nor does a given vortex line necessarily remain within it over its entire length" [11]. In order for an integral curve through a vector field to coincide with the core, the vector field must be aligned with the core.

Vorticity and Enstrophy

Jiminez *et al.* propose a scheme for tracing vortex cores that shares the spirit of our technique [11]. They consider points of maximum enstrophy (squared magnitude of vorticity) to lie along vortex cores. Given such a point, they integrate along the core using a two-step process. The first step is to follow the vorticity to the next grid plane. Then, within that plane, they inspect the nearest four grid points and select the one with the largest enstrophy. The method marches from grid point to grid point within the volume. They applied the technique to locate vortices within isotropic turbulence. Near the wall of a shear flow, there is a large magnitude of vorticity even when no vortices are present. Thus the technique is not well-suited to the task of identifying vortices in a shear flow. Instead of consulting the enstrophy, our technique uses pressure gradients for the corrector phase. In addition, we use higher-order interpolation in order to resolve the vortex core between grid points.

Curvature and Helicity

Yates and Chapman [12] carefully explore two definitions of vortex cores. Unfortunately, the analyses and conclusions for both definitions are appropriate only for steady flows. By one definition, the vortex core is the line defined by the local maxima of normalized helicity (the dot product of the normalized velocity and vorticity). Fig. 1c shows an isosurface of constant helicity. Notice that the surface fails to capture the "head" on the upper-right side of the hair-

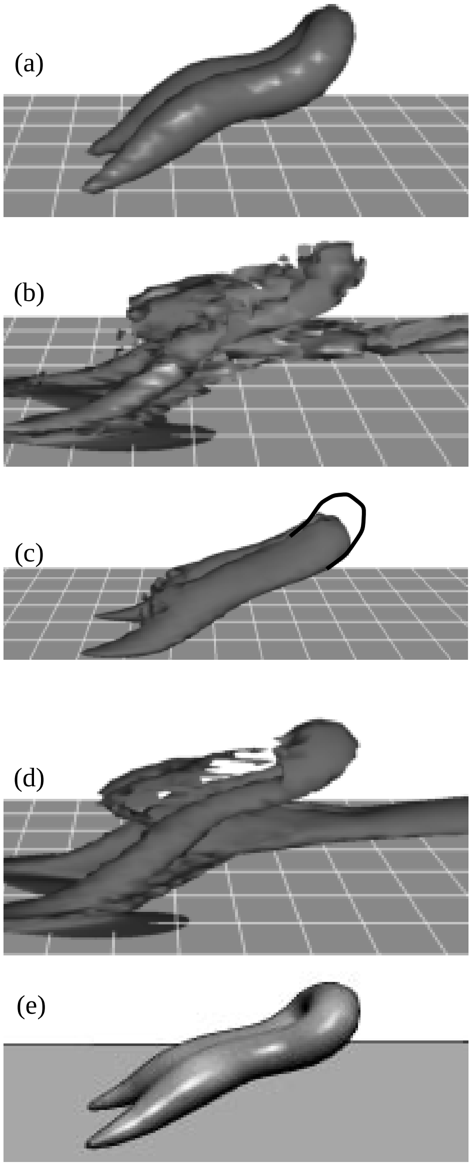


Figure 1.
Different schemes used to identify a vortex. Each image visualizes the flow at the same time step. From top: (a) isosurface of constant pressure; (b) isosurfaces of complex-valued eigenvalues of the velocity-gradient matrix; (c) isosurface of constant helicity (dark line indicates missing vortex head; (d) isosurfaces of constant vorticity; (e) our predictor-corrector technique with Fourier cross-sections.

pin vortex. This shows that the local maxima fail to follow the core. In the other definition, a vortex core is an integral curve that has minimum curvature. If there is a critical point on a vortex core, then that point must be a spiral-saddle. The eigenvector belonging to the only real eigenvalue of the spiral-saddle corresponds, locally, to an integral curve entering or leaving the critical point. By integrating this curve, the entire vortex core may be visualized [13]. For our particular flow data, however, we find that these curves (as calculated by FAST, the Flow Analysis Software Toolkit [14]) can miss the vortex completely. It may be that the critical points are not sufficiently resolved in the flow data for this technique to capture the cores; in that case the amount of data must be more finely sampled in order to locate vortex cores with this technique, at the expense of increasing the data storage and slowing the numerical simulation. Since the technique is derived for steady flows, it may be that even with finer sampling the cores would not be detected.

User-guided Search

Bernard, Thomas, and Handler [15] use a semi-automated procedure to identify quasi-streamwise vortices. Their method finds local centers of rotation in user-specified regions in planes perpendicular to the streamwise direction of a turbulent channel flow. Experienced users can correctly find the critical vortices responsible for the maintenance of the Reynolds stress. Their method captures the vortices that are aligned with the streamwise direction, but in free-shear layers and transitional boundary layers, the significant spanwise vortices go undetected. Because it depends heavily on user intervention, the process is tedious and is dependent upon the individual skill of the user.

2.4 Vortex Shape Detection

Vortices exhibit the characteristic shape of elongates tubes. Below we describe two identification schemes that exploit this shape-knowledge to locate vortices.

Cylinder With Maximum Vorticity

Villasenor and Vincent [16] present an algorithm for locating elongated vortices in three-dimensional time-dependent flow fields. They start from a seed point and compute the average length of all vorticity vectors contained in a small-radius cylinder. They repeat this step for a large number of cylinders that emanate from the seed point. The cylinder with the maximum average becomes a segment of the vortex tube. They use only the magnitudes (not the directions) of vorticity; as a consequence the algorithm can inadvertently capture structures that are not vortices.

Vorticity and Vortex Stretching

Zabusky *et al.* [17] use vorticity $|\omega|$ and vortex stretching $|\omega \cdot \nabla \mathbf{u}| / |\omega|$ in an effort to understand the dynamics of a vortex reconnection process. They fit ellipsoids to the regions of high vorticity. Vector field lines of vorticity and of vortex stretching emanate from the ellipsoids. In flows with solid boundaries or a mean straining field, the regions with large vorticity magnitudes do not necessarily correspond to vortices (fig. 1d); hence, the ellipsoids do not always provide useful information.

2.4 Summary of Survey

Some of the above techniques share a simple property: they aim to capture vortices by consulting a scalar field derived from certain flow quantities. Without having a canonical scalar definition of a vortex, one should only treat these techniques as heuristics. The experienced flow physicist is apt to identify vortices in a flow field based on his own knowledge of the flow characteristics, even if this judgment is at odds with one of the above definitions.

Notice that the pressure surface in fig. 1a is smoother than the isosurfaces in figs. 1b and 1d. The latter surfaces are based on derivatives of local flow quantities and are therefore subject to numerical error due to differentiation. In contrast, pressure is obtained by integration which filters out noise. It may be difficult, in general, to develop a robust technique for locating vortices if one appeals to quantities derived through repeated differentiation.

The isosurfaces that define the boundaries of the vortices are unstructured sets of polygons. If one wishes to archive the vortex geometry over the course of hundreds or thousands of time steps, the isosurfaces can require large quantities (hundreds of gigabytes) of storage. While techniques exist for decimating isosurfaces, such decimation is not a trivial task. By contrast, the vortex cores can be represented economically by one-dimensional curves or polylines. For vortices in the shape of elongated tubes, skeleton curves together with a radius function provide a natural and efficient representation.

The methods in the survey all experience success in finding vortices under certain flow conditions. But all of them have problems capturing vortices in unsteady shear flow and/or representing them in the most economical way. We were led, therefore, to develop another technique which could tolerate the complexity of a transitional flow (from laminar to turbulent) and would offer substantial data reduction. For comparison, fig. 1e shows the results of applying our predictor-corrector method with Fourier cross-sections.

3 The Predictor-corrector Method

We now present the heart of our vortex identification scheme: the velocity-predictor, pressure-corrector method. The method was designed to capture elongated vortices (shaped like spaghetti) rather than broad vortex sheets (shaped like lasagna). The method, like the techniques in the survey, relies on heuristics: if a point is in a vortex, then the point is expected to possess certain properties. Possessing those properties does not guarantee that a point is in a vortex, however. The method is designed to locate the core of the vortex, rather than the surface bounding the entire vortex. The method uses vector quantities for both the predictor and the corrector steps and uses scalar values as thresholds.

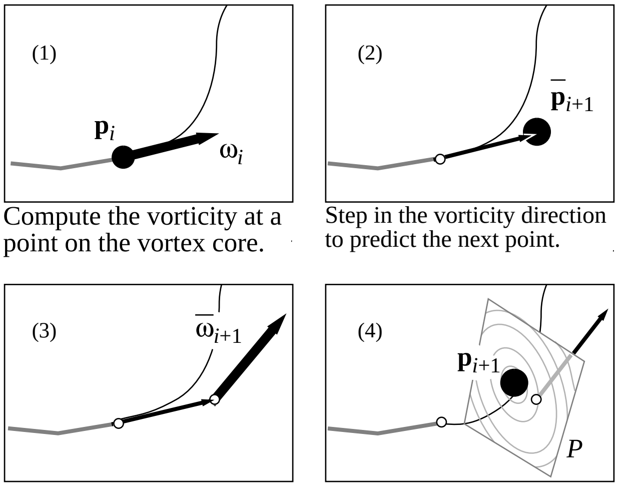
The predictor-corrector method produces an ordered set of points (the skeleton) that approximates a vortex core. Associated with each point are quantities that describe the local characteristics of the vortex. These quantities may include the vorticity, the pressure, the shape of the cross-section, or other quantities of interest. The method produces lines that are similar to vorticity lines, but with an important difference. Whereas vorticity is a mathematical function of the instantaneous velocity field, a vortex is a physical structure with coherence over a region of space. In contrast to vorticity lines, which may wander away from the vortex cores, our method is self-correcting: line trajectories that diverge from the vortex core reconverge to the center.

In this section we discuss the procedure used to find an initial seed point on the vortex skeleton. We then explain the predictor-corrector method used for growing the vortex skeleton from the seed point. Finally, we address how to terminate the vortex skeleton.

3.1 Finding a Seed Point

Vorticity lines begin and end only at domain boundaries, but actual vortices have no such restriction. Therefore we must examine the entire flow volume in order to find seed points from which to grow the vortex skeletons. We consider low pressure and a large magnitude of vorticity to indicate that a vortex is present. Low pressure in a vortex core provides a pressure gradient that offsets the centripetal acceleration of a particle rotating about the core. Large vorticity indicates that such rotation is probably present. These are heuristic arguments: vortical motion is presumed to be sustained by pressure gradients and to be indicated by vorticity. It is certainly possible to have low pressure (downstream of an obstacle, for example) or large vorticity (in a shear flow, for example) without a vortex present. Even so, the combination of the two is a powerful indicator of a vortex.

In our implementation, the flow field (a three-dimensional rectilinear grid) is scanned along planes perpendicular to the streamwise direction. The scanning direction affects the order in which vortices are located, but not the overall features of the vortices. In each plane, the values of the pressure and the vorticity magnitude are checked against threshold values of these two quantities. A seed point is a grid point that satisfies the two threshold values. Since new vortex tubes can emerge at any time, we re-scan the 3D grid anew to locate seed points at each time step. In a more steady flow, one could advect seed points from one time step as initial guesses at the next time step. Threshold values can be chosen *a priori*, or they can be a predetermined fraction of the extrema. The thresholds of pressure and vorticity-magnitude can be fairly strict. It is not necessary to include every point of the vortex core in the set of candidate seeds; it suffices to capture a single one. Even so, if the threshold of pressure is too low some structures will be missed entirely. We selected thresholds of pressure and vorticity that capture the essential structures in the flow field.



Compute the vorticity at a point on the vortex core.

Step in the vorticity direction to predict the next point.

Compute the vorticity at the predicted point.

Correct to the pressure min in the perpendicular plane.

Figure 2.
Four steps of the predictor-corrector algorithm.

We next refine the position of the seed point so that it is not constrained to lie on the grid. The seed point moves in the plane perpendicular to the vorticity vector until it reaches the location of the local pressure minimum. From this seed point we develop the vortex skeleton in two parts, forward and backward, to reach the endpoints of the vortex tube.

3.2 Growing the Skeleton

The predictor-corrector algorithm is illustrated in the schematic diagrams of fig. 2. The details for continuing the calculation from one point to the next are indicated by the captions. Steps 1-2 represent the predictor stage of the algorithm. The corrector stage is summarized by steps 3-4.

Once a seed point has been selected, the skeleton of the vortex core can be grown from the seed. The next position of the vortex skeleton is predicted by integrating along the vorticity vector (fig. 2, top) which is equivalent to Euler integration of a vorticity line. The predicted point typically misses the vortex core.

Next we invoke the heuristic that centripetal acceleration within a vortex is supported by low pressure at the core. In a plane perpendicular to the core, the pressure minimum is expected to coincide with the point where the core pierces the plane. The predicted point must be corrected to the pressure minimum in the plane that (1) is perpendicular to the core and (2) contains the predicted point. The location of the nearest core point is the unknown quantity, so condition (1) can only be satisfied approximately. We approximate the desired plane by choosing the plane perpendicular to the vorticity vector (fig. 2, bottom).

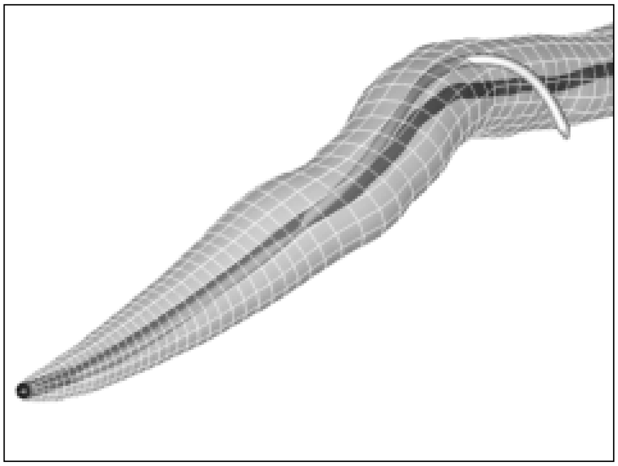


Figure 3.
Vorticity line (light) compared to predictor-corrector line (dark). Note that the vorticity line exits from the vortex tube while the predictor-corrector skeleton line follows the core.

Individually, integral curves of vorticity or of the pressure gradient are each unreliable at capturing vortex cores. Section 2.2 points out the problems with vorticity lines. The pressure gradient does not follow the core either; moreover, a vortex may have several distinct pressure minima in its interior, which would require piecewise integration of the gradient in order to connect the components of the core. Remarkably, the combination of the vorticity and the pressure gradient provides a robust method of following the vortex core. The continuous modification of the skeleton point lessens the sensitivity to both the initial conditions and the integration details.

The effectiveness of the predictor-corrector scheme is illustrated in fig. 3, in which data from the direct numerical simulations of Singer and Joslin [18] are analyzed. The transparent vortex tube (a portion of a hairpin vortex) is constructed with data from the full predictor-corrector method. Its core is indicated by the darker skeleton. The lighter skeleton follows the uncorrected integral curve of the vorticity. It is obtained by disabling the corrector phase of the scheme. The vorticity line deviates from the core, exits the vortex tube entirely, and wanders within the flow field. By appealing to Robinson’s ideal definition of a vortex we are able to confirm that the predictor-corrector skeleton is the one that follows the core. The velocity fields around the skeleton are consistent with nearly-circular streamlines in Robinson’s characterization; those around the vorticity line are eventually not.

3.3 Terminating the Vortex Skeleton

Vorticity lines extend until they intersect a domain boundary, but real vortices typically begin and end inside the domain. Therefore, the algorithm must always be prepared to terminate a given vortex skeleton. A simple condition for termination occurs when the vortex cross-sec-

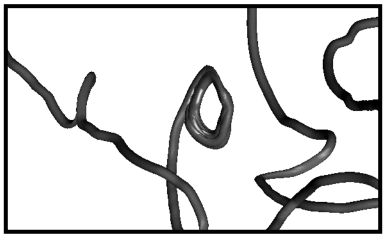


Figure 4.
Vortex skeleton at time 194.4 located by the predictor-corrector method. Note the spiral in the center.

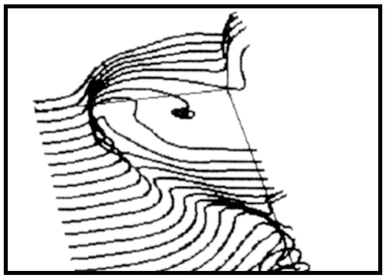


Figure 5.
Vorticity lines in a shear layer near a wall. Note the spiral near the top. From Jiminez and Moin, *JFM* v. 225, p. 235. © Cambridge University Press 1991. Reprinted with the permission of Cambridge University Press.

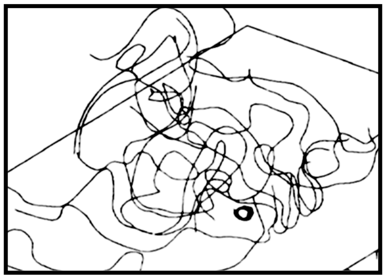


Figure 6.
Tangle of vorticity lines in a turbulent flow. Note the spiral near the bottom. From Kim and Moin, *JFM* v. 162, p. 343. © Cambridge University Press 1986. Reprinted with the permission of Cambridge University Press.

tion (discussed in section 4) has zero area. As fig. 3 shows, the reconstructed vortex tubes taper down to their endpoints, where the cross-section vanishes. The predictor-corrector method is not guaranteed to terminate. On rare occasions the skeleton can enter a nearly-circular loop (fig. 4). We have observed this undesirable phenomenon in a small fraction of the skeletons. The spirals seem to occur in the vicinity of vortex junctures, but we have no reason to believe that the vortex core truly enters a closed loop. There are examples of this phenomenon in other works, although those examples do not receive any particular discussion. Figs. 5 and 6 show similar situations in other simulated flows [19] [10] where a vorticity line enters a tight spiral. In order to guarantee termination, we exploit our knowledge of the spatial extent of the 3D computational domain and limit the total arclength along a skeleton to be about twice the longest grid dimension. By guaranteeing termination in this way, we find that an average time step requires about 1400 Cray-seconds in calculating the 3D numerical simulation of the flow and about 20 Cray-seconds in identifying the vortex cores and calculating their cross-sections.

3.4 Filaments That Connect Vortex Tubes

Sometimes it is useful to continue the skeleton beyond the end of the vortex tube. For instance, if a low-intensity region exists between high-intensity regions of the same vortex, then the low-intensity region might not satisfy the criteria for a finite cross section. One would like to see the connective filament between two strong vortices even if the connection does not satisfy the requirements for a non-zero cross-section. The criteria for determining the cross-section can be made more generous in order to capture the connection, but this strategy does not solve the problem: in addition to capturing the weak connective vortex, we will also capture unwanted low-intensity structures that may themselves possess regions that are weaker still. Our resolution of this problem exploits the asymmetric nature of the predictor-correction method.

Because the predictor-corrector method follows the core of a vortex regardless of the criteria used to define the vortex cross section, the vortex skeleton can be extended even when the cross-sectional area of the vortex is equal to zero. The vortex of interest may either re-intensify or dissipate; if the vortex re-intensifies then the continuation of the skeleton line will provide a link between the two more-intense regions of the vortex. This link can be visualized as a thread that connects the two disjoint regions. On the other hand, if the vortex dissipates, the continuation of the skeleton line will wander through the flow field and eventually either intercept a domain boundary or enter a new vortex. If a domain boundary is reached, then the segments of the skeleton that lie outside the last-found vortex (having non-zero cross-section) are discarded. Similarly a potentially connective filament is discarded if it enters a new vortex from the side, rather than through one of the vortex endpoints.

To determine whether a new-found region of finite cross section is a continuation of the original vortex or an entirely different vortex, we march the predictor-corrector scheme backwards for the same number of steps taken since the previous region of nonzero cross section was exited. Some possible scenarios are illustrated in figs. 7-9. In fig. 7, the skeleton line leaves the first vortex tube at point $\mathbf{p_1}$ and continues for n steps until it encounters the second vortex tube at point $\mathbf{p_2}$. The predictor-corrector scheme is then marched backwards n steps from $\mathbf{p_2}$ to $\mathbf{p_3}$. The distance between points $\mathbf{p_1}$ and $\mathbf{p_3}$ is small relative to the distance between $\mathbf{p_1}$ and $\mathbf{p_2}$ (a 10-percent criterion is used); hence, the link between $\mathbf{p_1}$ and $\mathbf{p_2}$ is most probably a low-intensity vortex, and we retain the connective thread between these vortex tubes.

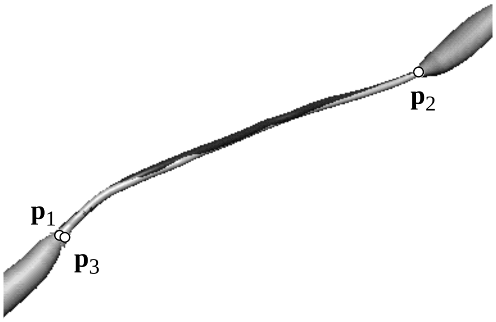


Figure 7.
Forward integration from p1 to p2 gives approximately the same path as reverse integration from p2 to p3. Points p1 and p2 are therefore connected by a weak vortex.

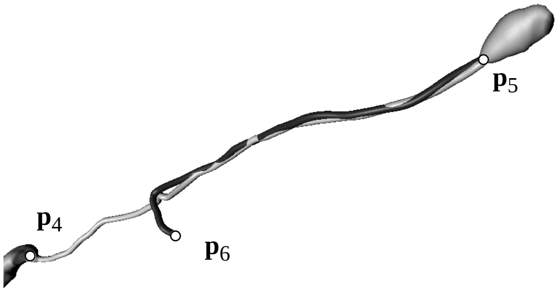


Figure 8.
Forward integration from p4 to p5 differs markedly from reverse integration from p5 to p6. The two vortex tubes are not connected; the core of the vortex on the left terminates at p4.

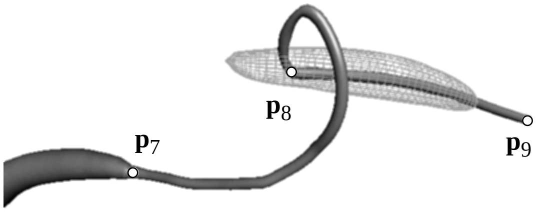


Figure 9.
Integration from p7 to p8 intersects side of vortex tube (wireframe). Reverse integration from p8 to p9 follows axis of the new vortex, away from original tube. The vortex tubes are not connected.

However, in fig. 8 the vortex tube dissipates beyond point $\mathbf{p_4}$, and the continuation of its skeleton lacks clear direction and wanders through the flow field. The line intercepts another vortex tube at $\mathbf{p_5}$ after *m* steps. The predictor-corrector method is marched backwards *m* steps from $\mathbf{p_5}$ to $\mathbf{p_6}$. Initially, the reverse integration retraces the forward integration, but halfway between $\mathbf{p_5}$ and $\mathbf{p_6}$ the two lines diverge rapidly and become uncorrelated. The distance from $\mathbf{p_4}$ to $\mathbf{p_6}$ is a large fraction of the distance from $\mathbf{p_4}$ to $\mathbf{p_5}$, so the algorithm concludes that the vortex tube intersected at $\mathbf{p_5}$ is different from the vortex tube that ends at $\mathbf{p_4}$. The points on the vortex skeleton line that connect the two tubes are discarded, and the vortex skeleton is terminated.

Finally, in fig. 9, the continuation of the skeleton line of the vortex tube that ends at point $\mathbf{p_7}$ intersects the side of another vortex tube (shown as a wireframe) and is immediately carried to the pressure minimum at $\mathbf{p_8}$. The reverse integration for this case follows along the axis of the new vortex tube away from the original vortex. The point $\mathbf{p}_9$ is far from $\mathbf{p}_7$; hence, the two vortex tubes are distinct from each other and the line connecting them is discarded.

3.5 Implementation Details

Optimal performance of the predictor-corrector technique requires careful attention to implementation details. This section addresses issues that are important to the successful use of the method. It is not exhaustive; additional details are provided by Singer and Banks [20].

Eliminating Redundant Seeds and Skeletons

Recall that seed points are chosen based on pressure and vorticity-magnitude, allowing multiple seeds to generate a given vortex core. Sampling every grid point produces an overabundance of seed points and hence a multitude of nearly-coincident vortex skeletons (fig. 10). These skeletons each follow the same core, sampling it at different locations; yet one representative skeleton suffices. The redundancies are eliminated when points inside a tube are excluded from the pool of future seed points. We accomplish this by flagging any 3D grid cell in the computational domain that lies within a spherical volume of a skeleton point. The constant term of a Fourier representation of the cross-section's radius (see section 4.2) is taken to be the radius of the spherical volume. A future candidate seed is ignored if it lies in a flagged cell.

Eliminating Spurious Feeders

A seed near the surface of the vortex tube can produce a “feeder” vortex skeleton that spirals toward the vortex center. Intuitively, these seeds lie within grid cells that should have been flagged but were missed because they lie slightly outside the spherical volumes of exclusion. Examples of these feeders are illustrated in fig. 11. We eliminate feeders by taking advantage of the fact that the predictor-corrector method is convergent to the vortex core. A feeder skeleton, begun on the surface of the tube, grows toward the core; by contrast, a skeleton growing along the core does not exit through the surface of the tube. To validate a candidate seed $\mathbf{p}_0$, we integrate forward n steps to the point $\mathbf{p}_n$ and then backward again by n steps. If we return

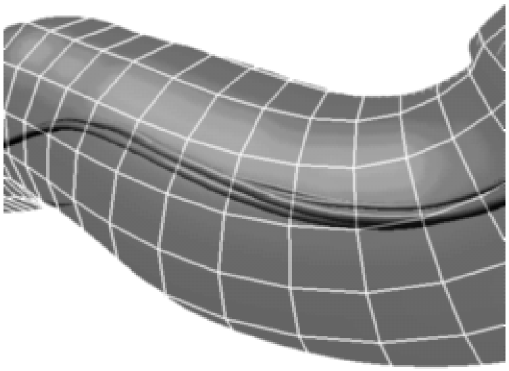


Figure 10.
Multiple realizations of the same vortex tube from different seed points. Each seed point generates a slightly different skeleton line, although all the skeletons remain close to the vortex core.

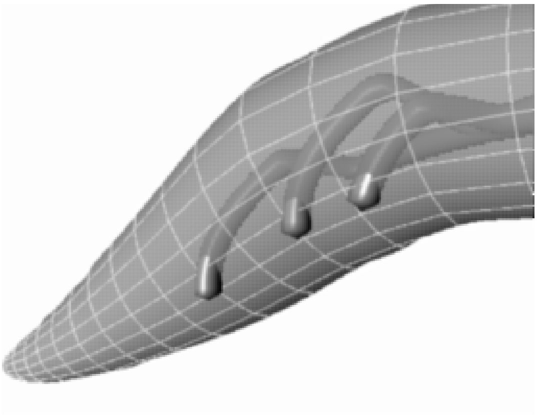


Figure 11.
Feeders merge with a large-scale hairpin vortex. Three points that satisfy the threshold criteria lie on the edge of the vortex tube. Their trajectories curve inward toward the core and then follow the main skeleton line.

very close to $\mathbf{p}_0$ then the candidate was a "true" seed point. This is the same reverse-integration strategy that is used for establishing that a filament actually connects two vortical regions.

Numerical Considerations for Interpolation

Neither the predictor nor the corrector step is likely to land precisely on a grid point; hence, we must interpolate the pressure and vorticity within the flow field. A linear approximation of the pressure gradient (the corrector step) will possess minima only at grid points. A three-point quadratic interpolation can produce minima within grid cells, but a three-point interpolation within a cell introduces bias toward one side or the other. To reduce any bias from the interpolation, we use a four-point Lagrange interpolation (found in textbooks on numerical computation) in each of the three coordinate directions. The high-order interpolation is justified by the accuracy of the numerical simulation, which is spectral in the spanwise and wall-normal directions (Fourier and Chebysheff, respectively) and fourth-order in the streamwise direction. The interpolation scheme works quite well, although it is the most expensive step in our implementation.

The interpolation scheme makes the predictor-corrector method at least first-order accurate: skeleton points are located to within the smallest grid dimension. This ensures that, on data sets with well-resolved vorticity and pressure, the method successfully locates vortex cores.

Predictor Step

The vorticity integration can be performed with a variety of methods. First, we used a fourth-order Runge-Kutta approach. This produced satisfactory results; however, step-size optimization was difficult to automate. Instead, we developed a technique whereby the point-to-point distance in the vorticity integration is always equal to the smallest dimension of the local grid cell. The new point location is found by advancing this distance in the direction of the local vorticity vector. This procedure ensures that successive points will not be more than one grid cell apart, so that if the original calculation is well resolved, then the vorticity-line calculation will also be sufficiently resolved. The procedure also reduces the chance of wasting many calculations inside a single grid cell.

Corrector Step

Our implementation of the pressure-minimum correction scheme uses the method of steepest descent to find the local pressure minimum in the plane perpendicular to the vorticity vector. The smallest grid-cell dimension is used as a local length-scale to march along the gradient direction.

The corrector phase can be iterated in order to converge to the skeleton, but such convergence is not guaranteed. We therefore limit the angle that the vorticity can change during a repeated iteration of the corrector phase, requiring that the cosine of the angle between the predicted and corrected vorticity be at least 0.9. In case it is not, we simply quit the corrector phase. We could choose a smaller step-size and re-try, but we have not found this to be necessary.

4 Finding the Cross-section

Having produced skeletons that follow vortex cores, we must next determine the shapes of the vortices through which they pass. A vortex generally assumes an elongated shape which is well-approximated locally by a cylinder. Our goal is to determine the cross-sections of the vortex tubes in planes perpendicular to the core. Since it is unclear how to precisely define which points lie in a vortex (see section 2), it is also unclear how to determine the exact shape of a vortex tube's cross-section. Determining an appropriate measure of the vortex cross-section has been one of the more difficult practical aspects of this work.

A point on the vortex skeleton serves as a convenient center for a polar coordinate system in the plane perpendicular to the skeleton line. We have chosen therefore to characterize the cross-section by a radius function. Note that this scheme correctly captures star-shaped cross-sections. Cross-sections with more elaborate shapes are truncated to star shapes, with discontinuities in the radius function (fig. 12). In practice this choice does not seem to be very restrictive, as section 4.2 indicates.

In examining the cross-section plane there are two important questions to address. First, what determines whether a point in the plane belongs to the vortex tube? Second, how should

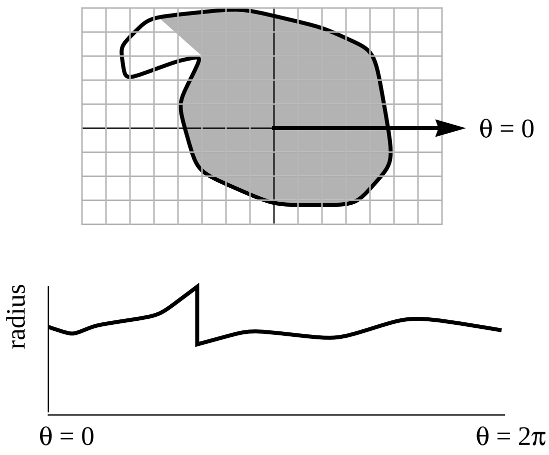


Figure 12.
Representation of the cross-section in polar coordinates. The star-shaped interior (gray) of a non-convex curve (black) is represented by a radius function (bottom). In general, the vortex cross-sections have continuous, periodic cross-sections suitable for Fourier representation.

the shape of the tube's cross-section be represented? This section summarizes the strategies that we found to be successful.

4.1 Criteria for Determining Membership

As the survey demonstrated, there are many heuristics for deciding whether a point is a member of a vortical structure. Most techniques appeal to some scalar quantity derived from flow quantities: a certain threshold of that quantity determines membership in a vortex. Since the predictor-corrector method relies on pressure and vorticity, we wish to re-use these quantities for determining membership in a vortex. For massive datasets there is a significant penalty for storing or calculating additional scalar quantities.

For isolated vortices, a threshold of pressure provides an effective criterion to determine whether a point belongs to a vortex. But when two or more vortices interact, their low-pressure regions merge and distort the radius estimate of any single vortex. This difficulty is resolved if the angle between the vorticity vector on the skeleton line and the vorticity vector at any radial position is restricted. Any angle greater than 90 degrees indicates that the fluid at the radial position is rotating in the direction opposite to that in the core. We have found that the 90-degree restriction works well in combination with a low-pressure criterion for the vortex edge.

For the actual computation of the radial distance, the pressure and the vorticity are sampled along radial lines, emanating from the skeleton, lying in the perpendicular plane. We step along each radial line until a point is reached that violates the vorticity or the pressure-threshold criterion.

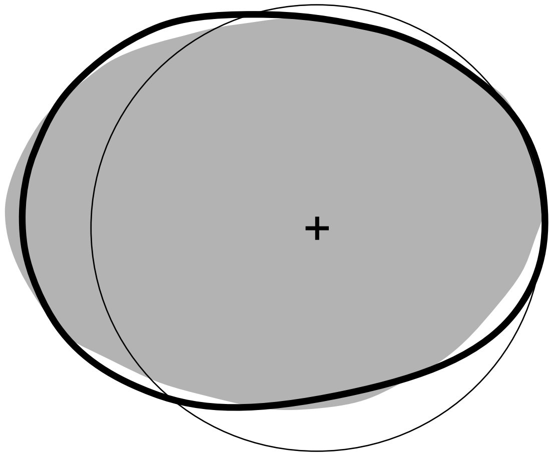


Figure 13.
Comparison of different ways to represent the cross-section of a vortex tube. The shaded region is the finely-sampled radius function. The thin line is an approximating circle. The thick line is a 5-term Fourier representation.

4.2 Representation of the Cross-section

If the radius of the cross-section were sampled at 1-degree increments, then 360 radial distances (and a reference vector to define the 0-degree direction) would be associated with each skeleton point. That is a great deal of data to save for each point of a time-varying set of vortex skeletons. We have found that an average radius is sufficient to describe the cross-section of an isolated vortex tube.

When vortices begin to interact, the cross-section becomes non-circular and so the average radius does not provide a good description of its shape. A truncated Fourier representation of the radial distance provides a convenient compromise between the average radius and a full set of finely-sampled radial locations. The series is easy to compute, easy to interpret, and allows a large range of cross-sectional shapes. In our work, we keep the constant term, the first and second sine and cosine coefficients, the vorticity ω, and a unit reference vector **x** that defines the 0-degree direction in the cross-sectional plane. The cross-sectional radius is thus parametrized by

$$r(t) = a_0 + a_1 \cos t + b_1 \sin t + a_2 \cos 2t + b_2 \sin 2t$$

where $0 \leq t \leq 2\pi$.

In general, the magnitudes of the last two coefficients (a_2 and b_2) are comparatively small, indicating that the neglected terms are not significant. That observation also validates our assumption that the cross-section is well-represented by a continuous polar function.

Fig. 13 illustrates a single cross section of a vortex extracted from direct numerical simulation data. The shaded region is the interior of the vortex tube, sampled at 1-degree intervals. The thin line is a circle, centered at the skeleton, showing the average radius of the vortex tube. The thick line is the truncated Fourier series representation of the vortex cross-section, providing a better approximation than the circle.

In our time-varying data a single vortex develops into 44 vortices over the course of 231 time steps. In total there are 3,584 individual cores and 365,839 positive-area cross-sections. We calculated the relative energy represented by the last Fourier coefficients according to the fraction

$$E_{rel} = \frac{a_2^{\ 2} + b_2^{\ 2}}{a_0^{\ 2} + a_1^{\ 2} + b_1^{\ 2} + a_2^{\ 2} + b_2^{\ 2}}$$

In 87% of the cross-sections, the relative energy E_{rel} due to the last two coefficients accounts for less than one-tenth of the total energy.

5 Data Reduction and Reconstruction

Time-varying volumetric datasets generally consume vast amounts of storage. This section is concerned with the problem of reducing the data size to permit an interactive examination of a time-varying flow. The typical non-interactive avenue for producing an animation of 3D volumetric structures is to extract isosurfaces at each time step, generate an image frame, and record each frame to videotape or to disk. The individual datasets may take a long time to retrieve from remote mass-storage devices and the isosurfaces may take a long time to extract, but this pre-processing step is incurred only once to produce an animation. Replaying the animation on a workstation presents other problems. A two-minute animation, at 30 frames per second, requires 3600 frames. A full color frame, at a resolution of only 640×480 pixels, requires about a megabyte. The total of 3.6 gigabytes of storage exceeds the range of current workstation memories. The animation can be compressed using MPEG, but decoding and displaying it at 30 frames per second is a challenge. Even if the animation could be replayed conveniently, the general strategy of extracting isosurfaces from massive remotely-stored volumetric datasets does not promise interactive exploration of the time-varying flow in the foreseeable future.

There are alternative techniques for compressing the volumetric data and even for rendering images from the compressed format. Ning and Hesselink [21] report compression factors of about 5-fold by using vector quantization. The technique improves the performance of their volume renderer to about one minute per frame. Shen and Johnson [22] use frame-to-frame differencing, with a fixed viewpoint, to achieve compression factors up to about 700-fold at a rendering rate of better than one second per frame. We desire a scheme that offers both substantial data reduction to permit local storage and fast rendering to permit real-time interaction.

There are techniques that reduce the number of polygons in a surface representation of a solid, as opposed to rendering it volumetrically. By visualizing only the polygonalized boundaries of vortex tubes, one benefits from the fast rendering speed of the graphics hardware, as compared with a slower volume-rendering of the vortex interiors. Hoppe [23] reduced the polygon count of unstructured meshes by factors of 10 to 16. Turk [24] reduced the polygon count of unstructured meshes by factors of 10 to 18. Schroeder [25] used multiple passes to reduce the polygon count by factors of up to 10. These techniques are designed to apply to somewhat arbitrary surface shapes. In the case of vortex tubes we exploit their elongated cylindrical shape to achieve even more aggressive data-reduction using the Fourier series. In addition we are able to specify, at run-time, the polygonal resolution of the reconstructed vortex tubes. The details of reduction and reconstruction are described below.

We performed a flow simulation using Cray computers over the course of two calendar years, using about 2000 Cray2 hours of processing time. The numerical grid grows with the size of the evolving flow structures from an initial grid size of 301×121×41 (in the streamwise, wall-normal, and spanwise directions) to a final grid size of 461×161×275. Each grid point holds 1 data-word for pressure and 3 data-words for vorticity. A Cray word is actually 8 bytes, but 4 bytes per word would be adequate. The storage needs for each time step range from 24 megabytes to 326 megabytes, assuming a 4-byte word. The entire set of 3D grids requires at least 45 gigabytes of storage. By using vortex skeletons with Fourier-series cross-sections we are able to reduce the data significantly and then reconstruct the vortex tubes locally on a workstation.

5.1 Data Reduction

In our DNS data, a typical vortex skeleton is a polyline composed of 30 to 200 samples. The time steps in the numerical simulation are non-uniform: the non-integer time increment is determined by bounding the amount of integration error it introduces. The vortex tubes pictured in fig. 7 are calculated at time step 152.8 and contain 1397 skeleton points. Each sample in a vortex skeleton requires 60 bytes of data to represent its position, tangent, reference vector, cross-section coefficients, and velocity magnitude. Thus a reduction from 227 MB to 84 KB is achieved at this particular time step, a 3000-fold improvement over the volumetric data size.

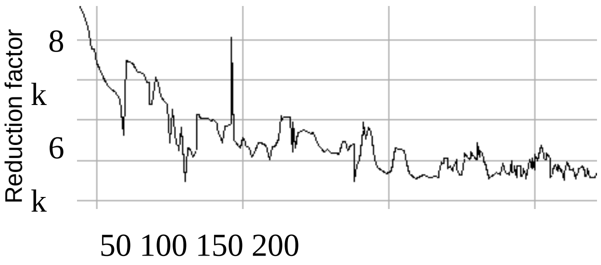


Figure 14.
Reduction factors achieved using vortex skeletons. Horizontal axis indicates time step in the numerical simulation of an unsteady flow. Vertical axis indicates ratio of the size of original 3D grid to the size of the skeletal representation of vortices.

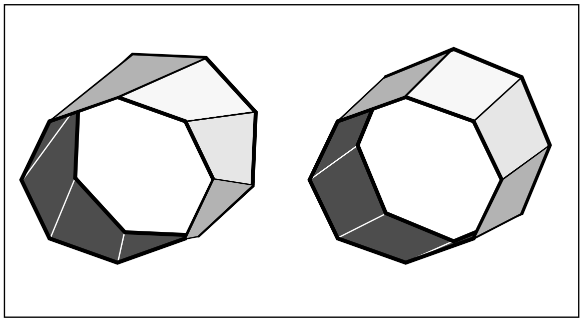


Figure 15.
A quadrilateral mesh connects consecutive cross-sections (each with 8 samples) in a tube. On the left, 20° of twist between cross-sections causes the mesh to skew. On the right, the cross-section at the back has samples which are aligned with those at the front.

Fig. 14 shows the reduction factors for the vortices over a range of time steps. At the end of the simulation the flow becomes fully turbulent and the 3D grid contains many interacting vortices over a large sub-volume of the computational domain. Even so, the technique continues to reduce the dataset by factors of one to three thousand. The vortex data from the entire simulation can be reduced from the 45 GB volumetric grid to a 24 MB skeletal representation. This is an average reduction factor of about 1800.

5.2 Faithful Reconstruction

The significant data-reduction that vortex skeletons provide does not come without cost. There is still the matter of reconstructing polygonal tubes from the skeletons. If the tubes have circular cross-sections, they are generalized cylinders. Bloomenthal gives a clear exposition of how to reconstruct a generalized cylinder from a curve through its center [26]. The coordinate system of the cross-section usually twists from one skeleton point to the next. The key issue is how to keep the rate of rotation about the skeleton’s tangent vector small. Excessive twist is visible in the polygons that comprise the tube: they become long and thin and their interiors approach the center of the tube (fig. 15). Our tubes are not cylinders: the additional terms in the Fourier series produce non-circular cross-sections. But a coordinate frame that twists along the skeleton will produce the same visible artifacts in a polygonal mesh.

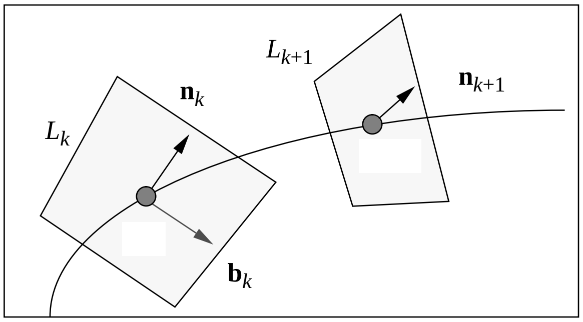


Figure 16.
Basis vector $\mathbf{n}_k$ at a point $\mathbf{p}_k$ on a curve is projected onto the cross-section plane L_{k+1} to produce a new basis vector $\mathbf{n}_{k+1}$.

In order to reduce twisting of the coordinates, we project the coordinate bases from one cross-section onto the next cross-section (fig. 16). Let $\mathbf{p}_k$ be a point in the vortex skeleton with normal $\mathbf{n}_k$ and binormal $\mathbf{b}_k$. The tube's cross-section lies in the plane L_k defined by coordinate axes $\mathbf{n}_k$ and $\mathbf{b}_k$. The following point $\mathbf{p}_{k+1}$ has cross-section plane L_{k+1}. We project $\mathbf{n}_k$ onto plane L_{k+1} to produce a new normal vector $\mathbf{n}_{k+1}$. This produces a new coordinate system that has not twisted compared to its predecessor. The initial normal $\mathbf{n}_0$ and binormal $\mathbf{b}_0$ can be chosen in a variety of ways. We use $(1, 1, 1) \times \omega$ as an initial choice of the normal vector $\mathbf{n}_0$, where ω is the vorticity and the coordinates tuple corresponds to the (stream, wall-normal, spanwise) directions. In the rare case that (1, 1, 1) and ω are aligned, we use (1, 0, 0) as a second choice to produce the normal vector. The new normal vector might be different from the reference vector (which indicates the 0-degree direction) for the Fourier representation of the cross-section. To reconstruct the cross-section, we phase-shift the angle in the Fourier series by the angular difference between the normal and the reference vector.

In general, 20 to 80 samples suffice to reconstruct a cross-section of acceptable image-quality. We keep the number of cross-sectional samples constant along a reconstructed vortex tube so that the tube can be represented as a quadrilateral mesh. Many graphics libraries have drawing routines that are optimized for quadrilateral meshes.

Our original 3D grids, over 230 time steps, require at least 45 GB of storage. But in the reconstructed vortex tubes there are only 404,428 skeleton points. A point on the polygonal mesh requires 27 bytes (for position, normal, and color). If each cross-section has 20 samples, the entire polygonalized, time-varying dataset requires about 220 MB of storage, which is easily within the reach of large-memory workstations.

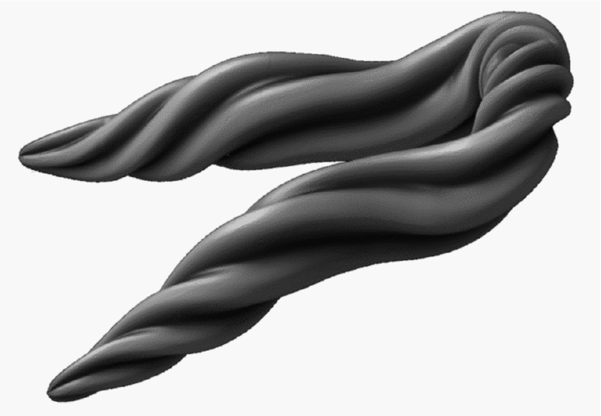


Figure 17.
Enhanced reconstruction of a hairpin vortex tube. The grooves follow integral curves of velocity, constrained to follow the surface of the tube.

5.3 Enhanced Reconstruction

Sometimes there is good reason for a “reconstruction” that is not faithful to the original shape of the vortex tube. The faithful reconstruction in fig. 1e does not convey the spiraling motion along the surface of the vortex tube. We experimented with different methods of visualizing the velocities on the tube itself. One helpful technique is to create a texture on the surface, drawing curves to indicate the helical flow. This visualization is enhanced dramatically when the curves are displaced inward to produce grooves.

Fig. 17 demonstrates this technique on a single hairpin vortex. The grooves follow integral curves of the surface-constrained velocity vectors. That is, a curve is developed on the surface of the tube by projecting the velocity vectors onto the tube surface and integrating. The three curves in the figure begin from initial trajectories that are shifted in phase by increments of 120 degrees. In an informal survey of a dozen colleagues, we found that none could estimate the amount of helical motion in a faithful reconstruction (as in fig. 1e) of a vortex tube; after all, there are no visual indications of the vortical motion. On the other hand, the same subjects instantly identified the direction and amount of rotation in the enhanced image of fig. 17. The model in the figure uses over 250,000 polygons to represent the vortex. This polygon count is prohibitively large for contemporary graphic systems to display in real time. For a static visualization, however, a large polygon count is reasonable in the trade-off between image quality and rendering speed. As graphics architectures begin to deliver 30 million polygons per second [27], we expect that such enhanced reconstructions of flow features will become more common.

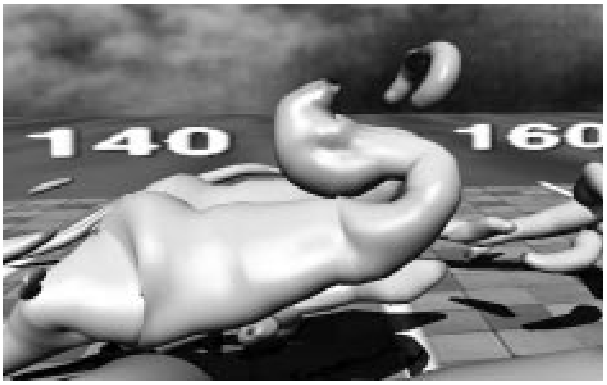


Figure 18.
S-shaped vortex head at time 184.6 displayed in the Tracktur system. The white stripes on the flat plate mark units in the computational domain.

5.4 Interactive Time-Varying Visualization

The predictor-corrector scheme was developed in order to visualize vortical structures in a time-varying turbulent flow. The scheme has the added benefit that it represents the vortex tubes very efficiently. We wish to visualize and explore the flow dynamically; to that end we have developed an interactive application called “Tracktur” [28] which allows investigation of the vortices as they evolve in a flow. There are other systems that have been developed for similar purposes [29] [30]. Tracktur differs from them by exploiting the data-reduction that the predictor-corrector scheme provides in order to display vortices in an unsteady flow. In addition, Tracktur provides 3D head-tracking, stereo display, and 3D hand-tracking to let a viewer navigate among the vortices and probe quantitative values within the flow. The system sustains about 15 updates per second on a full-screen display of about 8000 polygons using the Silicon Graphics Onyx with Reality Engine 2 graphics.

Our ultimate goal is to better understand how a turbulent spot develops. Since this is a complex and dynamic process, we expected that a time-varying visualization tool like Tracktur would provide significant support. Other researchers report modest success in applying visualization systems to study scientific problems of interest to them [31] [32]. By using Tracktur we have discovered a backward-tilting S-shaped vortex head (fig. 18) that had been seen experimentally in a similar flow (fig. 19) [1], but had not been identified before in the flow data we were investigating.

6 Future Work

There are two important issues in data-reduction and reconstruction still to be addressed. First, we would like to minimize the number of samples along a vortex skeleton. Where the vortex skeleton has high curvature or where the cross-section changes shape quickly, many samples are required to produce an accurate reconstruction. But most vortex tubes have long, straight portions with nearly-circular cross-sections of nearly-constant radius. This characteristic should permit us to represent the vortex tube with fewer samples along its skeleton.

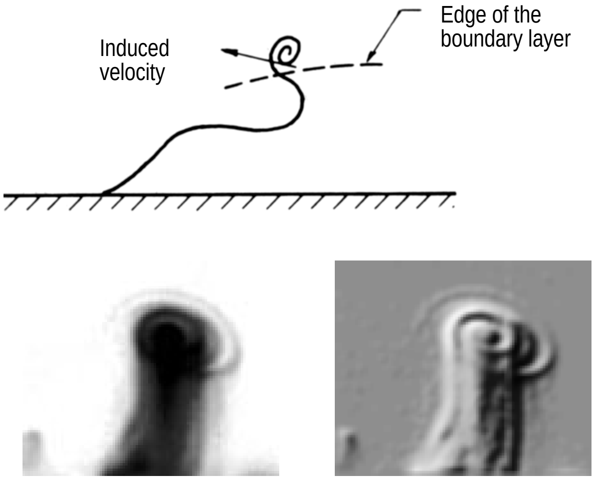


Figure 19.
S-shaped vortex head in an experimental shear flow over a flat plate. Top: schematic diagram of the profile. From Acarlar and Smith, *JFM* v. 175, p. 71. © Cambridge University Press 1987. Reprinted with the permission of Cambridge University Press. Bottom left: dye injected into the flow develops into an upright head. Image courtesy of C. R. Smith. Bottom right: intensity gradients of the image at left produce a bas-relief image.

The second issue concerns interpolation. In reviewing the development of a vortical flow, a scientist may be especially interested in narrowing the interval of animation to only a few of the original time steps. It would be helpful to generate in-between frames from the given data. One could interpolate the original 3D grids to extract interpolated vortex skeletons, but that would require a great deal of data communication and computation. Interpolating between the skeletal representations, on the other hand, could be done in memory. Unfortunately, it is difficult to interpolate vortex tubes as they appear, branch, merge, and disappear over time. Other researchers have addressed the issue of matching corresponding isosurfaces in unsteady flows [33]. Matching and interpolating the skeletal representation remains as future work. Concerning the enhanced vortex reconstruction, it may be possible to animate the spiral grooves by advecting the displacement coordinates according to the flow velocities. Max, Crawfis, and Williams have used a similar technique to visualize wind velocities [34].

7 Conclusions

The innovative use of a two-step predictor-corrector algorithm has been introduced to identify vortices in flow-field data. Unlike other approaches, our method is able to self-correct toward the vortex core even in a turbulent shear layer. The principle of using the vorticity vector field to predict the location of the next point and the gradient of the scalar pressure field to correct this position distinguishes this method from others. The theoretical justification for the technique is that vortices are generally characterized by large magnitudes of vorticity and low pressures in their core. The presence of these two characteristics in a cross-section defines the shape of the vortex interior.

This paper discusses a number of novel approaches that we have developed to deal with matters such as eliminating redundant vortices, eliminating feeders, and representing the cross-section of a vortex tube. Sample extractions of vortices from various flow fields illustrate the different aspects of the technique.

The vortex skeletons are an economical way to represent vortical structures within a flow, offering data-reduction on the order of more than a thousand-fold even in a complex flow. This presents an opportunity to store hundreds of frames of vortex geometry in workstation memory. As a proof of concept, we implemented a system that lets a user interactively explore an evolving turbulent spot. Where interactivity is not important, a vortex tube can be enhanced during reconstruction by modelling grooves in the surface in order to help display the dynamics of vortical flow in a static image.

Acknowledgments

The images in fig. 1a-d were rendered on a Silicon Graphics Indigo workstation using the FAST visualization system. The images in figs. 2, 4, and 7-11 were rendered on a Silicon Graphics Indigo^2 using the Explorer visualization system. Figs. 1e and 18 were produced using Tracktur. The image in fig. 17 was rendered on an Intel Paragon using PGL (Parallel Graphics Library) [35].

We thank Gordon Erlebacher for his helpful insights regarding vortex identification schemes. We thank Greg Turk and the reviewers for their suggested improvements to this paper.

References

[1] M. S. Acarlar and C. R. Smith, “A study of hairpin vortices in a laminar boundary layer. Part 2. Hairpin vortices generated by fluid injection,” Journal of Fluid Mechanics, Vol. 175 (1987), pp. 43-83.

[2] A. Leonard, “Vortex Methods for Flow Simulation,” *Journal of Computational Physics*, **37** (1980), pp. 289-335.

[3] D. C. Banks and B. A. Singer, “Vortex Tubes in Turbulent Flows: Identification, Representation, Reconstruction,” *Proceedings of Visualization ’94*, IEEE Press, pp. 132-139.

[4] S. K. Robinson, “Coherent motions in the turbulent boundary layer,” *Annual Review of Fluid Mech*anics, **23** (1991), p. 601.

[5] S. K. Robinson, “A review of vortex structures and associated coherent motions in turbulent boundary layers,” in *Proceedings of Second IUTAM Symposium on Structure of Turbulence and Drag Reduction*, Federal Institute of Technology, Zurich, Switzerland (July 1989), pp. 25-28.

[6] S. K. Robinson, S. J. Kline, and P. R. Spalart, “A review of quasi-coherent structures in a numerically simulated boundary layer,” NASA TM-102191 (1989).

[7] M. S. Chong, A. E. Perry, and B. J. Cantwell, “A general classification of three-dimensional flow fields,” *Physics of Fluids A* **2**:5 (1990), p. 765.

[8] J. Soria, and B. J. Cantwell, “Identification and classification of topological structures in free shear flows,” *Proceedings of IUTAM Eddy Structure Identification in Free Turbulent Shear Flows* (1992).

[9] P. Moin and J. Kim, “The structure of the vorticity field in turbulent channel flow. Part 1. Analysis of instantaneous fields and statistical correlations,” *Journal of Fluid Mechanics,* **155**, (1985). p. 441.

[10] J. Kim, and P. Moin, “The structure of the vorticity field in turbulent channel flow. Part 2. Study of ensemble-averaged fields,” *Journal of Fluid Mechanics*, **162** (1986), pp. 339-363.

[11] J. Jiminez, A. A. Wray, P. G. Saffman, and R. S. Rogallo, “Intense vorticity in isotropic turbulence,” *Journal of Fluid Mechanics*, **255** (1993), pp. 65-90.

[12] L. A. Yates, and G. T. Chapman, “Streamlines, Vorticity Lines, and Vortices,” AIAA Paper 91-0731 (1991).

[13] A. Globus, C. Levit, and T. Lasinski, “A Tool for Visualizing the Topology of Three-Dimensional Vector Fields,” *Proceedings of Visualization ’91*, IEEE Press, pp.33-40.

[14] G. V. Bancroft, F. J. Merritt, T. C. Plessel, P. G. Kelaita, R. K. McCabe, and A. Globus, “FAST: A Multi-processed Environment for Visualization of Computational Fluid Dynamics,” *Proceedings of Visualization ’90*, IEEE Press, pp. 14-27.

[15] P. S. Bernard, J. M. Thomas, and R. A. Handler, “Vortex dynamics and the production of Reynolds stress,” *Journal of Fluid Mechanics*. **253** (1993), p. 385.

[16] J. Villasenor and A. Vincent, “An algorithm for space recognition and time tracking of vorticity tubes in turbulence,” *CVGIP: Image Understanding* **55**:1 (1992), p. 27.

[17] N. J. Zabusky, O. N. Boratav, R. B. Pelz, M. Gao, D. Silver, and S. P. Cooper, “Emergence of coherent patterns of vortex stretching during reconnection: A scattering paradigm,” *Phys. Rev. Let*. **67**:18 (1991), p. 2469.

[18] B. A. Singer and R. D. Joslin, “Metamorphosis of a Hairpin Vortex into a Young Turbulent Spot,” *Physics of Fluids*, 6 (1994), p. 3724.

[19] J. Jimenez and P. Moin, “The Minimal Flow Unit in Near-wall Turbulence,” *Journal of Fluid Mechanics*, Vol. 225 (1991), pp. 213-240.

[20] B. A. Singer and D. C. Banks, “A Predictor-Corrector Scheme for Vortex Identification,” ICASE Report No. 94-11; NASA CR-194882 (1994).

[21] P. Ning and L. Hesselink, “Fast Volume Rendering of Compressed Data,” *Proceedings of Visualization ’93*, IEEE Press, pp. 11-18.

[22] H.-W. Shen and C. R. Johnson, “Differential Volume Rendering: A Fast Volume Visualization Technique for Flow Animation,” *Proceedings of Visualization ’94*, IEEE Press, pp. 180-187.

[23] H. Hoppe, T. DeRose, T. DuChamp, J. McDonald, and W. Stuetzle, "Mesh Optimization," *SIGGRAPH '93 Conference Proceedings*, ACM Press, pp. 19-26.

[24] G. Turk, "Re-tiling Polygonal Surfaces," *SIGGRAPH '92 Conference Proceedings*, ACM Press, pp. 55-64.

[25] W. J. Schroeder, J. A. Zarge, and W. E. Lorenson, "Decimation of Triangle Meshes," *SIGGRAPH '92 Conference Proceedings*, ACM Press, pp. 65-78.

[26] J. Bloomenthal, "Calculation of Reference Frames Along a Space Curve," *Graphics Gems I*, (Andrew Glassner, ed.), Academic Press, INC. (1990).

[27] S. Molnar, J. Eyles, and J. Poulton, "PixelFlow: High-Speed Rendering Using Image Composition," *SIGGRAPH '92 Conference Proceedings*, ACM Press, pp. 231-240.

[28] D. C. Banks and M. Kelley, "Tracking a Turbulent Spot in an Immersive Environment," *Proceedings of the 1995 Symposium on Interactive 3D Graphics*, ACM Press, pp. 171-172.

[29] D. A. Lane, "Visualization of Time-Dependent Flow Fields," *Proceedings of Visualization '93*, IEEE Press, pp. 32-38.

[30] A. J. S. Hin and F. H. Post, "Visualization of Turbulent Flow with Particles," *Proceedings of Visualization '93*, IEEE Press, pp. 46-51.

[31] S. E. Parker and R. Samtaney, "Case Study: Tokamak Plasma Turbulence Visualization," Proceedings of Visualization '94, IEEE Press, pp. 337-340.

[32] N. Zabusky and D. Silver, "Case Study: Visualizing Classical Problems in CFD," *Proceedings of Visualization '92*, IEEE Press, pp. 436-440.

[33] R. Samtaney, D. Silver, N. Zabusky, and J. Cao, "Visualizing Features and Tracking Their Evolution," *IEEE Computer,* **27**: 7 (July 1994). pp. 20-27.

[34] N. Max, R. Crawfis, D. Williams, "Visualizing Wind Velocities by Advecting Cloud Textures," *Proceedings of Visualization '92*, IEEE Press, pp. 179-184.

[35] T. W. Crockett and T. Orloff, "Parallel Polygon Rendering for Message-Passing Architectures," *IEEE Parallel and Distributed Technology,* Vol. 2, No. 2 (1994), pp. 17-28.