Physically-Based Feature Tracking for CFD Data

John Clyne, Pablo Mininni, and Alan Norton

Abstract—Numerical simulations of turbulent fluid flow in areas ranging from solar physics to aircraft design are dominated by the presence of repeating patterns known as coherent structures. These persistent features are not yet well understood, but are believed to play an important role in the dynamics of turbulent fluid motion, and are the subject of study across numerous scientific and engineering disciplines. To facilitate their investigation a variety of techniques have been devised to track the paths of these structures as they evolve through time. Heretofore, all such feature tracking methods have largely ignored the physics governing the motion of these objects at the expense of error prone and often computationally expensive solutions. In this paper, we present a feature path prediction method that is based on the physics of the underlying solutions to the equations of fluid motion. To the knowledge of the authors the accuracy of these predictions is superior to methods reported elsewhere. Moreover, the precision of these forecasts for many applications is sufficiently high to enable the use of only the most rudimentary and inexpensive forms of correspondence matching. We also provide insight on the relationship between the internal time stepping used in a CFD simulation, and the evolution of coherent structures, that we believe is of benefit to any feature tracking method applicable to CFD. Finally, our method is easy to implement, and computationally inexpensive to execute, making it well suited for very high-resolution simulations.

Index Terms—Feature tracking, flow visualization, time-varying data, CFD

1 Introduction

Large-scale numerical simulations of turbulent flows, enabled by rapid advances in supercomputing technology, are providing major insights into fields ranging from the earth and space sciences to aerodynamics. A remarkable finding is that most turbulent flows exhibit a range of coherent structures embedded in the more chaotic motions, with the coherent structures playing a profound role in the transport of important quantities such as energy, momentum, and magnetic fields. The advent of petascale supercomputing promises simulations of turbulence in real-world applications at unprecedented resolutions, yet at the same time accentuates the enormous data analysis challenges faced in understanding numerical solutions. The largest simulations may generate petabytes of data, all but precluding movement of a simulation output in its entirety, and severely limiting interactive exploration.

Reducing data set sizes to more manageable scales is an essential component of many scientific workflows. Because of their role in magnetohydro and fluid dynamics, individual coherent structures such as vortex tubes, filaments, and sheets are the subject of intense study. Isolating and extracting these complex, dynamic features at a single instant in time by thresholding such quantities as vorticity magnitude, helicity, current, or velocity, is an effective technique for facilitating their investigation and is readily supported by commonplace visualization methods such as direct volume rendering or isosurfacing. The need to understand the evolution of these structures over time, and not just at a single instant, has led to the design and application of numerous feature tracking methods. By automating the tracking of these structures significant reductions in the volume of data that must be processed in an interactive setting may be possible. The researcher may identify a structure of interest at one time step, and then apply an automated feature tracking algorithm to search through the entire spatiotemporal domain, subsetting and returning only the minimally enclosing subvolumes containing the feature at other time steps, achieving a vast reduction in data. Alternatively, the feature tracking algorithm may be integrated with an analysis capability that captures salient statistics that are isolated to an evolving feature.

Most of the reported feature tracking methods [1], [2], [3], [4], [5], [6], [7], [8], [9] applicable to general coherent structures are based on *correspondence matching*: features extracted from successive time steps are compared against a target feature from the current time step using a variety of metrics in an effort to uniquely correspond a structure from frame to frame. The matching criteria may include a combination of attributes such as size, volume, orientation, or a variety of statistics. Because the coherent structures contained in Computational Fluid Dynamics (CFD) simulation outputs evolve and change over time the comparisons are inexact and the matching approach is prone to error; incorrect features may be corresponded, or no match may be found even when one exists. While frequent time sampling reduces the evolutionary changes in features from frame to frame—making structures easier to track—the trend with

- *J. Clyne and A. Norton are with the National Center for Atmospheric Research, c/o NCAR, PO Box 3000, Boulder, CO 80305. E-mail: {clyne, alan}@ucar.edu.*
- *P. Mininni is with the Departamento de Fsica, Facultad de Ciencias Exactas y Naturales (FCEN), Universidad de Buenos Aires (UBA), Ciudad Universitaria, Pabellon 1, 1428 Buenos Aires, Argentina. E-mail: mininni@df.uba.ar.*

Manuscript received 29 Aug. 2012; revised 25 Jan. 2012; accepted 7 Aug. 2012; published online 15 Aug. 2012.
Recommended for acceptance by G. Scheuermann.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2011-08-0204.
Digital Object Identifier no. 10.1109/TVCG.2012.171.

longer, and larger scale simulations is toward less frequent output of time steps to minimize storage needs, exacerbating the difficulties with correspondence matching.

In this paper, we present a novel method for predicting the motion of coherent structures widely found in, and of significant importance to, numerous CFD applications. We take advantage of the fact that the paths of these features are not random, but are in fact deterministic, prescribed by the equations of fluid motion underlying the simulations. For a feature identified at a single instant of time we are able to precisely predict the position of that feature in the near future (or past) using only the sampled solution to the flow field. We show that our prediction method is more accurate than the approximate solutions proposed by other authors [5], [9], and we demonstrate its successful application in the context of a feature tracking system that relies solely on one of the most simple forms of correspondence matching: volume differences.

The second major contribution of this work is to provide guidance on the maximum temporal spacing for a CFD solution necessary for robust results using *any* feature tracking approach. When this guidance is followed the complexity of correspondence matching, which dominates many feature tracking implementations, can be significantly reduced, and the accuracy of the tracking improved. Finally, our prediction method is both simple to implement and computationally efficient.

2 Related Work

The vast majority of feature tracking algorithms suitable for following coherent structures in turbulent flows are based on correspondence matching [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. These methods may be divided into those that are cognizant of, and benefiting from, temporal coherence assumed in the data set [1], [2], [3], [5], [6], [9], and those methods that make no such assumptions [8], [7]. The former can be further divided into methods that not only benefit but also demand frame to frame coherence, in particular requiring that the spatial boundaries of corresponding features overlap in time [3], [11], [6], [12], [9], [13] and those methods that merely rely on temporal coherence as an optimization to reduce the search domain [1], [2], [5]. Our method does not require structures to overlap in time.

Silver et al. [1] and Samtaney et al. [2] report on some of the earliest work exploiting temporal coherence to track coherent structures. Their primary matching criterion is the distance between object centroids: features whose centroid distance are within a tolerance of the tracked feature's centroid are considered candidates for a match. The candidate pool is further narrowed using second order moments of inertia to characterize the shape and orientation of the feature. Reliability problems, particularly when dealing with small or fast moving features, led Silver and Wang to later develop a volumetric-based correspondence matching algorithm, and to further improve reliability, required candidate features to have spatial overlap with the tracked feature [3]. When multiple candidates exist the best match is determined by comparing normalized volume differences; the candidate with the minimum relative volume difference between itself and tracked feature is considered the best match. Weigle and Banks [11], as well as Ji et al. [6] take a somewhat different approach, implicitly tracking structures in a scalar field by constructing higher dimensional isosurfaces across the entire spatiotemporal domain. Conceptually, this approach is equivalent to finding overlapping features.

The preceding methods take advantage of temporal coherence by assuming that the magnitude of displacement of a feature between successive time steps is constrained to some tolerance. However, no guidance is given on how this tolerance should be determined. Moreover, no assumptions are made regarding the direction of displacement. Reinders et al. were among the first to exploit the nonrandom trajectory of coherent structures both in terms of distance and direction [5]. The method uses a prediction step based on linear extrapolation to estimate the new location of a tracked feature, followed by a correction step to verify the validity of the prediction. The verification process is essentially correspondence matching using techniques similar to those previously described. However, it is worth noting that the authors refine the matching process by including a user-defined weighting for the various comparison metrics. More importance can be given to feature mass than feature volume, for example. Muelder and Ma also explore prediction and correction methods, comparing linear and quadratic extrapolation [9]. Valid results are dependent on the predicted feature overlapping the correct feature and no others. Both of these methods require a boot strapping process to initiate the prediction: features must be correlated through some other means for the first two frames for linear extrapolation (three frames for quadratic). The work of Reinders et al. [5] and Muelder and Ma [9] is most similar to our own. Finally, we note that the methods benefiting from temporal coherence discussed earlier may be considered to be based on a form of *constant* extrapolation: the features are assumed to remain stationary.

Requiring the temporal sampling frequency to be high enough to guarantee that features of interest possess spatial overlap in successive time steps may limit the viability of the above algorithms for many real-world data sets where storage availability often dictates the time spacing between saved frames. Furthermore, the question of how to determine the maximum displacement of a feature between successive time steps, or, turning the problem around, determining the appropriate time sampling to bound the maximum displacement, has not been addressed.

Ji and Shen relax the time sampling restrictions by introducing a *global* tracking algorithm: no limitations are placed on the motion of a feature between time steps [7]. The algorithm corresponds structures by representing their attributes as spatial distributions and uses the Earth Mover's Distance [14], [15] to assess the cost of lifting one distribution to another. Hence, occupancy information that includes shape, orientation, position, and scale contribute to the cost assessment. Though in theory a structure may move anywhere in the domain between time steps, the cost algorithm is weighted heavily in favor of features that are in close proximity to the tracked feature. Caban et al. present another global algorithm [8]. They treat the rectilinear region enclosing the feature as a texture and apply image

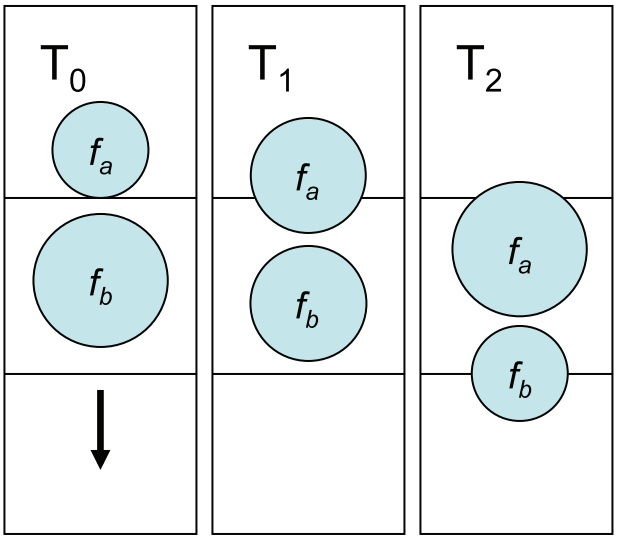


Fig. 1. This cartoon illustrates problems with correspondence matching when the data set is insufficiently sampled in the temporal domain. Two features, f_a and f_b, with identical attributes except for size are easily corresponded from time step T_0 to T_1, provided the tracking method is biased in favor of features that are nearby. However, if time step T_1 is not available, feature tracking algorithms that do not correctly consider the motion of the structures (velocity and acceleration) are unlikely to correctly correspond feature f_b from T_0 to T_2.

processing techniques to characterize and subsequently match the feature containing texture. A combination of 26 first-order, second-order, and run-length statistics are used to establish a feature (texture) signature. The statistics automatically weight the attributes by relevance using a minimal-redundancy-maximum-relevance algorithm [16]. Unlike all previous algorithms, no bias toward nearby features appears to be applied at the expense of having to correspond all features in the domain.

The reliability of all of the aforementioned methods is called into question whenever features with similar attributes are present, and the time sampling is inadequate to distinguish features through position alone; a situation that is not unreasonable to assume for many real-world data sets. Consider the somewhat contrived example in Fig. 1. Two downward moving features, f_a^t and f_b^t, are shown at three time steps, T_0, T_1, and T_2, denoted by the superscript t. The features have identical attributes except for size, which is changing independently for both. If feature f_b^0 is corresponded at time T_1 most of the algorithms could be expected to correctly match f_b^1 because of its position, despite the identical attributes, including size, of f_a^1 and f_b^1. However, if we try to correspond f_b^0 at time step T_2 the algorithms will incorrectly match f_a^2 because its centroid as well as its size are closer to that of f_b^0. Hence, robust and accurate feature tracking in applications where many similar structures may exist requires knowledge of feature motion. Fortunately, as we shall soon see, in many CFD domains such insight is readily available from the solution data itself.

Finally, we mention Feature Flow Fields (FFF) introduced by Theisel and Siedel [12] and recently refined to improve numerical stability by Weinkauf et al. [13]. FFF differs from most of the previously discussed approaches in that no explicit correspondence matching is preformed. The basic idea of FFF is to derive a field $\mathbf{f}(\mathbf{x}, t)$, a function of both space and time, such that all points on streamlines integrated in $\mathbf{f}$ have a constant value (both direction and magnitude) for some vector field $\mathbf{v}(\mathbf{x}, t)$, typically velocity. Thus, a streamline integrated starting from a critical point, for example, of $\mathbf{v}$ at time t_0 should pass through the same critical point for all time t, providing a way to track constant-valued, zero-dimensional objects through time. Though the authors report that FFF may be generalized to higher dimensional features through the use of correspondingly higher dimensional integration primitives (e.g., stream surfaces, stream volumes, etc.), as far as we are aware the method has only been applied to lower dimensional objects (points and lines). Though not discussed by the authors the robustness of FFF would appear to be highly sensitive to the temporal sampling rate: features displaced in adjacent time steps by more than a few grid points may not be connected by streamlines of $\mathbf{f}$. In this way, FFF is similar to the construction of higher dimensional isosurface reported by the authors of [11], [6] requiring spatial overlap of features in time. It is worth mentioning that our method presented here is also built on the concept of streamlines, but we use the fact that in fluid dynamics many theorems exist that prescribe what quantities are preserved along those lines.

3 The Physics of Advection of Coherent Structures

Many equations in fluid dynamics take the form

$$\frac{\partial \boldsymbol{\sigma}}{\partial t} + \mathbf{v} \cdot \nabla \boldsymbol{\sigma} = \mathbf{L}(\mathbf{v}, \boldsymbol{\sigma}), \tag{1}$$

where $\mathbf{v}$ is a velocity field and $\boldsymbol{\sigma}$ can be a scalar (e.g., the temperature or a pollutant) or a vector quantity (e.g., magnetic fields, vorticity, or the velocity itself); $\mathbf{L}$ is a linear or nonlinear differential operator. As an example, the Euler equation for an incompressible velocity field of unit density is

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \nu \nabla^2 \mathbf{v}, \tag{2}$$

with p the pressure, and ν the kinematic viscosity. Similarly, in magnetohydrodynamics, the induction equation for the magnetic field $\mathbf{b}$ in Alfvénic units reads

$$\frac{\partial \mathbf{b}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{b} = \mathbf{b} \cdot \nabla \mathbf{v} + \eta \nabla^2 \mathbf{b}, \tag{3}$$

where η is the magnetic diffusivity. In the context of CFD, the objective is most typically to find (and output) solutions for $\boldsymbol{\sigma}$ and, $\mathbf{v}$ or $\mathbf{b}$ that balance the above equations.

The nonlinear term on the l.h.s. of these equations tells us that the velocity field advects structures in the magnetic field or in the velocity field itself; the property that we will exploit for feature path prediction. This advection can be seen from a Taylor expansion of any quantity $\boldsymbol{\sigma}$ at time $t + dt$. If the quantity is advected by the velocity field $\mathbf{v}$, a region with spatial coordinate $\mathbf{x}$ at time t will be at $\mathbf{x} + \mathbf{v}dt$ at the later time. From the Taylor expansion of

$$\boldsymbol{\sigma}(\mathbf{x}+\mathbf{v}dt, t+dt) \approx \boldsymbol{\sigma}(\mathbf{x},t) + \frac{\partial \boldsymbol{\sigma}(\mathbf{x},t)}{\partial t}dt + \mathbf{v}\cdot\nabla\boldsymbol{\sigma}(\mathbf{x},t)dt, \tag{4}$$

and taking the limit $dt \to 0$, it follows that the terms on l.h.s. of (1)-(3) express the total rate of change of the quantity because of its time evolution (the first term) and because of its motion in space as the quantity is advected by the velocity field (the second term). The terms on the r.h.s. of the equations, on the other hand, are associated with deformation—the first term of (2,3), through surface forces acting in each element of fluid in the case of the Euler equation, and through stretching of magnetic field lines in the case of the induction equation—and to diffusion and dissipation (the second term). Indeed, the structure of the equations is similar to the equations satisfied by the density of a conserved quantity advected by and diffused within the fluid.

We note that the property (4) of the terms on the r.h.s. of (1)-(3) relates advection of quantities in CFD with correspondence methods used for feature tracking. Indeed, in the absence of diffusion and dissipation—when the second term on the r.h.s of (2), (3) is zero—quantities can be identified in the fluid that are conserved as the fluid elements are advected, and which allow one-to-one correspondence between points at different times. One example of this is given by Kelvin's theorem for the velocity field, which follows from (2) (see, e.g., [17])

$$\frac{d}{dt}\oint \mathbf{v}\cdot d\mathbf{l} = \boldsymbol{\nu}\oint \nabla^2\mathbf{v}\cdot d\mathbf{l}, \tag{5}$$

or by the equivalent Alfvén's theorem for the magnetic field [18]. In an inviscid flow ($\boldsymbol{\nu}=0$) the quantity on the l.h.s. of (5), the circulation $\oint \mathbf{v}\cdot d\mathbf{l}$, does not change in time as its time derivative is zero. In other words, any closed material line in the flow (any closed line connecting fluid elements) will be advected preserving its circulation, giving, e.g., a quantity that can be used on physical grounds for correspondence methods. Note the material line may be deformed and the local intensity of the field may vary, but the circulation will be the same. Moreover, these material lines can be tracked by advecting only one point in the line, as the one-to-one correspondence allows reconstruction of the line at a later time. Advecting any point along such a streamline at time t_0 to time t_1, and integrating a new streamline using the advected point as a seed, yields the equivalent streamline at time t_1 as advecting all points along the streamline at time t_0. It is this property that is the basis of our feature tracking approach.

4 Application to Structure Advection

There is no unique definition for structures in fluid dynamics. The term coherent structure is often used to indicate, in opposition to purely random "structureless" flows, the emergence of long living structures, which are advected and deformed by the flow [19]. Such structures often correspond to regions in the flow with particular topological properties of the field lines, resulting in accumulation and wrapping of the field lines around the region. Examples are given by vortex filaments and worms in isotropic and homogeneous turbulence, horseshoe vortices in pipe flows, Taylor columns in rotating flows, and current sheets in conducting fluids.

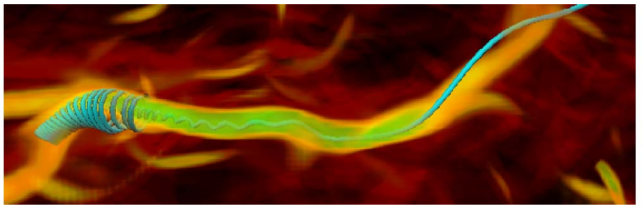


Fig. 2. Coherent structures arising from CFD simulations may be observed both by thresholding scalar quantities, such as vorticity magnitude, and from the topological properties of material lines. Properties of the latter, such as the tight wrapping of velocity streamlines around the vortex core seen above, may be exploited to predict the motion of these structures as they evolve through time.

In vortex filaments, velocity streamlines wrap fast around them following helical trajectories; tornadoes in the atmosphere have similar geometrical properties. An example of this phenomenon is illustrated in Fig. 2. Similarly, current sheets in conducting fluids result from a sharp change in the direction and intensity of magnetic field lines. Thus, if such structures can be identified by streamlines associated with the structures at some initial time, the equations described above give an exact way to track them in flows by their advection by the velocity field, provided there is no dissipation.

When dissipation is present (nonzero viscosity in the hydrodynamic case, or nonzero magnetic diffusivity in the magnetohydrodynamic case) the one-to-one correspondence between points breaks down, as expressed by the nonzero term on the r.h.s. of (5). In that case, the advection of streamlines is broken; the presence of dissipation allows streamlines to diffuse away, or to change their connectivity, and a streamline belonging to one structure may connect with one belonging to another structure. As a result, when dissipation is present there is no unambiguous way to follow a streamline or a structure in time unless all the terms on the equations are computed. In other words, the l.h.s. solutions, computed as a part of a numerical simulation, to (1)-(3) alone are not sufficient for streamline advection; the complete fluid equations of motion must be solved.

It would therefore appear that the advection of material lines using the velocity field resulting from solving the fluid equations of motion would only be appropriate for ideal fluids where dissipation or diffusion are absent. For most CFD applications this is not the case; the effects of diffusion or dissipation are present. We can, however, relax the restriction to ideal fluids by advecting only points for which the effect of diffusion and dissipation is minimal. The motivation is to minimize the r.h.s. of (1-3), or of (5). For Navier-Stokes in the incompressible case, the energy dissipation is given by

$$\Phi = \boldsymbol{\nu}\left(\frac{\partial v_i}{\partial x_j}\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\frac{\partial v_j}{\partial x_i}\right), \tag{6}$$

where v_i are the Cartesian components of the velocity field, and summation over repeated indices is assumed [17]. For compressible fluids the general form of the dissipation

includes extra terms and can be found in [17]. In the case of magnetic fields, the dissipation is proportional to the square of the current density (the curl of the magnetic field).

4.1 Time Stepping

In every CFD simulation, a time step Δt is used for time stepping in computing numerical solutions to equations similar to (1)-(3). The choice of Δt is based on stability conditions of the numerical methods and is in general small compared with all relevant time scales in the problem. Because of storage constraints, only a few *snapshots* of the fields are saved after several time steps. We define the time-sampling period between saved outputs as Δt_s. This latter value will be compared with another time often used as the reference time in simulations of CFD: the turnover time $\tau = L/U$, where L is the typical length of a structure and U is the root mean square (RMS) velocity. For columnar or filamentary structures, L is their diameter, while for sheets L should be interpreted as the structure's thickness. From the physical point of view, the turnover time is proportional to the turnaround time of an eddy—the time in which it takes an eddy to make a complete revolution—and in a turbulent flow it is also, significantly for our purposes, proportional to the time in which significant deformation of a structure takes place. As a result, if Δt_s is much larger than τ, we can expect tracking to be hard or even impossible for any feature tracking method, as the structure may be strongly deformed or even destroyed. However, the fact that (especially for structures of intermediate size in multiscale flows) the turnover time can be much larger than the stepping time, will allow us to track structures even for cases with $\Delta t_s \gg \Delta t$, as long as $\Delta t_s \lesssim \tau$.

4.2 Algorithm

With the preceding discussion in mind we are nearly ready to formulate a method for tracking coherent structures in a CFD data set. It would appear that we could simply, first, select a streamline associated with a structure of interest, passing through an area of minimum dissipation at a time t; second, advect a point on the streamline where dissipation is minimal; and third, integrate a new streamline passing through the resulting advected point at time $t + \Delta t_s$, where $\Delta t_s < \tau$. The process would then be repeated for successive time steps.

A number of difficulties arise with this simple approach. First, while at the initial time step we are at liberty to select a point possessing minimum dissipation in the vicinity of the structure, we are not guaranteed that successive streamlines at later time steps will pass through regions of low dissipation. Thus, the effects of diffusion or dissipation may not be avoided at later times. Second, streamlines are not in general confined to a single structure; they may wander around the domain, indefinitely in some cases, changing topological properties (e.g., helical winding) as they do. How to constrain a streamline to a single structure is unclear. Finally, a pragmatic issue presents itself; structures of interest are often defined and visually presented by thresholding scalar quantities. Vortices, filaments, and sheets can often most quickly be revealed by displaying isosurfaces or direct volume renderings of certain scalar fields such as the magnitude of vorticity, velocity, or magnetic field. It is these

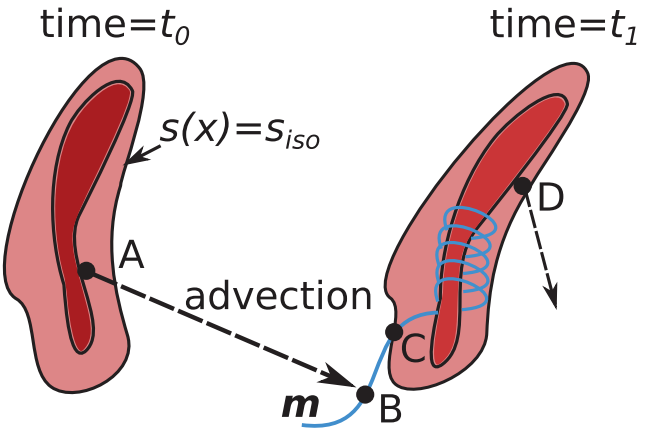


Fig. 3. This figure illustrates the feature path prediction and tracking algorithm. A feature defined by an isosurface of a scalar quantity $s(\mathbf{x}) = \mathbf{s_{iso}}$ is identified at time t_0. The feature at t_0 is labeled with a connected component algorithm, and the location of minimum dissipation within the feature is found at point A. Point A is advected by the velocity field to a new location, point B, at time t_1. Point B happens to reside outside of the tracked feature at time t_1. A streamline $\mathbf{m}$ of a vector field associated with the flow, velocity for example, is integrated through point B. The streamline $\mathbf{m}$ is then traversed to find all features intersecting the material line. In this example there is a single intersection at point C. Correspondence matching takes place using only the intersected features as candidates for matching, and if successful, a new point, D, of minimum dissipation is found and will be advected to time t_2.

thresholded quantities that researchers are most often interested in tracking, and to which the majority of the feature tracking literature has addressed. Thus, we desire an algorithm that avoids the difficulties mentioned above, while providing the convenience of tracking the motion of thresholded scalar quantities.

A naive refinement of the preceding algorithm would be to ignore material lines altogether and simply advect a point within a structure, as defined by a thresholded quantity, where the dissipation is minimal. In addition to operating directly on thresholded scalars, this would increase the probability of finding the region of lowest dissipation (or diffusion if magnetohydrodynamic) associated with the structure by virtue of increasing the search domain. However, (1)-(3) do not ensure that a point within a structure will remain in that structure—defined by a thresholded quantity—at successive time steps. They tell us only that a material line will remain with the structure-although it may enter and leave the thresholded quantity numerous times (see the point labeled "B" in Fig. 3).

With this knowledge we arrive at a final version of our algorithm:

1. Threshold at time t a quantity of interest that defines all structures in the domain. Select the structure, f^t, that we wish to track and gather whatever attributes we will use for correspondence matching.
2. Within the selected structure find the point of minimum dissipation, and advect that point by the velocity field to time $t + \Delta t_s$.
3. The advected point may or may not reside within the structure, $f^{t+\Delta t_s}$ (as defined by the thresholded quantity). Integrate a streamline $m^{t+\Delta t_s}$ passing through the new point in both the forward and backward direction.

4. Search along the streamline, $m^{t+\Delta t_s}$, for curve segments that are within the thresholded value. Each of these segments is contained within some structure $f_i^{t+\Delta t_s}$ that is a candidate for matching the tracked structure associated with the streamline. Perform correspondence matching between all structures $f_i^{t+\Delta t_s}$ intersected by the streamline $m^{t+\Delta t_s}$ and the tracked structure f^t.
5. Select the best matching structure for the streamline and repeat the process.

5 Implementation

Implementation of the algorithm is fairly straight forward, and was performed by extending the Open Source VAPOR visualization package [20], [21], [22]. VAPOR already provides methods for advecting field lines [22], and has much of the machinery needed for implementing our algorithm.

5.1 Initialization

Features are selected by displaying isosurfaces of an isovalue s_{iso} of a desired scalar field, $s(\mathbf{x}), \mathbf{x} \in \mathcal{R}^3$, wherever $s(\mathbf{x}) = s_{iso}$. Once a feature, f^t, of interest at an initial time step t is identified by visual inspection its constituent voxels are labeled as members using a connected component algorithm. A discrete voxel with coordinates $\mathbf{x}$ is considered a connected component of the structure if $s(\mathbf{x}) \geq s_{iso}$ (or $s(\mathbf{x}) \leq s_{iso}$ depending on the quantity), and it is face adjacent to a voxel already considered a connected component (or $\mathbf{x}$ is the initial seed point; the first voxel considered).

In addition to labeling the constituent voxels, the connected component algorithm also computes the volume of the structure (number of voxels), and identifies the coordinates of the voxel with minimum dissipation. Dissipation is computed by (6) and estimating spatial derivatives using a sixth-order centered finite difference scheme applied to the velocity field.

5.2 Advection

Once a point with minimum dissipation within the structure is located, it is advected by the velocity field $\mathbf{v}$ to a new location at time $t + \Delta t_s$. An adaptive line integration [23] using a fourth-order Runge-Kutta scheme is used for both steady (streamline) and unsteady (advection) integration. The values of the velocity field $\mathbf{v}$ at an arbitrary point used in the integration are determined by a trilinear interpolation of velocity values defined on integer grid coordinates; and, for time-varying integration, the field values are linearly interpolated between time steps. As the integration proceeds, the interval size is iteratively doubled or halved to ensure that the angular change between successive line intervals stays between three and 15 degrees. The integration accuracy can be controlled by adjusting the minimum and maximum length of the integrated line intervals. At the lowest accuracy level (0.0) the interval size lies between four and 10 grid cells; at the highest accuracy level (1.0), the step size is between 0.05 and 0.25 grid points. For all of our testing we use an integration accuracy level of 0.9, allowing the integration interval to vary between 0.4 and 1.2 grid points.

5.3 Streamline Integration

The point resulting from advection from time t to $t + \Delta t_s$ becomes the seed point for streamline integration. Streamline integration uses the method just described for advection in steady fields, however, the integration is carried out both forwards and backwards. The length of the integration is fixed to approximately equal the maximum length of the domain (or until the streamline exits the domain). The vector field $\mathbf{v}$ used for streamline integration may be velocity or a derived vector field such as vorticity, or magnetic field (for magnetohydrodynamic case). Following the integration of the streamline in $\mathbf{v}$ the discrete curve is traversed forwards and backwards until the end of the curve is reached. The result of this step is zero or more points contained within candidate structures $f_i^{\Delta t_s}$

5.4 Correspondence Matching

If no points along the streamline of $\mathbf{v}$ at time $t + \Delta t_s$ intersect a structure then tracking cannot proceed; the structure has been lost. If one or more points are contained within structures each of those points is used to seed the connected component algorithm. Note that it is possible for multiple points to reside in the same structure. The resulting labeled structures, $f_i^{t+\Delta t_s}$, are correspondence matched against the tracked structure f^t from the preceding time step using simple volume differences. The best match is considered the structure with the minimal relative volume difference between it and the tracked structure. After the best matching structures are found for each tracked structure the entire process is repeated, starting with finding a new advection point in the region with minimum dissipation. If multiple structures are tracked we emphasize that for each tracked structure f_j^t the list of candidate structures for correspondence matching are only those structures intersecting the single streamline, $m_j^{t+\Delta t_s}$ resulting from the advection of the point of minimum dissipation within f_j^t.

The algorithm is illustrated in its entirety in Fig. 3.

5.5 Multiresolution Data

Lastly, we mention that VAPOR supports a multiresolution data model [20], enabling the structure advection to be performed and evaluated at coarser grid resolutions. We find this particularly valuable when working with high-resolution, time-varying data sets, where the analysis costs are often dominated by the time it takes to read data from disk. In general, analysis techniques can be applied at lowered resolution to provide an initial impression, and subsequently verified later at the native grid resolution. We are interested in the sensitivity of our method to grid coarsening.

6 Results

We tested the method using three different data sets. We first consider the data stemming from a direct numerical simulation of rotating, incompressible turbulence in a periodic box with spatial resolution of $1{,}536^3$ grid points [24]. The kinematic viscosity is $\nu = 1.6 \times 10^{-4}$ and the rotation rate of the fluid is $\Omega = 9$, which leads to a Reynolds number $Re = 5{,}100$ and a Rossby number $Ro = 0.06$. The

Reynolds number is the ratio of advection to dissipation, which controls the separation of scales and thus the smallest structures that develop in the flow. $Re = 5{,}100$ also indicates that though the fluid is not ideal, dissipation is small relative to advection, suggesting that the data are well suited to our algorithm. The Rossby number is defined as the ratio of advection to rotation, Ω, or of the rotation period to the turnover time. The time step for the simulation is $\Delta t = 2.5 \times 10^{-4}$. The components of the velocity field $\mathbf{v}$ were output every 500 time steps, leading to $\Delta t_s = 500\Delta t = 0.125$. The characteristic turnover time for large-scale eddies in the simulation is $\tau \approx 1.0$, thus $\Delta t_s \approx 0.125\tau$. As a result of rotation, the flow develops large-scale columnar structures that live for long times and are advected by the velocity field (see Fig. 5).

The second data set is from a direct numerical simulation of isotropic and homogeneous Taylor Green (TG) incompressible turbulence [25] using $1{,}024^3$ grid points in a periodic box. Kinematic viscosity is $\nu = 3 \times 10^{-4}$ and the Reynolds number is $Re = 3{,}950$. Forty one single-precision snapshots of the velocity and the vorticity fields were available to us, for a total of ≈ 1.0 TB, with $\Delta t_s \approx 0.01$. In the next section, it will be shown using the turnover time of small-scale structures that $\Delta t_s \approx 0.03\tau$. The flow is characterized by a myriad of small-scale vortex filaments, which have a mean width of ≈ 0.003 of the size of the box, and a mean length of ≈ 0.07 (see Fig. 7).

The first data set has large scale structures that evolve slowly as a result of a self-organization process. On the other hand, in the isotropic and homogeneous turbulence data set the energy cascades to smaller scales, creating smaller and smaller structures. These structures evolve much more rapidly, hence the differences in the Δt_ss for the two data sets.

The third data set is from a simulation by the Weather Research Forecast Model (WRF-ARW) [26] of Hurricane Bill that occurred in the Atlantic Ocean in August 2009. The model uses three nesting levels to track the hurricane; however, our feature tracking experiments used only the outer nesting level, a terrain-following grid of dimensions $468 \times 423 \times 35$, with 40 snapshots, three hours apart ($\Delta t_s = 3$ hours), starting at midnight on August 18.

6.1 Rotating Turbulence

We first present results tracking columnar structures defined by isosurfaces of the Z component of velocity, v_z, in the $1{,}536^3$ rotating turbulence data set. These data are of particular interest due to their size—a single snapshot at 32 bit precision occupies over 14 GBs of space per variable—as well as the rather coarse time sampling of the available, stored solution data. Only every 500 time steps (Δt) were output. Furthermore, only every tenth output snapshot saved was available to us for feature tracking. The output snapshots are indexed from 0 to 216, and Δt_s is the time sampling between index i and $i+1$. Fig. 5 shows the tracking of a single structure at outputs 136, 146, 156, and 216 from left to right, top to bottom. Note, that the tracked structure crosses the periodic lower boundary a number of times. All of the available stored data, every tenth output from 136 to 216, were used by the tracking algorithm. Thus, the effective $\Delta t_s \approx 1.25\tau$, which is $10\times$ the real Δt_s. Recall that if Δt_s is much larger than τ, we can expect tracking to be hard or even impossible, as the structure may be strongly deformed or even destroyed.

Two isovalues are shown: $v_z = -1.0$, colored cyan, and $v_z = +1.0$, colored yellow. Inside the cyan structure $v_z \leq -1.0$, while within the yellow structures $v_z \geq 1.0$. The tracked structure is defined by the negative isovalue, $v_z = -1.0$, and is colored magenta. Also shown in red are the streamlines of velocity integrated at each time step.

At the initial time step used for tracking, output 136 (upper, left), a streamline of the velocity field is integrated from the point of minimum dissipation within the tracked structure.[1] The streamline is almost completely contained within the structure boundaries. At subsequent time steps the velocity streamlines are integrated from the point advected from the previous output at the location of minimum dissipation within the tracked structure. Note how in subsequent outputs the streamlines continue to wrap around the tracked structure, but are not necessarily confined to the isosurface boundaries. Indeed, at output 146 (upper, right) the streamlines wander to another structure entirely. Rotating turbulence is characterized by the development of large scale columns in the flow ("Taylor columns") and by a self-organization process by which these columns merge to create larger columns (a process called "inverse cascade" of energy in turbulent flows, by which energy is transferred from smaller to larger scales). When the merging takes place, streamlines belonging to two or more different structures open and reconnect to create a thicker streamline that goes around both, until the structures merge into one. Without correspondence matching the tracked structure at output 146 would be ambiguous due to the streamlines intersecting two candidate structures.

At the coarse temporal resolution of these data the tracked structure does not have spatial overlap between the outputs used for advection. Indeed, the time between snapshots is ≈ 25 percent larger than the turnover time, τ. Since only one of every 10 outputs is used, this time is $10\Delta t_s = 1.25$ in units of the turnover time. Thus, substantial displacement and deformation of the structures take place in between. Nevertheless, the algorithm performs effectively at tracking the structure through the completion of the simulation at output 216.

In Fig. 6, we show a close up of the tracked structure at output 176. A gray diamond marks the advection point from the previous time step. Note that the advected point is not inside of the structure, but the resulting stream line integrated starting at the location of the diamond winds its way back inside of the tracked feature as predicted.

Finally, we remark that because of the unwieldy size of this data set we chose to take advantage of VAPOR's wavelet-based, multiresolution data model to speed processing; data were coarsened from their original $1{,}536^3$ to 384^3 grid points for preliminary work. Nevertheless, final results with the native grid resolution were indistinguishable from results with the coarsened data, suggesting that for these data our method is insensitive to fairly aggressive data reduction by grid point averaging.

1. We note that the streamline integration at the initial time step is superfluous to the algorithm and is only shown for illustrative purposes.

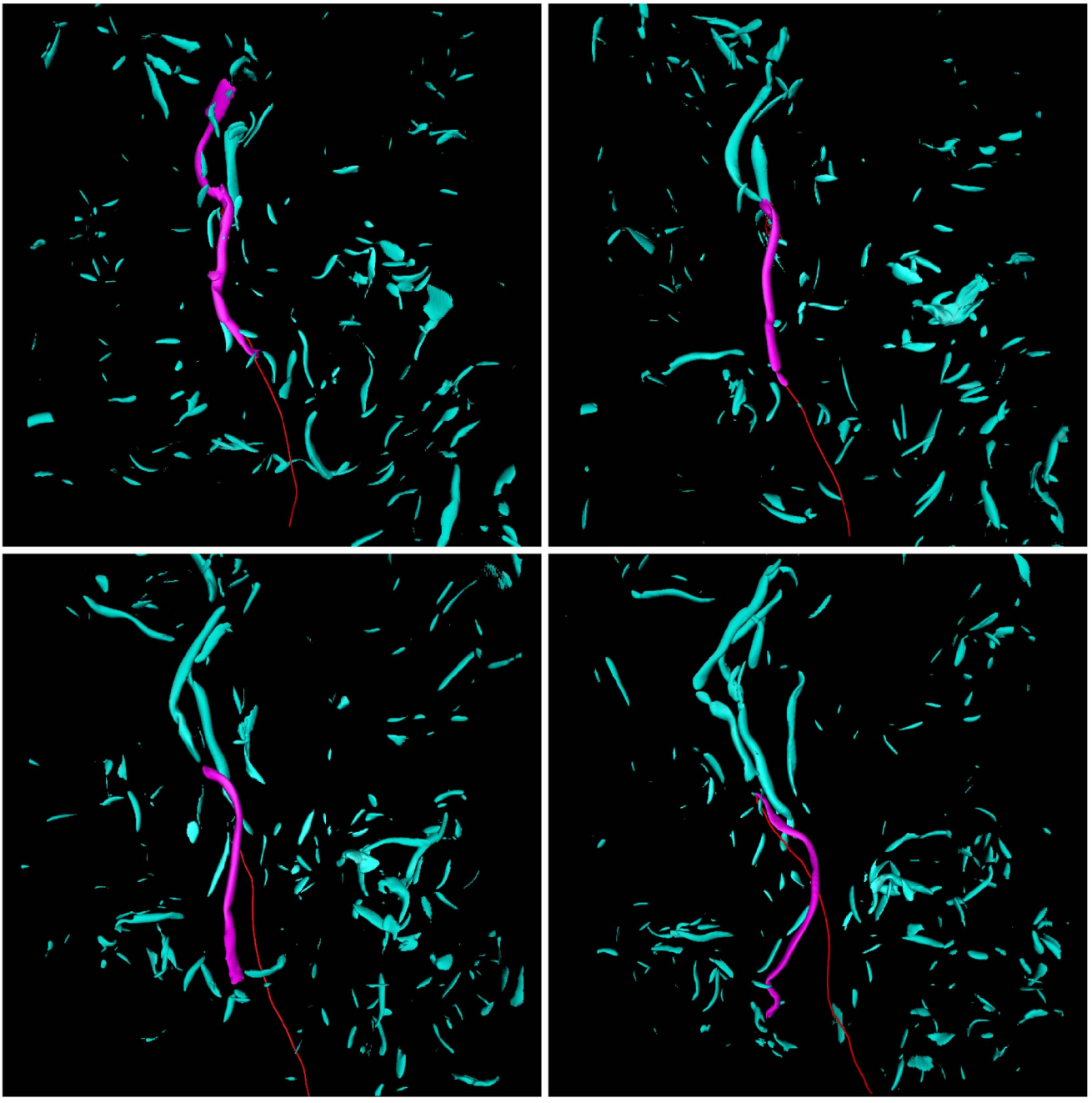


Fig. 4. Here, we illustrate the tracking of a vortex filament (shown in magenta) that is large and long-lived relative to other filaments in the surrounding domain of the Taylor Green data set. All 41 available snapshots were used for tracking. Shown from left-to-right, top-to-bottom are snapshots numbered 1, 14, 27, and 41, respectively, with $\Delta t_s \approx 0.01$. The structures are defined by an isosurface of the magnitude of the vorticity field. A single red streamline of the velocity field intersects the structure at each snapshot shown. The streamline is integrated through a point advected from the region of minimum dissipation within the structure at the preceding time step. The structure bifurcates near snapshot 14 (upper, right) and late in the simulation at snapshot number 41 (bottom, right) the structure has deformed noticeably.

6.2 Taylor Green Turbulence

The TG data set is challenging due to both its spatial resolution, $1{,}024^3$, and the number and complexity of the structures contained within as can be seen in Fig. 7, which shows a direct volume rendering of the magnitude of the vorticity field. Thousands of vortex filaments with complex motion and evolution are present. As a first experiment we elect to isolate and track one particularly large, and long lived filament as shown in Fig. 4. We again use minimum dissipation to select our advection point. Over the 41 snapshots for which we successfully track this feature it changes shape and size, bifurcating once about a quarter of the way through the sequence.

The visualized structure, an isosurface of vorticity magnitude with an isovalue of ≈ 78, has a length and width ≈ 0.8 and ≈ 0.08, respectively, (in a domain of size 2π). Its turnover time is estimated to be $\tau \approx 0.3$ (recall that $\Delta t_s = 0.01 \approx 0.03\tau$, substantially smaller than τ and suggesting minimal deformation between successive output time steps). In the final output (bottom, right of figure) significant deformation of the structure since the initial output (upper, left of figure) is observed, which is to be

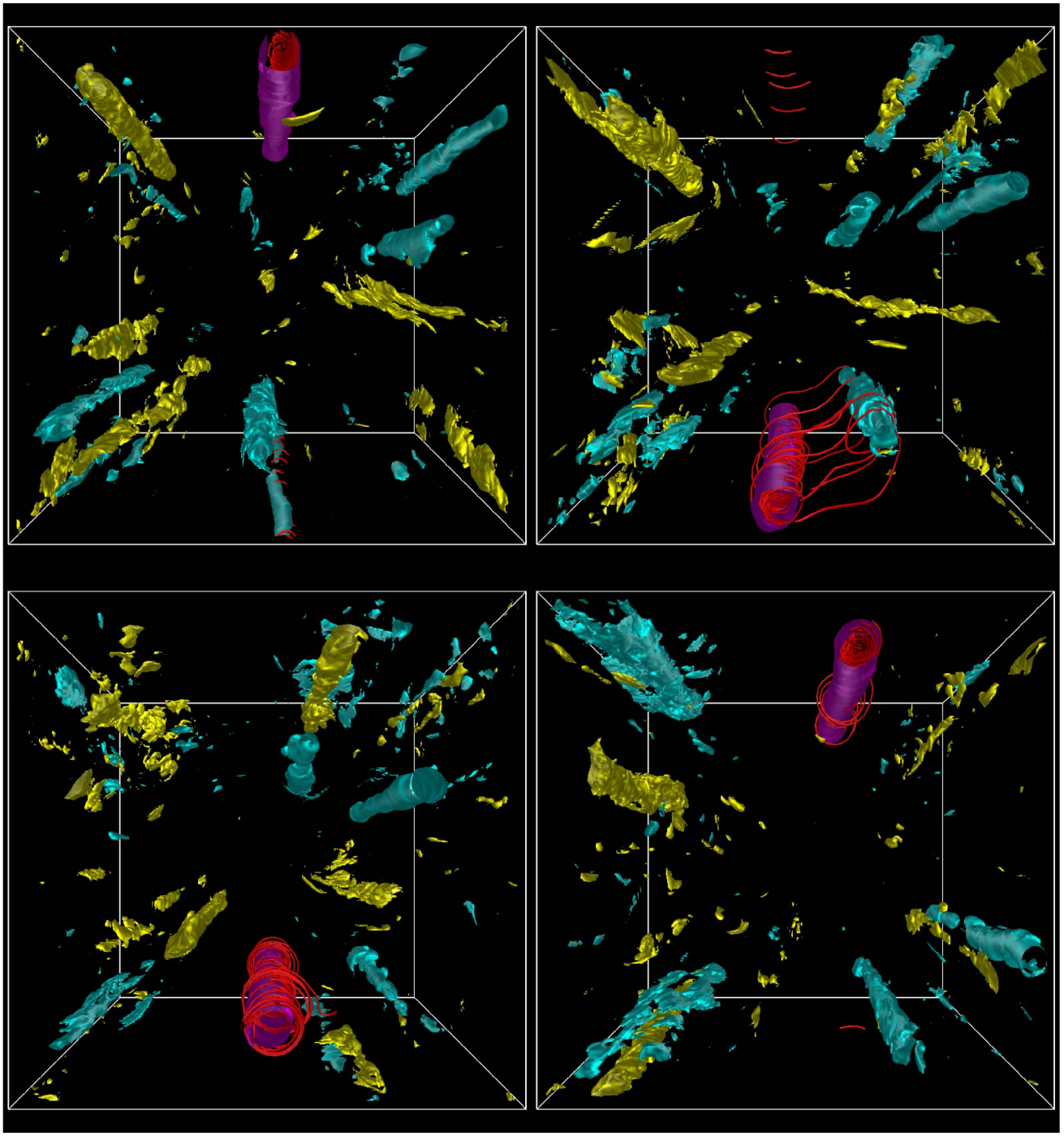


Fig. 5. Shown here are a sequence of images depicting the tracking a columnar structure from a $1{,}536^3$ hydrodynamics simulation with periodic boundary conditions. Snapshots numbered 136, 146, 156, and 216, with $\Delta t_s = 0.0125$, are shown from left-to-right, top-to-bottom, respectively. Positive and negative columns of the Z component of velocity are shown as yellow and cyan isosurfaces, respectively. The tracked structure, colored magenta, is an isosurface of $v_z = -1$. Streamlines (in red) of the velocity field are integrated from advection points arising from regions of minimum dissipation and are used to track the structure through time.

expected in a turbulent flow as the turnover time is proportional to the deformation time.

To explore the sensitivity of our path prediction to temporal sampling we coarsened the temporal resolution to find the maximum Δt_s between snapshots that still allowed us to track the structure; even with $\Delta t_s \approx 0.1 \approx 0.3\tau$ (using one snapshot for every 10 available) the structure can still be tracked. However, if we coarsen the temporal sampling even further, using only one snapshot for every 20 available ($\Delta t_s \approx 0.2 \approx 0.6\tau$), the method fails to correctly track the structure.

6.2.1 Tracking Multiple Structures

As a final experiment with the TG data we attempted to simultaneously track multiple structures. We selected 10 of the largest structures by volume from a subregion of the full domain—as seen in the left image of Fig. 8—that are completely contained within the subdomain. The total

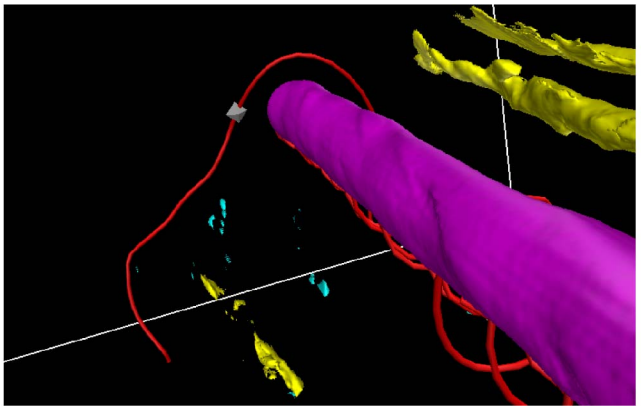


Fig. 6. A close up of the tracked structure from output number 176 of the rotating turbulence data. A gray diamond denotes the location of the point advected by the velocity field from the previous output, 166, at the location of minimum dissipation. Note that the diamond is not contained by the structure. However, streamlines of the velocity field integrated through the diamond intersect the structure providing a way to correspond the structure at output 176 with the structure at output 166 (not shown).

population of structures in the subdamain was $\approx$400, ranging in size from 16 to 5,544 voxels, with an average volume of 173 voxels. The 10 structures selected for tracking ranged in size from 336 to 5,544 voxels, or 0.00003 to 0.0005 percent of the total domain volume, respectively.

Of the 10 structures tracked, two were successfully so for all 40 time steps. Five of the structures were successfully tracked until they either exited the domain, or dissipated to zero volume. The remaining three structures were not successfully tracked throughout their lifetime. Of these three, two of the smallest bifurcated and in both cases the material lines resulting from advecting the point of minimum dissipation at the previous time step failed to pass through either of the child structures. The third lost

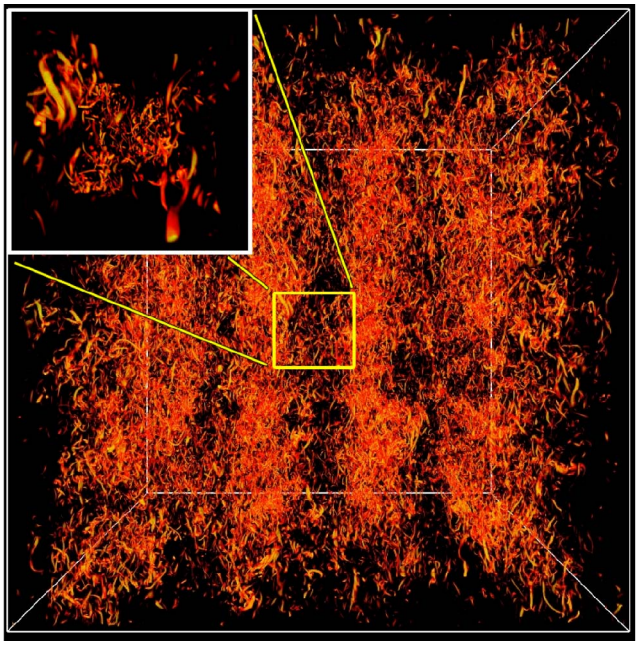


Fig. 7. A direct volume rendering of vorticity magnitude from a $1{,}024^3$ TG simulation. The flow is characterized by the presence of a myriad of fast moving, short-lived, small-scale vortex filaments.

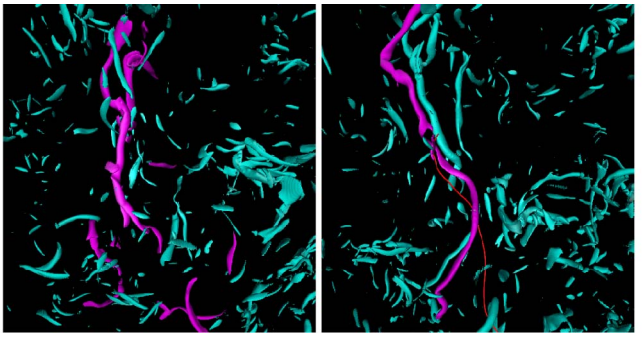


Fig. 8. Initial (left) and final (right) time steps showing the tracking of multiple vortices from the Taylor Green data set. Ten large structures (colored magenta, left image) were selected for tracking. After 40 time steps two structures remain, five structures either exited the domain or completely dissipated, and three smaller structures were not successfully tracked.

structure also bifurcated, but in this case the advected material line did in fact pass through one child. However, the material line also passed through a second structure and our simplistic correspondence matching algorithm incorrectly selected the second structure. We note that on average a material streamline passed through $\approx$1.2 structures at each time step. Table 1 summarizes the results, showing: the time-step index at which an event takes place; the number of structures successfully being tracked at that time; and the event type. An event type is one of:

miss A material line advected from time step t did not pass through the tracked structure at time step $t + \Delta t_s$.

mismatch The advected material line intersected multiple structures, including the correct one, but the correspondence algorithm failed to identify the tracked structure.

dissipation The volume of a tracked structure became zero.

exit A tracked structure exited the domain.

6.3 Hurricane Bill

We finally focus on the simulation of a hurricane generated by the Weather Research Forecast Model (see Fig. 9). A streamline of the wind field was seeded in the hurricane vortex at point of maximal wind velocity at the initial time step, and advected by the wind field. The vortex diameter is

TABLE 1
Tracking Multiple Structures in the TG Data Set

Time Step Offset	Structure Count	Event
1	10	
8	9	miss
21	8	dissipation
23	7	mismatch
24	6	miss
30	5	dissipation
31	4	exit
34	3	dissipation
40	2	exit

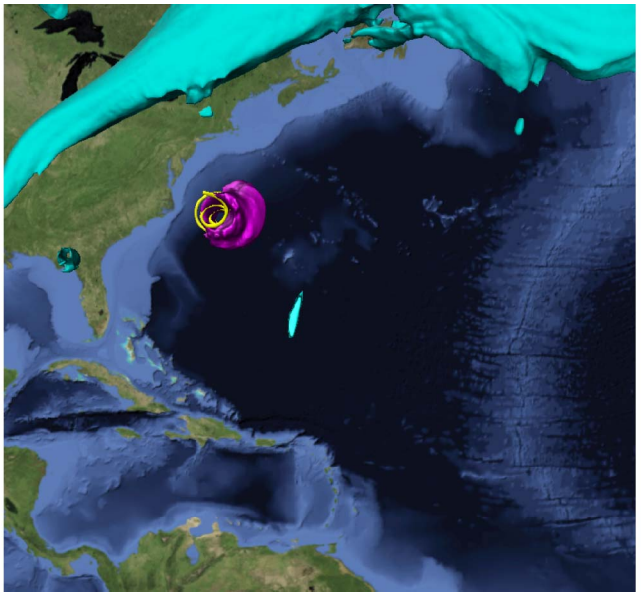


Fig. 9. A simulation of hurricane bill as it approaches the south east coast of the United States. The tracked structure—an isosurface of velocity magnitude shown in magenta—depicts the swirling hurricane vortex, hundreds of kilometers in diameter and with a turnover time on the order of 9 hours.

approximately 300 km, and the RMS of wind velocity in that region containing the vortex is approximately 30 m/s, indicating an approximate turnover time of $\tau = 9$ hours. Thus, $\Delta t_s \approx 0.3\tau$. We were able to successfully track the vortex, defined by isosurface $\|\mathbf{v}\| = 30$ m/s, for all 40 outputs (120 hours). As we did with the TG data we increased the effective Δt_s by omitting time steps until the algorithm failed, which occurred when $\Delta t_s \approx 1.5\tau$, using only every fifth saved time step.

6.4 Comparison with Other Prediction Methods

To compare the feature path prediction component of our method, which is based on the equations of fluid dynamics, with other prediction methods, we have implemented the prediction method of Muelder and Ma [9]. Muelder and Ma's method predicts the position of features at future time steps by extrapolating their position from previous time steps using either direct, linear, or quadratic extrapolation. For example, the position of a point, $\mathbf{x_t}$, relative to the containing structure's center $\mathbf{c_t}$ at time t is predicted using linear extrapolation by

$$\mathbf{x_t} = \mathbf{x_{t-1}} + (\mathbf{c_{t-1}} - \mathbf{c_{t-2}}). \quad (7)$$

Note that for linear extrapolation two preceding time steps, $t-1$ and $t-2$, are needed for the current time step, t. Quadratic extrapolation requires three previous time steps.

Once a feature is predicted, correspondence matching occurs between the predicted and actual features found at time t. Similar to our own method the features compared against the predicted feature are a subset of all the features in the domain: those features that possess spatial overlap with the predicted feature location. Thus, if the tracked feature does not overlap the predicted feature location at time t, the method fails.

We tested both linear and quadratic extrapolation with the rotating turbulence data set, attempting to track the same structure successfully tracked by our own method as reported in Section 6.1. The rotating turbulence data set is sampled coarsely in time. Nine time steps are available for tracking. Linear extrapolation requires two initial time steps to boot strap itself, leaving seven additional time steps for which we can track the structure. Of these seven time steps the linear prediction method was able to correctly predict and correspond the feature at only two time steps. Quadratic extrapolation requires three initial time steps to boot strap, leaving only six time steps for tracking. Using quadratic extrapolation the method performed slightly better and was able to correctly track the structure at three of the six time steps. We contrast this with our method, which successfully tracked the structure for all nine available time steps.

7 Discussion

The method that we have presented for predicting the trajectory of coherent structures in CFD is inspired by the physics of fluid motion. The foundations of this method are:

1. There exist *exact* theorems in fluid dynamics that give a precise prescription of how certain field lines evolve in time when the flow is *ideal* (nondissipative).
2. Flows in nature, and in numerical simulations in general, however, are not ideal, and dissipation is present.

To circumvent the problem of dissipation, we use a heuristic approach to advect fluid elements based on the theorems of fluid dynamics, and assume the points with minimum dissipation—or, in some cases, points with a minimum of certain quantities associated with the dissipation rate—follow, for a presumably short amount of time, a path along which dissipation remains small. Thus, the prescription given by the physical theorems gives a good approximation to the actual path. This hypothesis can be only justified a posteriori, considering the results we have obtained.

7.1 The Role of Time Sampling

In the three data sets the time sampling used for tracking ranges from a fraction of the turnover time to slightly more than the turnover time. The best results are obtained when Δt_s is smaller than the turnover time τ. For the TG data set three outputs per turnover was the minimum required to track a large structure during its lifetime. This data set, which corresponds to isotropic and homogeneous turbulence, is characterized by a "direct cascade" of energy, a process associated with the transfer of energy from larger to smaller scales, creating a myriad of vortex filaments at small scales. The smallest scale excited (and the cluttering of the structures at that scale) is controlled by the Reynolds number; the larger this number, the more complex the flow and the larger the separation between the largest and smallest structures in the flow. When we attempted to track multiple vortex filaments from this data set, the filaments we tracked varied in volume size by over an order of magnitude. Consequently, the turnover times of the

TABLE 2
Maximum Δt_s Expressed in τ for Successful Tracking

Data Set	Δt_s
Rotating turbulence	1.25
TG	0.3
Hurricane Bill	1.5

smaller structures can be expected to be shorter making them prone to greater deformation and displacement, and therefore more difficult to predict their paths or perform correspondence matching. Hence, our inability to track the smallest structures in the TG data set for long periods of time is not unexpected.

In the other two data sets, the time sampling can be made slightly larger than the turnover time, per virtue of the presence of slowly evolving structures at the largest scales in the flow. We conclude that in general a few outputs per turnover time may be enough to track structures using this method—which in practice results in one output every hundreds of numerical time steps Δt—and if large-scale structures develop in the flow the number of outputs can be decreased further.

Structures in the Hurricane Bill data sets can be tracked over very long times (much larger than their turnover times) using coarse temporal resolution of the order of the turnover time. We believe this is because the structures in these data sets are at large scales and stable, in the sense that they result from a self-organization process. On the other hand, structures in the TG data set, which corresponds to small-scale fully developed three dimensional turbulence, can be tracked for only a few turnover times. In this case, the structures deform rapidly, or are destroyed, hence the need for increased temporal resolution.

Table 2 summarizes the maximum Δt_s, expressed in units of τ, for which single-structure tracking was successful for all three data sets.

7.2 Other Coherent Structures

All coherent structures in CFD share similarities with the paradigmatic cases discussed here (corresponding to either small-scale features, or to large-scale structures resulting from self-organization processes). Examples of the first case are given by streaks, horseshoe and hairpin vortices, and structures observed in flows near walls, which share structural properties with the vortices tracked in the TG data set. An example of the second case are magnetic flux tubes, which are large-scale structures often observed in conducting flows and plasmas. In all cases, the arguments outlined in Section 3 hold for the different fields associated with these structures (velocity, vorticity, or the magnetic field). From the geometrical point of view, a somewhat different structure of interest to track is given by current sheets in conducting flows. These are often associated with abrupt changes in the direction of the magnetic field, and with critical points. In this context, we found that advection of points of minimum dissipation and integration of field lines successfully tracked current sheets in a simulation of a conducting flow at a resolution of $1{,}536^3$ grid points for several turnover times.

7.3 Algorithm Efficiency

We have stated that our algorithm is computationally efficient and this warrants discussion. First, by virtue of narrowing the set of candidate structures for correspondence matching (≈ 1.2 structures on average in the case of the densely populated TG data set) great savings in the cost of executing the correspondence matching algorithm can be realized because so few candidate structures exist. Second, the path prediction algorithm itself can be implemented efficiently. Its cost is proportional to the number and sizes of features tracked. Only a single point is advected for each structure, and only a single streamline is generated at each time step. While the required dissipation field defined by (6) is computed using sixth-order finite differences, requiring ≈ 40 multiplies and ≈ 40 adds per grid point, the calculation need only be performed within the tracked structure, which for many applications only occupies a fraction of the domain. Moreover, because the error introduced would not propagate in time, low ordered finite difference schemes might be perfectly acceptable.

7.4 Limitations

The primary contribution of our work is a novel method to predict the path of coherent structures arising from CFD simulations. We have demonstrated the viability of this approach through the implementation of a simplistic feature tracking algorithm that uses only the most rudimentary form of correspondence matching. As such, limitations result from our proof of concept implementation that are not inherent to our path prediction method. Two issues in this regard are threshold selection and feature event handling, specifically bifurcation.

The choice of thresholds in our study was not made by any physical property of the scalar quantities, but chosen somewhat arbitrarily such that visually distinct and well-defined structures resulted. Lacking a rigorous definition of what constitutes a coherent structure such imprecise selection is in practice the norm. A different choice of threshold value might still reveal what could reasonably be termed coherent structures, but might yield different tracking results from our demonstration implementation. For example, advected material lines that pass through a structure using one threshold value might miss the structure entirely with another value. A possible solution would be a more complex algorithm for determining candidate structures that would relax the requirement for structure intersection, and instead search for structures in the vicinity of aa material line. Relatedly, we have somewhat arbitrarily chosen to terminate streamline integration when the length of the resulting streamline approaches the domain size (see Section 5.3). This choice is not based on physical properties, but simply because in practice it produces good results. A more robust termination criterion might consider whether the streamline has intersected any candidate structures.

Our implementation does not handle structure bifurcation. However, a bifurcation of a thresholded quantity does not imply that the underlying topological properties germane to our path prediction method have themselves bifurcated. Again, a more sophisticated implementation could detect splitting of a thresholded quantity, yet still utilize the advected material lines as an aid to identify the resulting new thresholded structures.

Much has been reported in the literature on the tracking of critical points (see, for example, the survey article by Post et al. [10]). Our method is not specifically designed for, nor have we tested, tracking of these singularities. However, we note that many of the structures discussed in this paper, for example, current sheets, vortex tubes, and other structures, are associated with critical points. Our method tracks the field lines in the vicinity of such structures, thus providing an implicit form of critical point tracking that may be suitable for some applications. The explicit tracking of critical points using our path prediction method is a subject for future investigation.

8 Conclusion

We have presented a method for forecasting the motion of coherent structures resulting from numerical simulations of fluid flow, and demonstrated accurate feature path prediction's value to feature tracking. Unlike previously reported approaches ours uses properties of the underlying equations of fluid dynamics to accurately determine the position of a structure at a point in time based on an initial location. For simulations of ideal fluids, when the effects of dissipation or diffusion are not present, the solutions to the equations of motion yield exact predictions. For the more general case, when dissipation is present, we have shown empirically with three separate CFD data sets that restricting advection to regions within a structure where dissipation is locally minimal produces viable results as well. We remark that we have demonstrated success tracking coherent structures using only a single, inexpensive correspondence matching test, volume difference, and that our main contribution in this work is highly reliable path prediction. More sophisticated and robust correspondence matching may certainly be applied if the application warrants. The benefit of combining more advanced matching with accurate feature motion forecasting is the potential substantial reduction in the number of candidate structures to test, which both reduces cost, and improves reliability.

Regardless of whether fluids are idealized or not, the temporal spacing between time steps plays a large role in the accuracy of the method, as is the case with any feature tracking method applied to evolving structures. The time sampling between stored frames, Δt_s, is typically larger, sometimes significantly so, than the time sampling, Δt used by the simulation, leading to inaccuracies in the advection calculations, and the opportunity for significant physical changes in features of interest between time steps. However, $\Delta t_s \gg \Delta t$ may not pose a problem provided that the turnover time, τ, of a structure is not much smaller than Δt_s. The physical properties of the fluid ensure that the deformation and motion of a structure over a period of τ is bounded. Thus, the second major contribution of our work is to provide guidance regarding the maximum time sampling between snapshots for *any* correspondence-based feature tracking method.

Acknowledgments

This work is funded in part through US National Science Foundations (NSF) grants 03-25934 and 09-06379. The authors thank Mark Rast, University of Colorado, for his insightful discussions on numerically modeled turbulence.

References

[1] D. Silver, N. Zabusky, V. Fernandez, M. Gao, and R. Samataney, "Ellipsoidal Quantification of Evolving Phenomena," *Scientific Visualization of Physical Phenomena,* pp. 573-588, Sprnger-Verlag, 1991.

[2] R. Samtaney, D. Silver, N. Zabusky, and J. Cao, "Visualizing Features and Tracking their Evolution," *Computer,* vol. 27, no. 7, pp. 20-27, July 1994.

[3] D. Silver and X. Wang, "Tracking and Visualizing Turbulent 3d Features," *IEEE Trans. Visualization and Computer Graphics,* vol. 3, no. 2, pp. 129-141, Apr. 1997.

[4] F. Reinders, F.H. Post, and H.J.W. Spoelder, "Attribute-Based Feature Tracking," *Data Visualization,* pp. 63-72, Springer Verlag, 1999.

[5] F. Reinders, F.H. Post, and H.J.W. Spoelder, "Visualization of Time-Dependent Data with Feature Tracking and Event Detection," *The Visual Computer,* vol. 17, pp. 55-71, 2001.

[6] G. Ji, H.-W. Shen, and R. Wenger, "Volume Tracking using Higher Dimensional Isosurfacing," *Proc. IEEE 14th Visualization (VIS '03),* p. 28, 2003.

[7] G. Ji, "Feature Tracking and Viewing for Time-Varying Data Sets," PhD dissertation, Columbus, OH, adviser-Shen, Han-Wei, 2006.

[8] J. Caban, A. Joshi, and P. Rheingans, "Texture-Based Feature Tracking for Effective Time-Varying Data Visualization," *IEEE Trans. Visualization and Computer Graphics,* vol. 13, no. 6, pp. 1472-1479, Nov./Dec. 2007.

[9] C. Muelder and K.-L. Ma, "Interactive Feature Extraction and Tracking by Utilizing Region Coherency," *Proc. IEEE Pacific Visualization Symp.,* pp. 17-24, 2009.

[10] F.H. Post, B. Vrolijk, H. Hauser, R.S. Laramee, and H. Doleisch, "The State of the Art in Flow Visualisation: Feature Extraction and Tracking," *Computer Graphics Forum,* vol. 22, no. 4, pp. 775-792, http://dx.doi.org/10.1111/j.1467-8659.2003.00723.x, 2003.

[11] C. Weigle and D.C. Banks, "Extracting Iso-Valued Features in 4-Dimensional Scalar Fields," *Proc. IEEE Symp. Vol. Visualization (VVS '98),* pp. 103-110, 1998.

[12] H. Theisel and H.-P. Seidel, "Feature Flow Fields," *Proc. Symp. Data Visualisation '03 (VISSYM '03),* pp. 141-148, http://portal.acm.org/citation.cfm?id=769922.769938, 2003.

[13] T. Weinkauf, H. Theisel, A.V. Gelder, and A. Pang, "Stable Feature Flow Fields," *IEEE Trans. Visualization and Computer Graphics,* vol. 17, no. 6, pp. 770-780, June 2011.

[14] Y. Rubner, C. Tomasi, and L.J. Guibas, "A Metric for Distributions with Applications to Image Databases," *Proc. Sixth Int'l Conf. Computer Vision (ICCV '98),* p. 59, 1998.

[15] Y. Rubner, C. Tomasi, and L.J. Guibas, "The Earth Mover's Distance as a Metric for Image Retrieval," *Int'l J. Computer Vision,* vol. 40, no. 2, pp. 99-121, 2000.

[16] H. Peng, F. Long, and C. Ding, "Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy," *IEEE Trans. Pattern Analysis Machine Intelligence.,* vol. 27, no. 8, pp. 1226-1238, Aug. 2005.

[17] G.K. Batchelor, *An Introduction to Fluid Dynamics.* Cambridge Univ. Press, 1974.

[18] H.K. Moffatt, *Magnetic Field Generation in Electrically Conducting Fluids.* Cambridge Univ. Press, 1978.

[19] A. Tsinober, *An informal Introduction to Turbulence.* Kluwer Academic Publishers, 2001.

[20] J. Clyne and M. Rast, "A Prototype Discovery Environment for Analyzing and Visualizing Terascale Turbulent Fluid Flow Simulations," *Proc. Visualization and Data Analysis '05,* pp. 284-294, Jan. 2005.

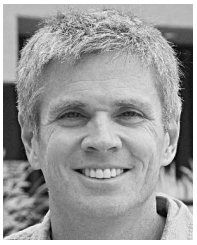
[21] J. Clyne, P. Mininni, A. Norton, and M. Rast, "Interactive Desktop Analysis of High Resolution Simulations: Application to Turbulent Plume Dynamics and Current Sheet Formation," *New J. Physics,* vol. 9, no. 301, 2007.

[22] P. Mininni, E. Lee, A. Norton, and J. Clyne, "Flow Visualization and Field Line Advection in Computational Fluid Dynamics: Application to Magnetic Fields and Turbulent Flows," *New J. Physics,* vol. 10, no. 12, p. 125007, http://stacks.iop.org/1367-2630/10/i=12/a=125007, 2008.

[23] D. Kenwright and D. Lane, "Optimization of Time-Dependent Particle Tracing Using Tetrahedral Decomposition," *Proc. IEEE Visualization '95,* pp. 321-328, 1995.

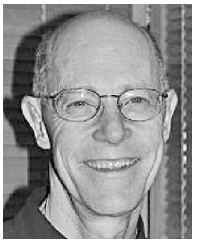
[24] P.D. Mininni and A. Pouquet, "Rotating Helical Turbulence. i. Global Evolution and Spectral Behavior," *Physics Fluids,* vol. 22, p. 035105, 2010.

[25] P.D. Mininni, A. Alexakis, and A. Pouquet, "Large-Scale Flow Effects, Energy Transfer, and Self-Similarity on Turbulence," *Physical Rev.*, vol. 74, p. 016303, 2006.
[26] J. Michalakes, J. Dudhia, D. Gill, T. Henderson, J. Klemp, W. Skamarock, and W. Wang, "The Weather Reseach and Forecast Model: Software Architecture and Performance," *Proc. 11th ECMWF Workshop the Use of High Performance Computing in Meteorology*, 2005.



John Clyne received the master's degree in computer science from the University of Colorado, Boulder, in 1991. He is a senior software engineer at the National Center for Atmospheric Research, where he has worked since 1988. His research interests include large data visualization, the application of wavelets to scientific data sets, and volume rendering.

Pablo Mininni received the diploma in 1999 and the doctoral degree in 2003, both in physics and from the University of Buenos Aires (UBA) in Argentina. From 2004 to 2007, he lived in the United States in Boulder, Colorado, where he was a postdoc at National Center for Atmospheric Research (NCAR). Now, he is a professor and the head of the Physics Department at the University of Buenos Aires, and scientist at NCAR. He works on the numerical and theoretical study of turbulent flows, with applications in geophysics and astrophysics. In the field of fluid dynamics, his expertise includes the application of statistical methods for the characterization and analysis of turbulent flows, spectral analysis of multiscale and multiphysics phenomena, subgrid modeling for turbulent flows, and wave-eddy interactions.



Alan Norton received the PhD degree in mathematics from Princeton University in 1976. He is currently a software engineer at the National Center for Atmospheric Research, where he is a software developer and architect for the VAPOR visualization platform. Previously, he has held research and development positions at IBM Research, Silicon Graphics, Inc., Colorado School of Mines, and the University of Utah, and has written research papers on mathematics, parallel processing, fractals, computer graphics, computer animation, and scientific visualization.