Stream Bundles - Cohesive Advection through Flow Fields

David Weinstein
Gordon Kindlmann
Eric Lundberg

Email: dmw@cs.utah.edu
gk@cs.utah.edu
lundberg@cs.utah.edu

UUCS-99-005

Department of Computer Science
University of Utah
Salt Lake City, UT 84112 USA

June 4, 1999

Abstract

Streamline advection has proven an effective method for visualizing vector flow field data. Traditional streamlines do not, however, provide for investigating the coarser-grained features of complex datasets, such as the white matter tracts in the brain or the thermal conveyor belts in the ocean. In this paper, we introduce a cohesive advection primitive, called a *stream bundle*. Whereas traditional streamlines describe the advection patterns of single, infinitesimal micro-particles, stream bundles indicate advection paths for larger macro-particles. Implementationally, stream bundles are composed of a collection of individual streamlines (here termed *fibers*), each of which only advects a short distance before being terminated and re-seeded in a new location. The individual fibers combine to dictate the instantaneous distribution of the bundle, and it is this collective distribution which is used in determining where fibers are re-seeded. By carefully controlling the termination and re-seeding policies of the fibers, we can prevent the bundle from becoming frayed in divergent regions. By maintaining a cohesive form, the bundles can indicate the coarse structure of complex vector fields. In this paper, we use stream bundles to investigate the oceanic currents.

1 Introduction and Background

Since its introduction, streamline advection [5, 11] has proven an effective method for visualizing vector flow fields. Streamlines are the computational analog of the physical streamers used to evaluate flow in wind-tunnel experiments. Tiny particles are seeded within a velocity field, and their paths are traced as they advect through the field according to the integral equation:

$$\mathbf{p}(s) = \int_0^s \mathbf{v}(\mathbf{p}(\hat{s}))d\hat{s}, \tag{1}$$

where the flow vector $\mathbf{v}$ and the position $\mathbf{p}$ along the path are arc-length parameterized by s.

Advection is accomplished computationally by discretely integrating with methods such as the fourth-order Runge-Kutta method [6]:

$$\mathbf{p}(s+h) = \mathbf{p}(s) + \frac{1}{6}(\mathbf{V}_1 + 2\mathbf{V}_2 + 2\mathbf{V}_3 + \mathbf{V}_4), \tag{2}$$

where

$$\begin{aligned}
\mathbf{V}_1 &= h\mathbf{v}(\mathbf{p}(s)) \\
\mathbf{V}_2 &= h\mathbf{v}(\mathbf{p}(s) + \frac{1}{2}\mathbf{V}_1) \\
\mathbf{V}_3 &= h\mathbf{v}(\mathbf{p}(s) + \frac{1}{2}\mathbf{V}_2) \\
\mathbf{V}_4 &= h\mathbf{v}(\mathbf{p}(s) + \mathbf{V}_3).
\end{aligned}$$

From this initial streamline concept, various extensions have been developed. These include stream-ribbons [12], stream-surfaces [3], stream-polygons [10], streamballs [2], and flow volumes [9]. In stream-ribbons, seed particles are paired together, and as they are advected, they define the two edges of a ribbon. This ribbon can be rendered as a triangle strip, providing the user with depth and occlusion cues. Extending this idea to more than two particles, the user can seed and advect a line of these articles. Connecting the edges swept out by neighboring pairs (and terminating or adding streamlines when neighbors become too close together or too far apart), this algorithm constructs a stream-surface. Alternatively, if instead of seeding particles along a line, one seeds particles at the vertices of a closed polygon, the stream-surface that is swept out is now a closed surface that encloses a flow volume. These extensions to traditional streamlines have several advantages. Primarily, the user can visualize twist, convergence, and divergence of the field. However, they tend to clutter the volume and occlude one another, since they are now two-dimensional, rather than one-dimensional entities.

Kajiya [4] introduced a method for lighting lines in 1985, and Banks [1] generalized this work in 1994. Zöckeler *et al.* [13] extended this work by developing a faster method of shading lit lines and applied their method to specifically to streamlines. This extension provided a rapid lighting model for line segments, giving the user lighting cues to indicate line direction. In 1998, Löffelmann and Gröller [7] introduced threads of streamlets, which enable the visualization of the local field around a particular streamline. Advecting a thread of streamlets is equivalent to choosing one "base trajectory" fiber which is advected the full length of the propagation, and then having short streamlets constantly being advected, terminated, and re-seeded around that base trajectory fiber.

With all of the above methods, the user is investigating what happens to infinitesimally small particles as they advect through the field. However, these methods break down when the user is searching for more macroscopic characteristics of a field. For example, these methods have failed to illuminate the white-matter tracts in the brain [8], as well as cohesive currents in the ocean. Such methods fail specifically because each individual streamline only tracks an individual advection path, without being affected by the paths of the streamlines around it. As a result, all of the streamlines splay in various directions. Such divergence gives no indication of the average or primary flow direction. As a result, though there is a clear (though complex) pathway between the language center of the brain and the primary auditory cortex, traditional streamline advection methods have failed to illuminate this fiber tract. Similarly, charting large-scale currents through the worlds oceans has remained an elusive goal for automatic streamline advection methods.

In Figure 1, in the left image we see a group of streamlines advected from the corpus callosum. As the main communication pathway between the left and right hemispheres of the brain, the corpus callosum has primarily, though not solely, lateral connectivity. However, when we try to advect traditional streamlines through this region of the brain, only some of the streamlines indicate the gross bilateral direction of this neural structure. In general the streamlines diverge and do not present a cohesive picture of the corpus callosum. Advecting a thread of streamlets [7] for the right image in Figure 1, we can see a local neighborhood around one particular streamline (the so-called base trajectory, $\mathcal{T}$). But if that particular streamline does not follow the general path in which we are interested, as in this example, then it is of limited utility in revealing macroscopic structure. What we would like is a visualization method that tracks the *global structure* of the corpus callosum, despite the local complexity of the individual white matter fibers. The method we introduce here, termed *stream bundles* accomplishes this goal. In Figure 2, we see the result of

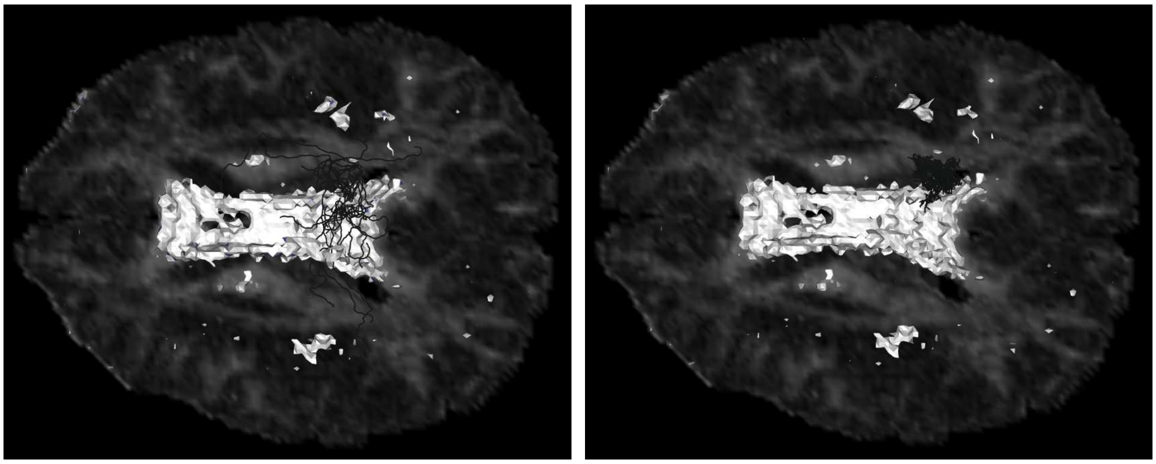

Figure 1: Visualization of advection from the corpus callosum. A slice of the dataset is shown underneath an isosurface indicating the corpus callosum. Streamlines are shown as dark line in the image on the left, and a thread of streamlets is shown as a dark mass in the image on the right. Though both methods locally track the white matter fiber directions through the region, neither method effectively reveals the macroscopic lateral directionality of the corpus callosum.

advecting a stream bundle through the corpus callosum. The primary lateral connectivity is clearly evident in this image, since the collection of streamlines has effectively been prevented from diverging.

Implementationally, a stream bundle is a collection of many short streamlines, which we call *fibers*. The advection of the stream bundle is determined by the collective advection of its fibers; and the re-seeding of the fibers is determined by the distribution of the bundle. As a result, the user sees the averaged advection path of some neighborhood through the volume. Intuitively, it is similar to advecting a macro-, rather than micro-particle through the volume. We believe this method is well suited to investigating grosser structures within vector fields, and in this paper we demonstrate its efficacy in visualizing oceanic currents.

The rest of this paper describes our method for advecting stream bundles, and how we control the distribution of the constituent fibers. In Section 2, we describe our methodology and implementation; in Section 3, we present results of applying stream bundles to an oceanic dataset; and in Section 4, we summarize and conclude with some ideas for future work.

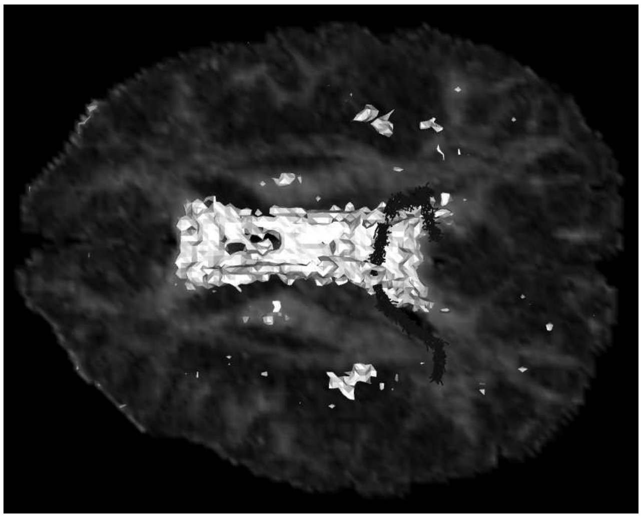

Figure 2: Stream bundles advected from the corpus callosum. The primary lateral flow direction of the structure is clearly revealed.

2 Methods and Implementation

As defined above, a stream bundle is a collection of many short, individual streamlines, which we call fibers. Running through the interior of the bundle is the bundle's *midline*. The midline is defined as the average positions of the bundle fibers at every step along the path. When we refer to the *direction* of the bundle, we are really referring to the direction of this midline. Furthermore, when we refer to the *shape* of the bundle at a particular step, we are really referring to the distribution of the bundle's constituent fibers at that step. In keeping the bundle together, the shape and direction are important descriptive terms; as such, we will be using these terms throughout the paper.

By definition, a stream bundle is a collection of individual fibers. The fibers of the stream bundle are independently advected through the field for a short distance before they terminate. When a fiber terminates, a new seed is placed within the bundle and a new fiber is advected. As defined in Equation 2, the advection method for the fibers is accomplished using a simple fourth-order Runge-Kutta algorithm [6]. To maintain the coherence of the fibers, they must be advected in lock-step. This is important because as fibers die off, they have to be re-seeded according to the distribution of the other fibers in the bundle. Having chosen an advection method, two questions remain: 1. how long should we advect an individual fiber? and, 2. where should we re-seed after that fiber has been terminated?

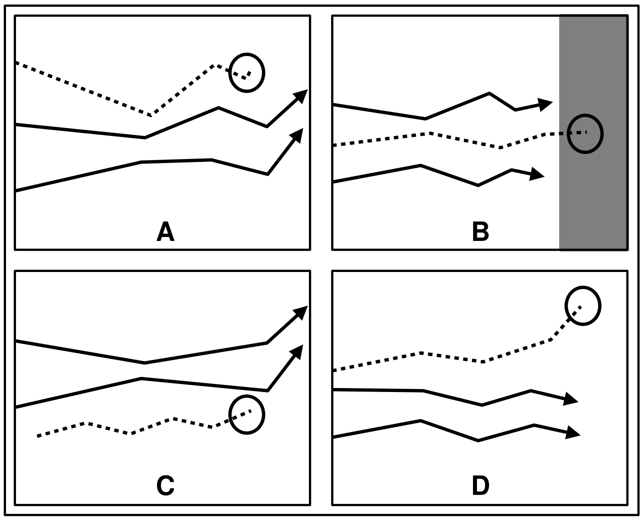

Figure 3: Termination policies for stream bundle fibers: A) stagnation, B) exiting field, C) random lifetime duration, D) divergent outlier.

It is important to note that the way in which the fibers stay together to produce a coherent bundle is *not* by influencing each other's paths as they all advect. Quite to the contrary, each fiber advects completely autonomously, its path determined by independent advection. Coherence is instead enforced via the bundle's re-seeding and termination policies. By pruning away divergent fibers, and then re-seeding new fibers near the bundle's midline, the algorithm controls the collective behavior of the bundle. As such, the termination and re-seeding policies are pivotal in dictating the direction and shape of the bundle. We discuss several such policies below.

2.1 Fiber Termination

We have investigated four different criteria for fiber termination, which we will now describe. The methods are pictorially represented for a two-dimensional vector field in Figures 3a-d, and extend naturally to three-dimensions.

There are two hard-coded termination criteria which are common to all of our implementations. The first is that once a fiber stagnates (*i.e.,* lands in a region of no flow), it is terminated. This case is indicated in Figure 3a. The second termination criterion is if a fiber leaves the bounds of the field, as indicated in Figure 3b. For the second case, the fiber is not restarted - allowing for bundle termination as the individual fibers exit the field.

In addition to these hard-coded methods, we investigated a "random lifetime" method

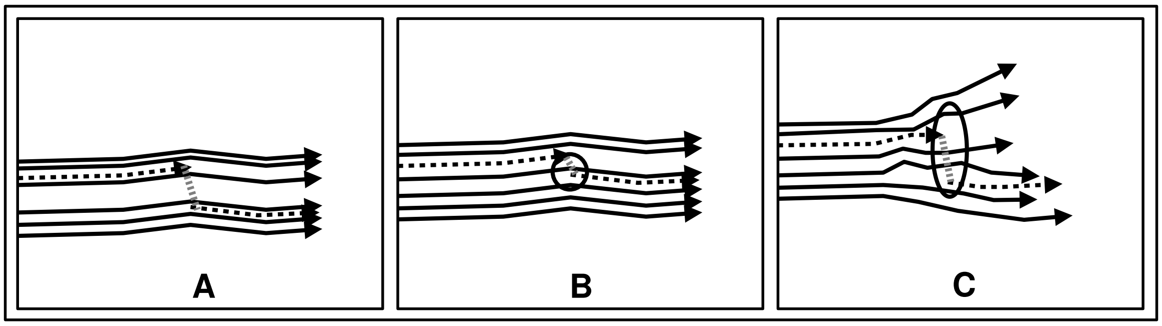

Figure 4: Re-seeding methods for stream bundle fibers: A) distribution-based, B) constant radius, C) evolving radius.

and an "outlier termination" method. In the first method, the fiber is given a lifetime (measured in advection steps) at the time it is created. The fiber advects this number of steps with the other fibers in the bundle and then promptly terminates. The length of the lifetime is randomly chosen based on a user-chosen maximum lifetime, lt_{max}, using the following equation: $lt = lt_{max} \times (1 - (rand())^2)$. This termination method is illustrated in Figure 3c, where the dotted fiber is terminated because it has reached its pre-determined lifetime of five steps.

For the "outlier termination" method, our algorithm determines if a fiber has strayed too far from the rest of the bundle, in which case it is immediately terminated and re-seeded. Fibers that stay close to the pack are allowed to continue to advect indefinitely (unless they stagnate or leave the field, in which case they are terminated as described earlier). This final termination case is shown in Figure 3d. The dotted fiber is terminated, as it strays too far from the center of the other fibers.

2.2 Fiber Re-seeding

In combination with the termination criteria, the re-seeding policy for the fibers will determine the shape and path of the bundle. We have investigated three methods for re-seeding. As with the previous termination figures, we have pictorially illustrated our re-seeding methods for the case of a two-dimensional vector field in Figures 4a-c.

The first method is based on re-seeding the new fiber to maintain the distribution of the other fibers in the bundle. Specifically, we find the midpoint of the present positions of the other fibers, and construct a histogram of the distances from the fibers to that midpoint. Outliers are then removed, and the histogram is Gaussian blurred. After normalization, this histogram is exactly the probability density function from which a radius for the new seed point can be chosen. A new seed point on the corresponding circle is then chosen at random, and a new fiber is advected from that seed point. An example of this re-seeding is shown for the dotted fiber in Figure 4a.

Our second re-seeding method is simply based on the midpoint of the existing fibers and a user-chosen radius. The user chooses a radius for the initial distribution sphere (or a set of three radii and axis directions for an ellipsoidal distribution). The midpoint of the fibers is computed as the bundle advects. In this way, it is always ready and available when a terminated fiber needs to be re-seeded. The seeding position within the sphere is chosen based on a uniform distribution within its volume. The advantage of this method is that the resulting bundle is easy to interpret, as it has a uniform shape throughout. This method is illustrated in Figure 4b. The dotted fiber has been re-seeded within the outlined elliptical region.

Our third method is a variant of the second method. Rather than maintaining a constant radius, though, the radius is allowed to vary as a function of the average velocity of the fibers. In regions of high velocity, the radius is made to shrink, thereby constricting the bundle. In regions of low velocity, the radius grows, permitting the slowly moving fibers to diffuse out. The motivation for this method is that in areas of high velocity, flow fields often hold together more tightly. In contrast, in regions of low flow there is often a more diffusive behavior. Allowing the radius to grow in these low velocity pools will allow us to expand our search, in hopes of finding another high velocity current out. An example of this re-seeding policy is indicated in Figure 4c.

2.3 Special Cases

We asserted earlier that the termination and re-seeding policies uniquely define a stream bundle algorithm. Furthermore, by choosing specific policies, we can mimic the behavior of other methods. For example, blurring the field as a pre-process, and then advecting traditional streamlines is equivalent to choosing a constant spherical distribution for our fibers, giving them varied, short lifetimes, and advecting the bundle.

Another special case of our algorithm is the “thread of streamlets” method of Löffelmann and Gröller [7]. The re-seeding policy for their method would require re-seeding within a sphere about a particular “base trajectory” fiber, instead of about the midline. Further, their termination policy is precisely our random lifetime method, with the caveat that the base-trajectory fiber never be terminated.

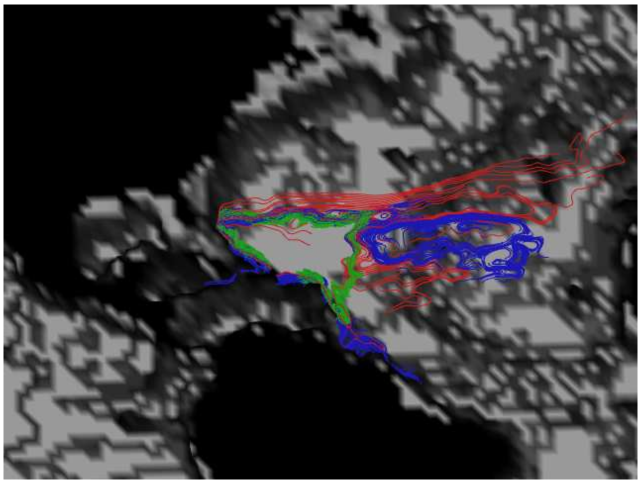

Figure 5: Streamlines, streamlets, and a stream bundle advected through the Gulf of Mexico. The streamlines and streamlets diverge through the region, whereas the stream bundle remains unfrayed. Please refer to the color plate for a clear differentiation of the lines.

3 Results

In this section, we present the results of applying our stream bundles algorithm to vector visualization in an oceanic dataset. The POP ocean dataset from the Advanced Computing Lab at Los Alamos National Lab represents the flow of oceanic currents within a 1024x768x42 volumetric model of the earth. This dataset is being studies in an effort to better chart pathways of major oceanic currents.

In Figure 5 we see a close-up of the Gulf of Mexico. A rake of streamlines is shown in red, a thread of streamlets is shown in blue, and a stream bundle is shown in green. All lines are rendered using Zöckeler's lit line model [13]. Whereas the streamlines and streamlets have divergent paths, the streambundle maintains a cohesive form as it evolves through the region.

4 Conclusions and Future Work

We have introduced a framework for cohesive streamline advection (stream bundles), which enable the user to control the coherence among a propagating collection of streamlines (fibers). Streambundles enable the visualization of coarser properties within complex vector fields, which would otherwise be less apparent, if visible at all.

Working from the mindset of cohesive advection, we believe this framework could be applied to other vector field advection methods, including stream-surfaces, stream-ribbons and stream-volumes. Additionally, we are interested in exploring other termination and re-seeding methods.

5 Acknowledgments

This work was supported in part by awards from the Department of Energy and the National Science Foundation. The authors would like to thank Chris Johnson, Chuck Hansen and Matthew Bane for their valuable comments and suggestions. We also gratefully acknowledge Los Alamos National Lab for providing the oceanic dataset.

References

[1] D.C. Banks. Illumination in diverse codimensions. In *SIGGRAPH 94 Proceedings*, volume 28, pages 327–334, 1994.

[2] M. Brill, H. Hagen, H-C. Rodrian, W. Djatschin, and S.V. Klimenko. Streamball techniques for flow visualization. In *IEEE Visualization 94 Proceedings*, pages 225–231, 1994.

[3] J.P. Hultquist. Constructing stream surfaces in steady 3-d vector fields. In *IEEE Visualization 92*, pages 171–178, 1992.

[4] J.T. Kajiya. Anisotropic reflection models. In *SIGGRAPH 85 Proceedings*, volume 19, pages 15–21, 1985.

[5] D.N. Kenwright and G.D. Mallinson. A 3-d streamline tracking algorithm using dual stream functions. In *IEEE Visualization 92*, pages 62–68, 1992.

[6] D. Kincaid and W. Cheney. *Numerical Analysis*. Brooks/Cole Publishing Company, 1991.

[7] H. Löffelman and E. Gröller. Enhancing the visualization of characteristic structures in dynamical systems. In *EUROGRAPHICS 98 Proceedings*, pages 35–46, 1998.

[8] N. Makris, A.J. Worth, G. Sorensen, G.M. Papadimitriou, O. Wu, T.G. Reese, V.J. Wedeen, T.L. Davis, J.W. Stakes, V.S. Caviness, E. Kaplan, B.R. Rosen,

D.N. Pandya, and D.N. Kennedy. Morphometry of *in vivo* human white matter association pathways with diffusion weighted mri. *Annals of Neurology*, 42(6):951–962, 1997.

[9] N. Max, B. Becker, and R. Crawfis. Flow volumes for interactive vector field visualization. In *IEEE Visualization 93 Proceedings*, pages 19–24, 1993.

[10] W.J. Schroeder, C.R. Volpe, and W.E. Lorensen. The stream polygon: A technique for 3d vector field visualization. In *IEEE Visualization 91 Proceedings*, pages 126–132, 1991.

[11] J. Vennard and R. Street. *Elementry Fluid Mechanics*. John Wiley and Sons, Inc., 1975.

[12] G. Volpe. Streamlines and streamribbons in aerodynamics. *AIAA 27th Aerospace Science Meeting*, 1989.

[13] M. Zöckeler, D. Stalling, and H-C. Hege. Interactive visualization of 3d-vector fields using illuminated streamlines. In *IEEE Visualization 96 Proceedings*, pages 107–113, 1996.