Topology-Aware Evenly-Spaced Streamline Placement

Keqin Wu, Zhanping Liu, Song Zhang, *Member, IEEE*, Robert J. Moorhead II, *Senior Member, IEEE*

Abstract— This paper presents a new streamline placement algorithm that produces evenly-spaced long streamlines while preserving topological features of a flow field. Singularities and separatrices are extracted to decompose the flow field into topological regions. In each region, a seeding path is selected from a set of streamlines integrated in the orthogonal flow field. The uniform sample points on this path are then used as seeds to generate streamlines in the original flow field. Additional seeds are placed where a large gap between adjacent streamlines occurs. The number of short streamlines is significantly reduced as evenly-spaced long streamlines spawned along the seeding paths can fill the topological regions very well. Several metrics for evaluating streamline placement quality are discussed and applied to our method as well as some other approaches. Compared to previous work in uniform streamline placement, our method is more effective in creating evenly-spaced long streamlines and preserving topological features. It has the potential to provide both intuitive perception of important flow characteristics and detail reconstruction across visually pleasing streamlines.

Index Terms— Evenly-spaced streamlines, flow topology, flow visualization, seeding strategy, streamline placement

1 Introduction

Streamlines, as opposed to texture-based techniques, provide a sparse representation of the underlying flow. The effectiveness is highly dependent on the placement of integral curves that involves seed selection and density control. There are several criteria [1], [2], [3], [4] for placing streamlines: (1) feature preservation — important flow structures around singularities need to be retained, (2) uniform placement — evenly-spaced streamlines may help with visual interpolation between them, (3) continuous representation — discontinuities may cause distractions, and (4) reconstruction fidelity — minimum error is desired when reconstructing a flow field from the streamline representation [5].

Topology characterizes a flow in that the relatively uniform flow behavior in each topological region can be deduced from its boundary [6], [7], [8]. Thus topology preservation in an evenly-spaced streamline placement facilitates mental reconstruction of the flow. This issue is not sufficiently addressed in streamline placement algorithms [1], [3], [4], [9], [10] except for the efforts of Verma et al. [2] and Chen et al. [11]. However, Verma et al.'s method emphasizes flow topology at the cost of placement uniformity. Chen et al.'s work combines topology preservation with Jobard and Lefer's streamline placement algorithm [10]. We include a comparison to Chen et al.'s method in section 5.

This paper presents a topology-preserving and evenly-spaced streamline placement algorithm. This method exploits flow topology and orthogonal flow for seed selection. A flow field is decomposed into topological regions. In each region the longest orthogonal flow path originating from the boundary is chosen as a seeding path.

The remainder of this paper is organized as follows. Section 2 reviews previous streamline placement algorithms. In section 3 four criteria used to measure streamline layout quality are discussed. Section 4 describes our algorithm in detail. Results and comparison to previous methods are given in section 5 to demonstrate the merits of the proposed method. Section 6 concludes this paper with future plans.

2 Related Work

Turk and Banks [1] presented an image-guided streamline placement algorithm that iteratively refines an initial layout by editing streamlines to minimize an energy function in order to obtain an optimal output. Later, Mao et al. [12] extended it for streamline placement on curved surfaces.

There are several sample-based algorithms for placing evenly-spaced streamlines. Jobard and Lefer [10], [13], [14] proposed a neighborhood seeding strategy to steer streamline placement. Each sample of an accepted streamline provides a candidate seed on each side in the direction orthogonal to the flow. Liu et al. [3], [4] enhanced this method by designing a double-queue seeding mechanism to prioritize both topologically distributed seeds and long streamlines. In addition, they presented a rapid and robust loop detection strategy to prevent a layout from being cluttered by tightly spiraling streamlines [3]. Chen et al. adapted Jobard and Lefer's method [10] by adding topological separatrices as initial streamlines to the placement [11]. Mebarki et al. developed a global seeding scheme in which Delaunay

- *Keqin Wu and Robert J. Moorhead II are with the GeoSystems Research Institute, Mississippi State University, PO Box 9627, Miss. State, MS 39762. E-Mail: {keqin, rjm} @gri.msstate.edu.*
- *Zhanping Liu is with Kitware, Inc., 28 Corporate Drive, Suite 204, Clifton Park, NY 12065. E-Mail:zhanpingliu@hotmail.com.*
- *Song Zhang is with the Department of Computer Science and Engineering, Mississippi State University, PO Box 9637, Miss. State, MS 39762. E-Mail: szhang@cse.msstate.edu.*

triangulation of the streamline samples of the current placement is performed to place a seed at the center of the biggest cavity [9]. Li and Shen presented a framework for view-dependent placement and rendering of 3D streamlines [15]. Recently Liu and Moorhead proposed an interactive view-driven evenly-spaced streamline placement algorithm [16]. These sample-based algorithms mainly differ from one another in the greedy seeding strategy that drives the placement process until a dynamic queue of candidate seeds is empty. However, the separatrices of a flow are seldom shown in the resulting placement and therefore more mental reconstruction may be needed to understand the flow topology.

Verma et al. presented a flow-guided seeding strategy in which topological templates are designed to put seeds around singularities and then Poisson-disk distribution is used to place additional seeds elsewhere [2]. This work was later extended to 3D flows [17]. One advantage of this method is that topological features of the flow can be retained in the vicinity of singularities even when the global placement density is low. However, the topological templates only exhibit the flow patterns near the singularities, and the Voronoi diagram used to adjust the template size does not reflect the actual flow patterns sufficiently, leaving a large number of streamlines placed unevenly outside the templates. Thus the overall uniformity is compromised as the flow complexity and the distance between singularities increase. Chen et al. developed a similarity metric that implicitly accentuates regions of geometric interest to control the density of streamlines [5]. They also employed an error metric to evaluate placement quality by comparing the reconstructed flow against the original flow. Li et al. proposed a novel streamline selection scheme based on both local and global dissimilarity between streamlines to describe the underlying flow pattern [18]. However, uniform placement is not the goal of these two algorithms. Our algorithm aims to produce an aesthetic evenly-spaced streamline placement while capturing topology information such as singularities and separatrices. Our contribution includes a new seeding strategy, an enumeration of some streamline placement criteria, and a comparison between streamline placement algorithms.

3 Streamline Placement Quality

This section discusses several methods for evaluating streamline layout quality in terms of placement uniformity, streamline length, topological feature preservation, and flow reconstruction fidelity. In Section 5, these methods are used to test the performance of our algorithm.

3.1 Evenly-Spaced Placement

Evenly-spaced placement refers to the uniform density of streamlines [9]. This task usually involves two components: one is to select some locations to place seeds and the other is to govern the integration of each streamline by checking inter-streamline distance against a specified threshold [3], [4], [9], [10].

A quantitative evaluation of uniform density is used in the image-guided streamline placement method [1]. A low-pass filter is adopted to blur the binary-valued image of streamline representation. The resulting image is then compared against a uniform gray-scale image to drive an iterative layout refinement process.

3.2 Long Streamline Placement

Another important criterion is streamline length. Long streamlines are visually pleasing and may suppress discontinuities resulting from short streamlines [2].

Discontinuities may be categorized into those near singularities and those in laminar areas [9]. The former case may be unavoidable as streamlines approach one another near sinks or sources before being terminated. The latter case can be addressed by using an improved seeding strategy.

Since every streamline falls into a certain topological region [6], [7], [8], each case of discontinuity can be exemplified by analyzing the subset of streamlines in a single topological region. Fig. 1 illustrates the discontinuous streamline placement and the desired long streamline placement. As Fig. 1 (left) indicates, a short streamline occurs when (1) two neighboring long streamlines leave a space that is neither small enough to maintain a specified density nor large enough to allow for a longer streamline (shown in the blue square) and (2) streamlines fail to reach the vicinity of a source or sink (shown in the green square). A desirable streamline placement needs to effectively handle these two scenarios to prevent short streamlines. A better solution is given in Fig. 1 (right) where each streamline starts near the source and approaches the sink without breaking in the middle.

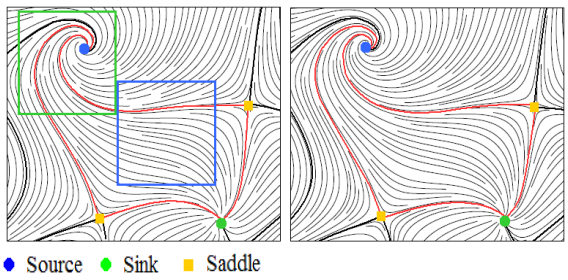


Fig. 1. Continuity of streamlines within a simple topological region. A topological region is indicated by its red boundary. **Left:** Discontinuities (in the blue rectangle) exist between the source and the sink as well as near the focus (in the green rectangle). **Right:** Discontinuities occur only in the vicinity of singularities.

3.3 Topological Feature Preservation

Singularities and separatrices provide a concise depiction of salient flow patterns. Fig. 2 compares two streamline placement strategies in preserving topological features. As shown in Fig. 2 (right), the more accurate representation of the flow field provides a detailed spiraling pattern around the focus and a clear segmentation around the saddle.

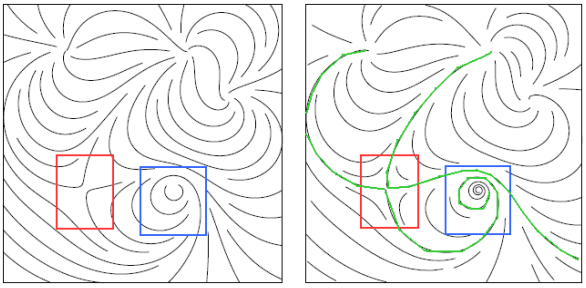


Fig. 2. Topological feature preservation through streamline placement. **Left:** Streamlines under-represent the topological features, i.e., the saddle in the red square and the focus in the blue square. **Right:** Streamlines effectively represent the topological features constituted by the saddle, the focus, and separatrices (green curves). Figure reproduced from [2].

3.4 Flow Reconstruction Fidelity

The error measurement in [5] offers a way to evaluate how well the flow field as a whole is preserved by a streamline placement. The error function is defined as follows.

Given a vector field V with n grid points, $v_0, v_1, ..., v_{n-1}$, let S denote a set of sampled streamlines. A vector field V' is reconstructed through Delaunay triangulation of these samples in S and n vectors, $v'_0, v'_1, ..., v'_{n-1}$, are re-sampled at the original grid points. Then the flow reconstruction error is defined by the average residual magnitude:

$$E(S) = \frac{1}{n}\sum_{i=0}^{n-1}\left\| v_i - v_i' \right\|$$

4 Topology-Aware Evenly-Spaced Streamline Placement

The main idea of our method is to extract the flow topology that divides the flow field into regions of uniform flow behavior. An orthogonal curve crossing all streamlines within each region is then chosen to guide streamline placement.

Our algorithm involves three main steps: (1) separating the flow field into regions by extracting the topological skeleton, (2) searching for a seeding path that orthogonally crosses all streamlines within each region, and (3) placing seeds evenly along each seeding path to integrate streamlines from the seeds.

The details are presented in the remainder of this section. First, the concept of generalized topology [19] is briefly reviewed and the existence of the paths crossing all streamlines in each topological region is discussed in section 4.1. Then, the selection of the longest seeding path and the distance control over streamline density are described in sections 4.2 and 4.3, respectively. Section 4.4 gives the framework of this algorithm.

4.1 Generalized Topology and Region-Crossing Paths

The topology of a 2D linear vector field V consists of all singularities (sinks, sources, saddles, and centers), periodic orbits, and separatrices of V [8]. The flow field is segmented by the separatrices into a set of topological regions so that all streamlines (except for periodic orbits) within each region share one source and one sink [8]. Fig. 3 shows a typical topological region whose boundary is connected with one sink, one source, and two saddles.

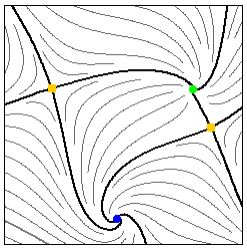


Fig. 3. The topology of a 2D flow field. There are four singularities, i.e., one source (blue point), one sink (green point), and two saddles (yellow squares). Separatrices (dark curves) connect the saddles with the sources and the sinks.

In general, separatrices in a 2D flow are streamlines that either start from or end at saddles. They can be computed by integrating streamlines from the saddles. However, for any bounded flow field, it is difficult to determine regions defined by separatrices that connect to saddles outside the domain. To deal with this issue, Scheuermann et al. proposed the local vector field topology that supports additional topological segmentation by identifying boundary saddles that exhibit inflow/outflow switches along the flow field boundary [19]. The boundary of a flow field is therefore segmented by boundary saddles into inflow boundaries and outflow boundaries that serve locally as sources where the flow is directed inwards and sinks where the flow is directed outwards [8]. The generalized notion of separatrices then includes streamlines connecting boundary saddles. Fig. 4 compares global topology and local topology. More detailed information about the generalized topology can be found in [8], [19].

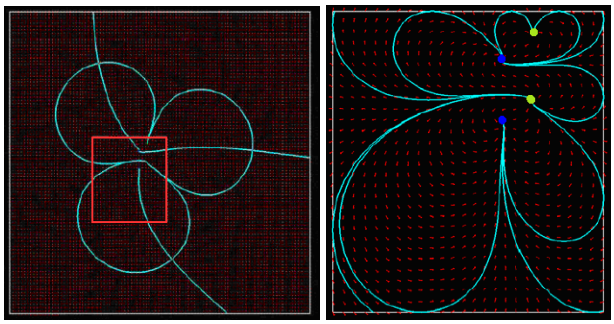


Fig. 4. Global topology (left) and local topology (right) (Scheuermann et al. [19]). Global topology is derived by considering the entire field. Local topology reveals the separatrices connecting boundary saddles with sources (blue points) and sinks (green points). The red square in the left figure indicates the region shown in the right figure.

With separatrices integrated from boundary and ordinary saddles, those streamlines that cannot be sorted into a subset but share the same source or sink are then classified into a new type of region, in which all streamlines share the same inflow or outflow boundary.

Both inflow boundary and source can be considered *entrances* where flow enters the region, and both outflow boundary and sink can be considered *exits* where flow leaves the region. For every region, a path crossing all the streamlines can be obtained between the edges of the region. Fig. 5 shows topology without boundary saddles and local topology with boundary saddles. Fig. 6 shows a set of region-crossing paths of a flow field.

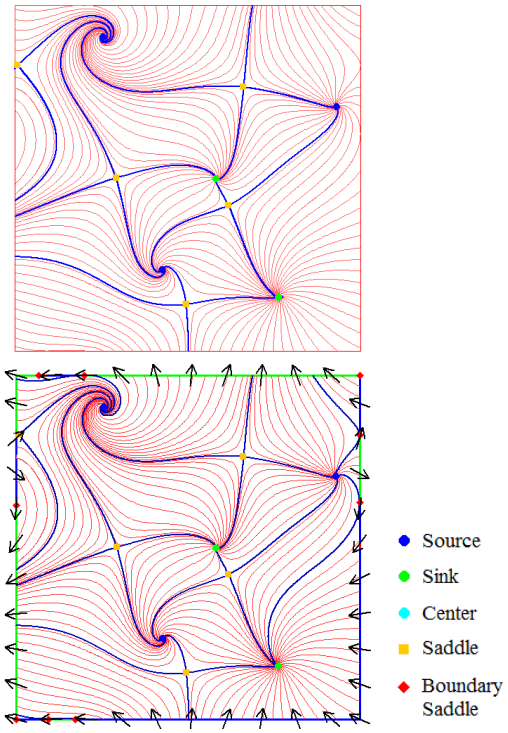


Fig. 5. The topology without boundary saddles (Top) and the local topology with boundary saddles (Bottom). The blue curves denote the separatrices. The black arrows indicate flow directions along the boundary.

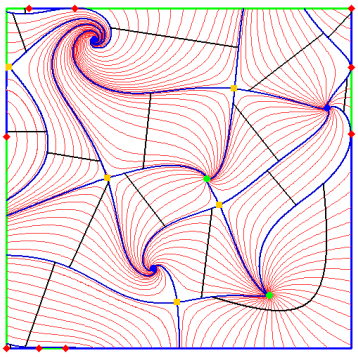


Fig. 6. A set of region-crossing paths (black lines) of a flow field.

4.2 The Longest Seeding Path

The topological segmentation method ensures that there is no more than one flow entrance (either inflow boundary or source) and no more than one exit (either outflow boundary or sink) along the boundary of a topological region. The rest of the boundary is then separated into two *edges*. An orthogonal flow field $V_\perp$ can be obtained by rotating all the vectors of a flow V 90^0 clockwise or counter-clockwise. The streamlines of $V_\perp$ are called *orthogonal curves* of the original flow V. Any orthogonal curve intersecting the two edges of a region crosses all the in-region streamlines of the original flow.

The desired seeding path is the one that can be used to generate evenly spaced streamlines. In general, a longer path often results in a more uniform placement than a shorter one. Given a fixed number of streamlines that are crossed, the longer the path, the larger the average distance between its intersections with streamlines. Seeding across large cavities may provide better control over the overall density of streamlines [9], [10]. In fact, seeding paths obtained from the orthogonal flow field are more effective than other choices in creating an elegant placement. Fig. 7 compares the streamlines spawned along three kinds of paths, i.e., a long straight path, a short orthogonal path, and the longest orthogonal path. The seeds are placed equidistantly along the three paths. We can visually verify that the longest orthogonal path produces the most evenly-placed streamlines. Fig. 8 shows the orthogonal curves crossing all the topological regions of a flow field.

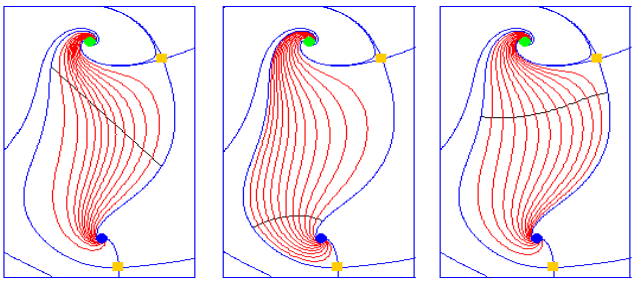


Fig. 7. Three comparative paths crossing a topological region. The topological domain is filled with the red streamlines spawned equidistantly along a long straight line (left), a short orthogonal curve (middle), and the longest orthogonal curve (right), respectively.

Orthogonal curves are obtained by integrating streamlines in the orthogonal flow field. The process of finding the longest curve is straightforward: first, we begin with the longer one of the two edges, sample it by interval D; integrate an orthogonal curve from each sample point toward the other edge through the inner region; and finally choose the longest curve.

The regions within periodic orbits or saddle-connected loops are addressed as special cases. Periodic orbits can be detected by using Wischgoll and Scheuermann's method [20] or Chen et al. methods [11]. A saddle-connected loop, e.g., a homoclinic loop, consists of one or more saddles and the separatrices connecting them. It can be identified by checking whether the boundary separatrices of a topological region are connected with only saddles. In both cases, the longest orthogonal curves that start from either the periodic cycles or saddle-connected loops toward the singularities and other closed streamlines within them (cycles and/or loops) are selected as the seeding paths. Fig. 9 shows the seeding paths (black curves) found for both cases. One flow field (left) captures the eye of Hurricane Lili and

the other (right) consists of three vortices [2]. The path selection scheme for any other kinds of topological regions is the same — finding the longest path between the edges of a topological region.

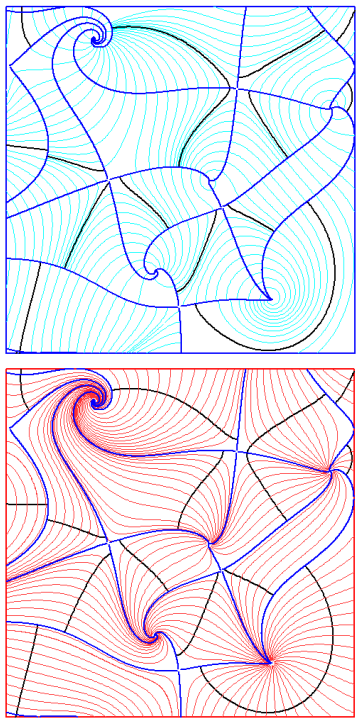


Fig. 8. The seeding paths. **Top:** The longest paths (black) are determined from the orthogonal curves (light blue) crossing the topological regions (divided by blue separatrices). **Bottom:** Streamlines (red) are spawned along the seeding paths (black).

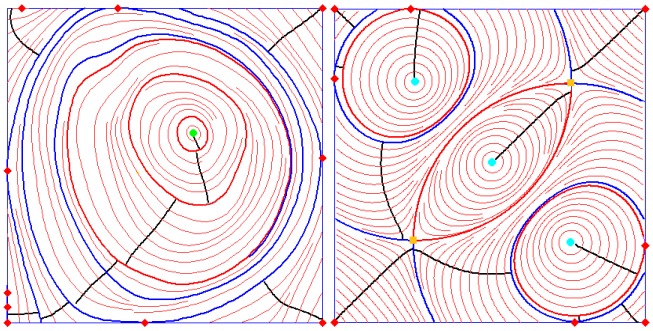


Fig. 9. The selection of seeding paths in two flows with closed orbits. **Left:** a flow field that contains three periodic orbits (bold red curve). **Right:** a flow field with three saddle-connected loops (bold red curves) which each contain a set of center orbits. The black curves represent the seeding paths.

4.3 Inter-streamline Distance Control

Once a new streamline is integrated from a seed on a seeding path, it usually approaches the adjacent streamlines as it goes towards the entrance or exit of the region. When two adjacent streamlines diverge, cavities emerge between them. Thus inter-streamline distance control is needed.

(1) Inter-streamline distance

Inter-streamline distance can be approximated by inter-sample distance [3]. Let point p be a sample point of streamline S_1, the inter-streamline distance from p to S_1's closest neighbor S_2 is the shortest distance between p and the sample points of S_2, i.e. $d(p,S_1,S_2)=\min(|\overline{pq_j}|)$ where q_j represents the sample point on S_2. However, this measurement is not symmetric between the two neighboring streamlines. For example, in Fig. 10, $d(p,S_1,S_2)=|\overline{pq_1}|$, whereas $d(q_1,S_2,S_1)=|\overline{q_1p'}|$, and $d(p,S_1,S_2)\neq d(q_1,S_2,S_1)$. Thus we propose the use of *orthogonal distance* to govern inter-streamline distance. The orthogonal distance from p to S_2, $d_\perp(p,S_1,S_2)$, is defined as the arc length of the orthogonal curve OS that starts from p and ends at the intersection point of S_2 and OS, q_2. The length of the orthogonal curve segment $|\overline{pq_2}|$ can be approximated by the sum of the distance between the sample points along the curve between p and q_2. As Fig. 10 shows, $d_\perp(p,S_1,S_2)=d_\perp(q_2,S_2,S_1)=|\overline{pq_2}|$. The seeding interval along the longest orthogonal path is measured by orthogonal distance.

(2) Orthogonal distance control

A streamline is uniformly sampled as it is integrated. The distance from each newly generated sample point along this streamline to an adjacent streamline is computed as the length of the orthogonal curve starting from that sample to the adjacent streamline, i.e., the orthogonal distance. Let D be the interval between two adjacent seeds along the seeding path, $D_s=aD \quad (0<a<1)$ the threshold for minimum distance control, and $D_l=bD \quad (b\geq 2)$ the threshold for maximum distance control. Once the inter-streamline distance is less than D_s, streamline integration is terminated. If the distance is greater than D_l, additional sample points are generated on the orthogonal curve by interval D and new streamlines are placed at the sample positions. Results indicate that this orthogonal distance metric is particularly well suited for low-density uniform placement. For dense representation, it could be replaced by a faster approximation by the inter-sample distance scheme.

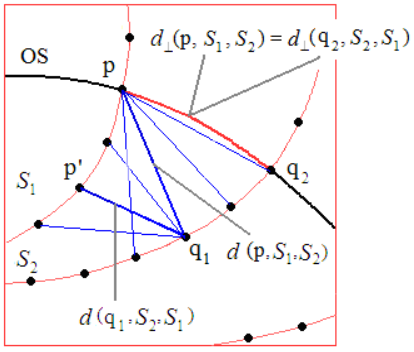


Fig. 10. The inter-streamline distance metric. The bold red curve indicates the orthogonal distance between S_1 and S_2. The two bold blue straight lines indicate the minimal distance from p to the sample points of streamline S_2 and the minimal distance from point q_1 to the sample points of streamline S_1 respectively.

4.4 The Pipeline

Suppose $TR_i (i = 1,...,n)$ are the topological regions of a flow field V, and $Path_i$ is the longest orthogonal path for TR_i. The pipeline of our algorithm is described as follows.

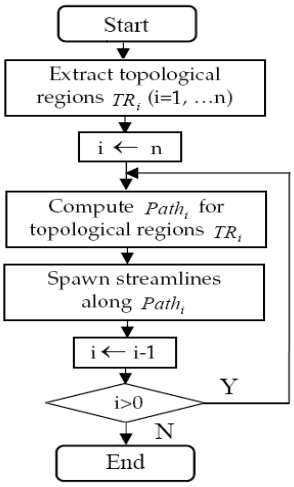


Fig. 11. The pipeline of our algorithm.

(1) Extract the topological regions of V:

1-1 Obtain singularities, boundary saddles, and periodic orbits;

1-2 Integrate separatrices from the saddles and boundary saddles;

1-3 Create a topology graph with separatrices, singularities, boundary saddles, and periodic orbits;

1-4 Extract topological regions based on the topology graph.

(2) Find $Path_i$ in each region TR_i:

Let $Edge_1$ and $Edge_2$ be the edges of TR_i and $Edge_1 > Edge_2$.

2-1 Compute a set of orthogonal curves evenly spaced by interval D along $Edge_1$;

2-2 Select the longest orthogonal curve as $Path_i$.

(3) Place seeds along $Path_i$ to integrate streamlines:

Let $Q = \{seed_j \mid j = 1,...,m\}$ be the queue of candidate seeds and S_j the streamlines placed within TR_i. S_j is uniformly sampled by interval D as it is integrated.

$Q \leftarrow$evenly-spaced (by D) samples of $Path_i$
$S_0 \leftarrow Edge_1$, $j \leftarrow 1$;
While $Q \neq []$
{
  $seed_j \leftarrow Q$.pop();
  For *direction* = (forward, backward)
  {
    Current sample point $P_c \leftarrow seed_j$
    Do
    {
      1. If ($P_c \in Edge_2$)
           get the next sample point P_n
           on the $Edge_2$ in the current *direction*;
         else
           Integrate S_j from P_c in the current *direction* to get the next sample point P_n on S_j;
         $P_c \leftarrow P_n$;
      2. Compute the orthogonal curve OS from P_c toward the closest streamline among $S_{j-1},...,S_0$;
         orthogonal distance $d \leftarrow$ the length of OS;
      3. Check d:
         If $d > D_l$
         {
           Sample OS by interval D;
           Q.push (samples);
         }
    } While (($P_c \in Edge_2$ and P_c is not the end point of $Edge_2$) or ($P_c \notin Edge_2$ and $d > D_s$ and P_c is within the flow boundary))
  }
  $j \leftarrow j+1$;
}
$S_j \leftarrow Edge_2$;
Render $S_n (n = 0,...j)$;

5 Results and Discussions

Our topology-aware streamline placement algorithm is applied to nine flow datasets. The distance thresholds used in our algorithm are D_s=0.5D and D_l=2D, where D is a percentage of the field width.

Fig. 12 demonstrates the streamline placements for flow 1 with a simple topology, flows 2 and 3 with periodic orbits, and flow 4 with waving flow patterns around eight vertically symmetric centers and foci. Fig. 13 shows two slices from a simulation of Hurricane Lili's wind field. The hurricane eye is clearly visible through the representation of evenly-spaced streamlines. The separatrices present in the images allow for accurate investigation of the interaction between the eye and the surrounding air flow. Fig. 14 displays a complex flow field obtained from a US Navy model of the Northeast Pacific Ocean. As Figures 12, 13, and 14 indicate, our algorithm is able to retain the topological skeleton in a considerably uniform streamline placement.

The comparison of our algorithm to several previous methods in terms of evenly-spaced placement, continuous representation, feature preservation, reconstruction fidelity, and time complexity follows.

Evenly-Spaced Placement Our method is capable of creating uniform streamline placements. The relatively high density near a singularity is due to the underlying flow characteristics. In fact, nearly all sample-based algorithms suffer from this inherent problem. Thus the comparison between different algorithms in placement quality is focused on laminar areas. The use of the longest orthogonal path for streamline seeding reduces discontinuities in each topological region and hence helps with the placement of evenly-spaced streamlines. As shown in Fig. 15, our approach is more effective than Verma et al.'s flow-guided method [2] in maintaining

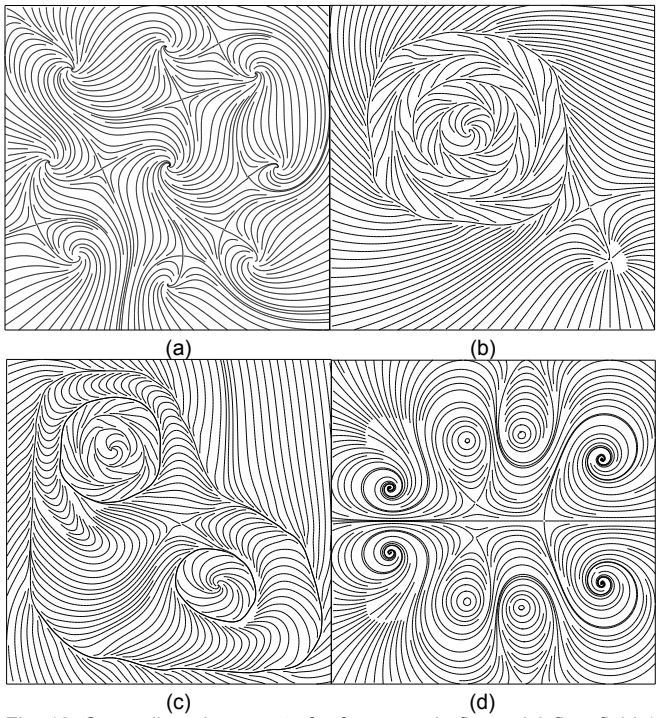


Fig. 12. Streamline placements for four sample flows: **(a)** flow field 1 with a simple topology, **(b)** flow field 2 with four periodic orbits, **(c)** flow field 3 with five periodic orbits, and **(d)** flow field 4 with waving flow patterns around eight vertically symmetric centers and foci.

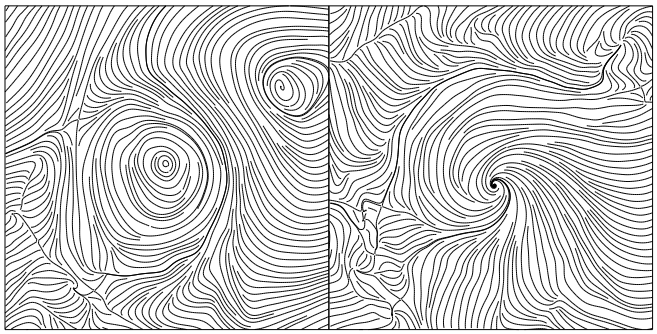


Fig. 13. Streamline placements for flow fields 5 and 6 (two slices of Hurricane Lili).

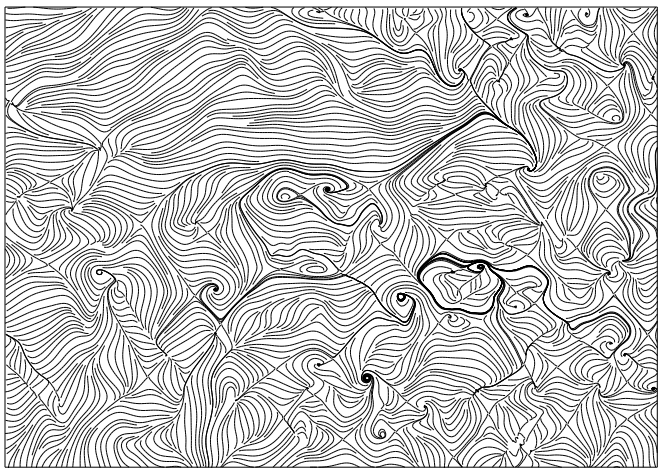


Fig. 14. Streamline placement for flow field 7 (an ocean flow field 149.5° W ~112.4° W, 35.0° N ~ 49.9° N).

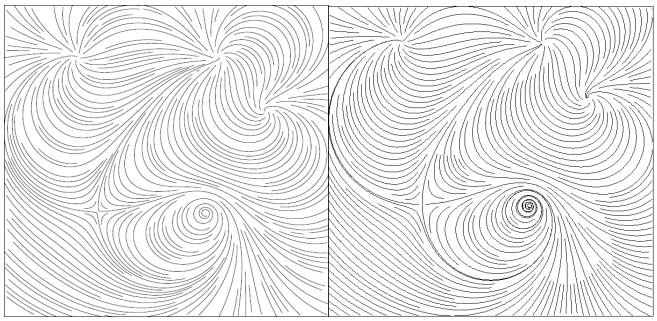


Fig. 15. Comparison of Verma's method [2] (left) with our method (right) in streamline placement for flow field 8.

overall placement uniformity, given the same streamline density input. As these algorithms differ in distance metric and inter-streamline density control, they exhibit differences in the number of the streamlines of a placement (Tables 1 and 2). Our algorithm tends to produce slightly denser streamlines than other methods because it adopts orthogonal distance that is measured as the length of a curved orthogonal streamlet instead of the inter-sample distance. In addition, it always seeds in sparse areas first in a way to help streamlines populate the flow field in a uniform way.

The quantitative comparison in uniformity is conducted by applying a Gaussian filter to the streamline placement image. The mean squared error (MSE) between the gray-scale image and the resulting image reveals how evenly-spaced the streamlines are. A smaller MSE indicates more uniform density. The filtered version of the streamline placement images of flow field 9 is given in Fig. 16. The related statistics of our algorithm, Jobard and Lefer's algorithm [10], Mebarki et al.'s algorithm [9], Liu et al.'s algorithm [3], and Chen et al.'s algorithm [11] are provided in Table 1 (for flow 1) and Table 2 (for flow 9). As shown by the average result in Fig. 17, our method produces a slightly higher gray-scale MSE than Liu et al.'s approach, possibly due to some degree of cluttering caused by the use of separatrices in our algorithm, but a lower gray-scale MSE than the other three algorithms.

Continuous Representation Our method exhibits the potential to generate long streamlines. There are two major sources for the occurrence of short streamlines (section 3.2). As streamlines are seeded along the longest orthogonal curve in each topological region, the sparsest area is filled first and then short streamlines are less likely to show up there. Thus the first cause — large cavities between neighboring long streamlines — is effectively addressed. The longest seeding path ensures that the streamlines are evenly-spaced in the relatively sparse region. The streamlines originating from this region can approach sinks or sources very well under streamline density control. Thus the second cause — the inability of streamlines to reach the vicinity of singularities — is considerably mitigated and the resulting placement can reveal salient flow structures clearly.

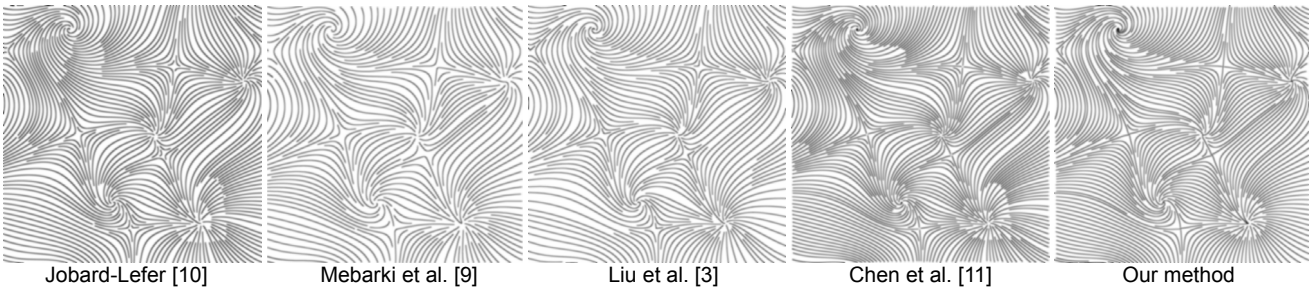


Fig. 16. Comparison of our method with Jobard and Lefer's method [10], Mebarki et al.'s method [9], Liu et al.'s method [3], and Chen et al.'s method [11] in the blurred image of streamline placement for flow field 9.

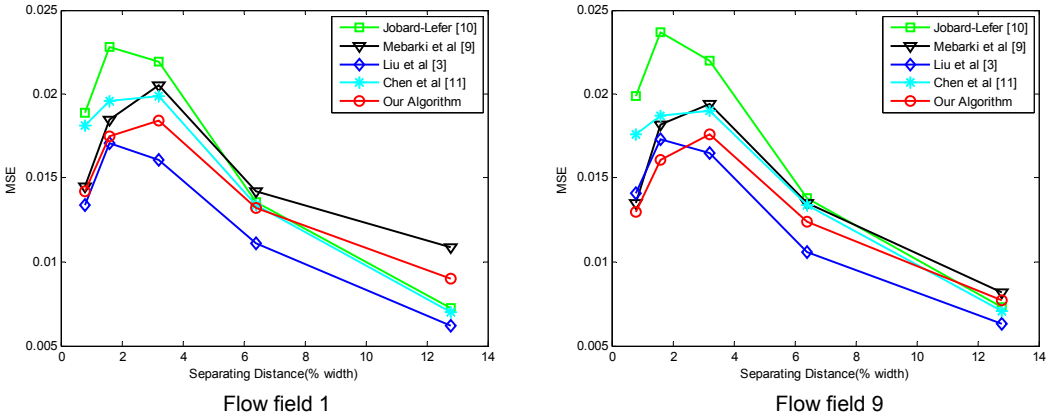


Fig. 17. Quantitative comparison of our method with Jobard and Lefer's method [10], Mebarki et al.'s method [9], Liu et al.'s method [3], and Chen et al.'s method [11] in gray-level MSE versus separating distance based on Tables 1 and 2.

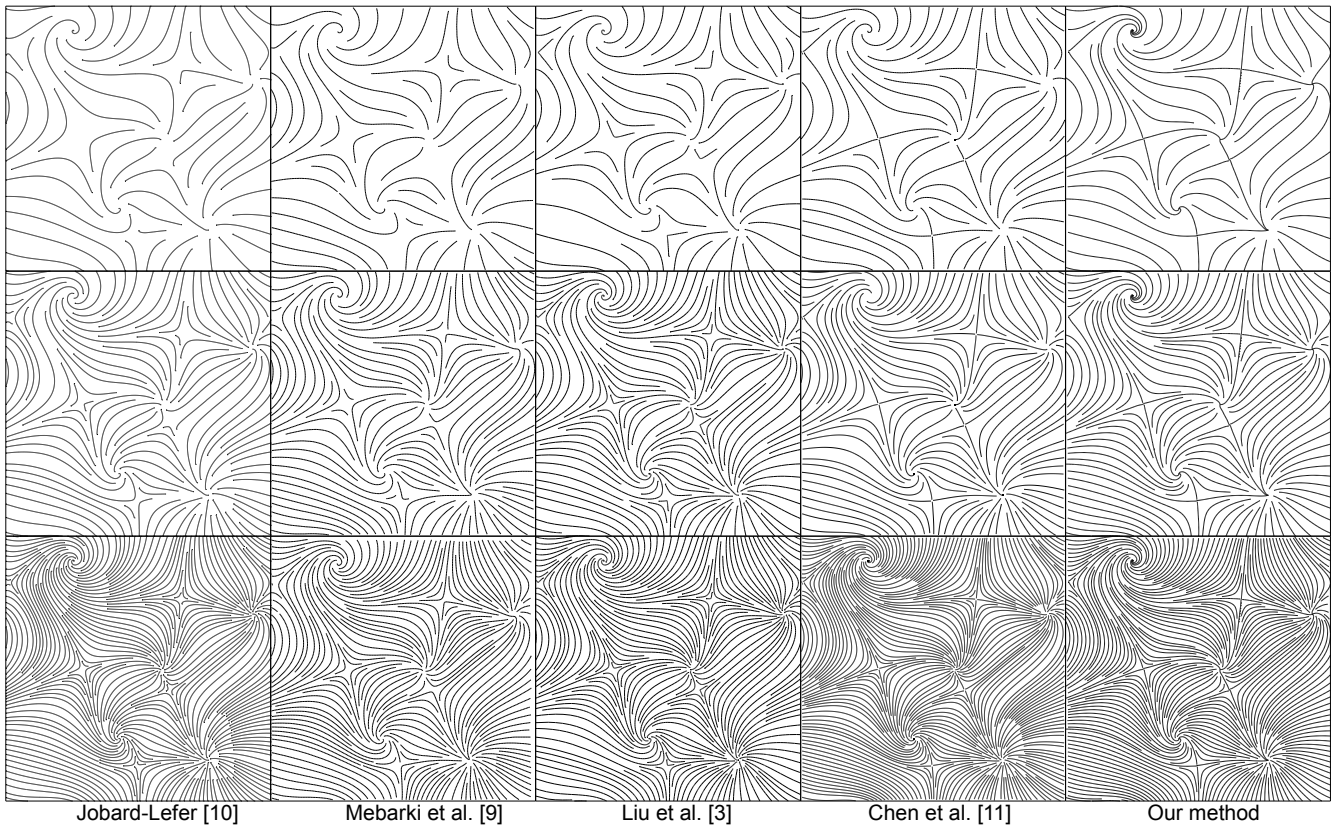


Fig.18. Comparison in the streamline placement of flow field 9. From left to right are streamline placements generated by using Jobard-Lefer [10], Mebarki et al[9], Liu et al.[3], Chen et al.[11], and our algorithm for three increasing densities, top to bottom, 3.2%, 1.6%, 0.8% of the flow field width.

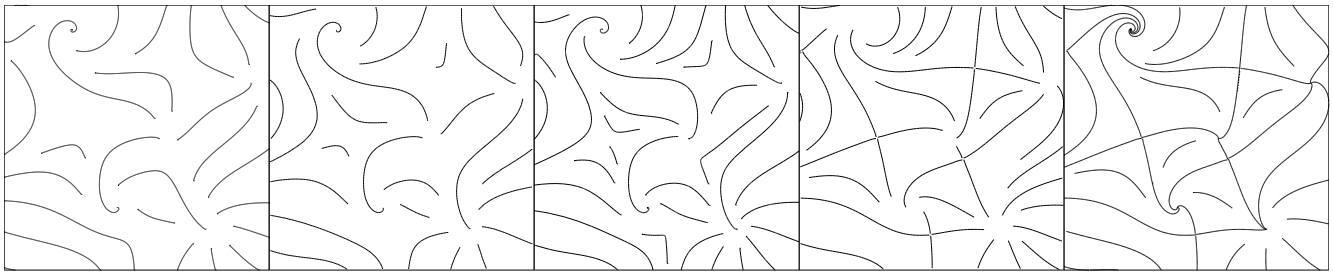


Fig. 19. Comparison in the streamline placement with a very low density for flow field 9.

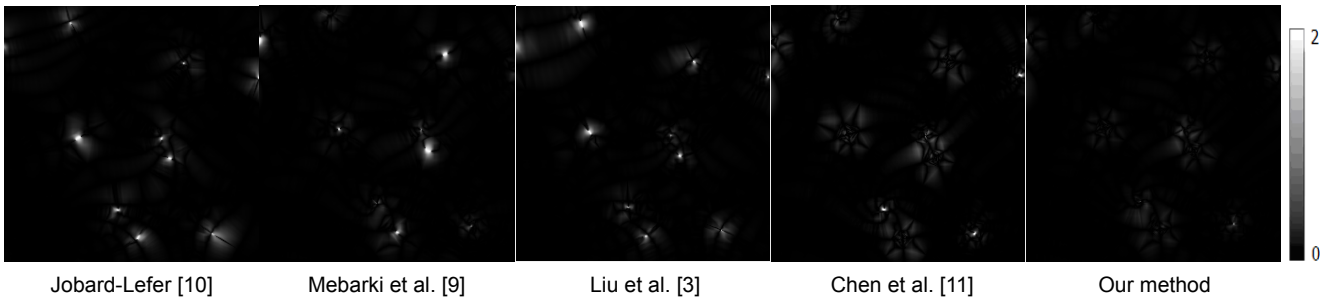


Fig. 20. Comparison in the reconstruction error of streamline placement for flow field 9.

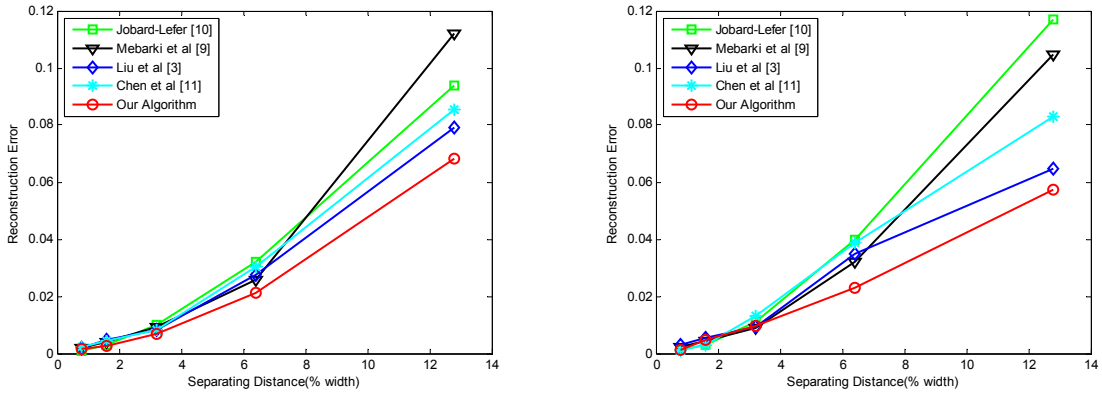


Fig. 21. Quantitative comparison of our method with Jobard and Lefer's method [10], Mebarki et al.'s method [9], Liu et al.'s method [3], and Chen et al.'s method [11] in reconstruction error versus separating distance based on Tables 1 and 2.

Feature Preservation As Fig. 18 shows, our algorithm retains topological elements like singularities and separatrices. The critical points, periodic orbits, and separatrices are not highlighted, as is the case with [11], to facilitate the comparison of our algorithm with others. However, highlighting these features can be easily implemented in our method and might be useful, especially for vector fields on surfaces, since only portions of a periodic orbit may be visible for any given viewpoint. In the other four algorithms, the loss of topology information may not be noticeable with a high streamline density (bottom of Fig. 18), but is pronounced with a low one (top of Fig. 18). Fig. 19 compares the five aforementioned algorithms using very sparse streamline placements, of which the one generated by our method is able to preserve the flow topology very well.

Reconstruction Fidelity The reconstruction error evaluation method described in [5] is applied to our test results. Fig. 20 compares the five algorithms regarding the reconstruction error of streamline placement, with white indicating the highest reconstruction error and black the lowest. Fig. 21, Table 1, and Table 2 compare these five algorithms in terms of the reconstruction error for flow fields 1 and 9. Our method produces a low reconstruction error, particularly as the streamline density decreases. In other words, our approach is more effective than the other four in reconstructing a flow field from a small number of streamlines.

TABLE 1
QUANTITATIVE COMPARISON IN UNIFORM DENSITY AND RECONSTRUCTION ERROR FOR FLOW FIELD 1

Separating Distance (% width)		0.8	1.6	3.2	6.4	12.8
Number of Streamlines	Jobard-Lefer [10]	551	261	130	58	24
	Mebarki et al. [9]	526	231	136	67	31
	Liu et al. [3]	497	219	150	78	32
	Chen et al. [11]	492	236	133	59	35
	Our Algorithm	466	295	152	78	45
MSE of Gray-level	Jobard-Lefer [10]	0.0189	0.0228	0.0219	0.0136	0.0072
	Mebarki et al. [9]	0.0145	0.0185	0.0205	0.0142	0.0109
	Liu et al. [3]	0.0134	0.0171	0.0161	0.0111	0.0062
	Chen et al. [11]	0.0176	0.0187	0.0190	0.0134	0.0070
	Our Algorithm	0.0142	0.0175	0.0184	0.0132	0.0095
Reconstruction Error	Jobard-Lefer [10]	0.0010	0.0030	0.0098	0.0322	0.0937
	Mebarki et al. [9]	0.0019	0.0040	0.0094	0.0259	0.1122
	Liu et al. [3]	0.0020	0.0048	0.0081	0.0276	0.0790
	Chen et al. [11]	0.0024	0.0043	0.0083	0.0302	0.0856
	Our Algorithm	0.0016	0.0026	0.0069	0.0211	0.0682
Running Time	Jobard-Lefer [10]	0.546	0.280	0.156	0.109	0.063
	Mebarki et al. [9]	0.274	0.147	0.090	0.067	0.042
	Liu et al. [3]	0.164	0.056	0.034	0.025	0.013
	Chen et al. [11]	0.574	0.297	0.177	0.127	0.066
	Our Algorithm	0.818	0.420	0.237	0.168	0.097

TABLE 2
QUANTITATIVE COMPARISON IN UNIFORM DENSITY AND RECONSTRUCTION ERROR FOR FLOW FIELD 9

Separating Distance (% width)		0.8	1.6	3.2	6.4	12.8
Number of Streamlines	Jobard-Lefer [10]	495	234	114	54	26
	Mebarki et al. [9]	451	180	115	57	25
	Liu et al. [3]	437	200	136	60	33
	Chen et al. [11]	467	216	115	59	33
	Our Algorithm	389	247	123	59	28
MSE of Gray-level	Jobard-Lefer [10]	0.0199	0.0237	0.0220	0.0138	0.0073
	Mebarki et al. [9]	0.0135	0.0182	0.0194	0.0135	0.0082
	Liu et al. [3]	0.0141	0.0173	0.0162	0.0100	0.0062
	Chen et al. [11]	0.0164	0.0175	0.0183	0.0126	0.0071
	Our Algorithm	0.0130	0.0161	0.0175	0.0124	0.0077
Reconstruction Error	Jobard-Lefer [10]	0.0011	0.0031	0.0110	0.0400	0.1169
	Mebarki et al. [9]	0.0023	0.0045	0.0089	0.0319	0.1046
	Liu et al. [3]	0.0030	0.0055	0.0092	0.0348	0.0649
	Chen et al. [11]	0.0012	0.0029	0.0130	0.0389	0.0831
	Our Algorithm	0.0014	0.0048	0.0098	0.0231	0.0574
Running Time	Jobard-Lefer [10]	0.421	0.218	0.125	0.078	0.031
	Mebarki et al. [9]	0.213	0.116	0.074	0.047	0.022
	Liu et al. [3]	0.156	0.046	0.028	0.012	0.010
	Chen et al. [11]	0.456	0.237	0.141	0.089	0.039
	Our Algorithm	0.630	0.327	0.193	0.179	0.048

Time Complexity The computational cost of our algorithm depends on (1) topology extraction, (2) seeding path selection, and (3) streamline integration under density control. Given the number of grid points N and the number of singularities C of a flow field, (1) needs $O(N)$ time for singularity computation and $O(C)$ time for topology region extraction. The combined execution time of (2) and (3) is $O(G)$ where G is the number of samples that are placed approximately evenly within the flow field. Thus the overall time complexity of our approach is $O(\max(N, C, G))$. Tables 1 and 2 compare the aforementioned five sample-based streamline placement algorithms for flows 1 and 9, respectively. The timing results measured in seconds are based on a 2.0GHz, 2GB RAM PC running Windows Vista. Within a large range of varying densities, our approach takes one to three times

the time needed by Jobard and Lefer's [10], Mebarki et al.'s [9], or Chen et al.'s method [11] to create an approximately equal-density streamline placement. In fact, this ratio is largely due to the two-stage layout strategy adopted in our algorithm, i.e., generating dumb streamlines in the orthogonal flow during the first stage to determine seeding paths and placing visible streamlines in the original flow during the second stage. On the other hand, our method exhibits an obvious advantage over the three in placement quality by showing accurate separatrices to preserve flow topology, independent of the density, in a layout of evenly-spaced streamlines. This capability facilitates level-of-detail investigation of complex flows (as in Fig. 14), for which a rapidly generated coarse yet uniform streamline placement reveals the precise topological skeleton to steer high-resolution exploration of regions of interest. This gain is particularly noticeable and may further blur the aforementioned performance gap in terms of overall visualization efficiency when handling unknown large-scale real-world flows. Liu et al.'s algorithm [3] is currently the fastest one of the five we compared, mainly because of the exclusive use of an adaptive step size integrator with error control plus a cubic Hermite polynomial curve sampler. Our method allows for incorporation of such an integrator to enhance the computational performance, while maintaining the novelty of addressing streamline placement through the orthogonal flow domain.

6 Conclusion and Future Work

We have presented a novel topology-aware evenly-spaced streamline placement algorithm. By exploiting the intrinsic relationship between a flow field and its orthogonal counterpart, a preferred seeding path crossing all the streamlines within each topological region is determined to place evenly-spaced long streamlines while preserving flow topology with accurate separatrices in the resulting layout.

The results demonstrate that our algorithm achieves great placement uniformity and high streamline continuity at a moderate computational cost. It outperforms the flow-guided seeding strategy [2] and Chen et al.'s version of neighborhood seeding strategy [11] in maintaining overall streamline uniformity as well as reconstruction fidelity, and is more effective than other methods in preserving the flow topology in the resulting placement.

As for future work, we would like to improve the robustness of topological region decomposition for handling even more complex topological structures involving periodic orbits and saddle-connected loops. In addition, we are interested in extending this algorithm to curved surface and volume flows.

Acknowledgements

This work was supported in part by the Northern Gulf Institute, a NOAA Cooperative Institute. The authors would like to thank Guoning Chen, Yuan Chen, Alex Pang, Bruno Jobard, and Abdelkrim Mebarki for providing datasets and/or images. Additional thanks go to Abdelkrim Mebarki for his online executable program.

References

[1] G. Turk and D. Banks, "Image-Guided Streamline Placement," *Proc. SIGGRAPH '96*, pp. 453-460, 1996.

[2] V. Verma, D. Kao, and A. Pang, "A Flow-Guided Streamlines Seeding Strategy," *Proc. IEEE Visualization '00*, pp. 163-170, 2000.

[3] Z. Liu, R. J. Moorhead II, and J. Groner, "An Advanced Evenly Spaced Streamline Placement Algorithm," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, pp. 965-972, 2006.

[4] Z. Liu and R. J. Moorhead II, "Robust Loop Detection for Interactively Placing Evenly Spaced Streamlines," *IEEE Computing in Science and Engineering*, vol. 9, no. 4, pp. 86-91, 2007.

[5] Y. Chen, J. Cohen, and J. Krolik, "Similarity-Guided Streamline Placement with Error Evaluation," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 1, pp. 1448-1455, 2007.

[6] J. L. Helman and L. Hesselink, "Representation and Display of Vector Field Topology in Fluid Flow Data Sets," *IEEE Computer*, vol. 22, no. 8, pp. 27-36, 1989.

[7] J. L. Helman and L. Hesselink, "Visualizing Vector Field Topology in Fluid Flows," *IEEE CG&A*, vol. 11, no. 3, pp. 36-46, 1991.

[8] X. Tricoche, "Vector and Tensor Field Topology Simplification, Tracking and Visualization," PhD thesis, Schriftenreihe Fachbereich Informatik (3), Universität Kaiserslautern, Germany, 2002.

[9] A. Mebarki, P. Alliez, and O. Devillers, "Farthest Point Seeding for Efficient Placement of Streamlines," *IEEE Visualization '05*, pp. 479-486, 2005.

[10] B. Jobard and W. Lefer, "Creating Evenly-Spaced Streamlines of Arbitrary Density," *Proc. Eurographics Workshop on Visualization in Scientific Computing*, pp. 45-55, 1997.

[11] G. Chen, K. Mischaikow, R. S. Laramee, P. Pilarczyk, and E. Zhang, "Vector Field Editing and Periodic Orbit Extraction Using Morse Decomposition," *IEEE Trans. Visualization and Computer Graphics*, vol. 13. no. 4, pp. 769-785, 2007.

[12] X. Mao, Y. Hatanaka, H. Higashida, and A. Imamiya, "Image-Guided Streamline Placement on Curvilinear Grid Surfaces," *Proc. IEEE Visualization '98*, pp. 135-142, 1998.

[13] B. Jobard and W. Lefer, "Multiresolution Flow Visualization," *Proc. Ninth International Conf. in Central Europe on Computer Graphics, Visualization, and Computer Vision (WSCG '01)*, pp. 33-37, 2001.

[14] B. Jobard and W. Lefer, "Unsteady Flow Visualization by Animating Evenly-Spaced Streamlines," *Proc. Eurographics '00*, pp. 21-31, 2000.

[15] L. Li and H.-W. Shen, "Image-Based Streamline Generation and Rendering," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 3, pp. 630-640, 2007.

[16] Z. Liu and R. J. Moorhead II, "Interactive View-Driven Evenly Spaced Streamline Placement," *IS & T / SPIE Conference on Visualization and Data Analysis (VDA '08)*, 68090A, pp. 1~12, 2008.

[17] X. Ye, T. K. David, and A. Pang, "Strategy for Seeding 3D Streamlines," *IEEE Visualization '05*, pp. 471-478, 2005.

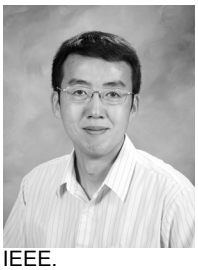
[18] L. Li, H.-H. Hsieh, and H.-W. Shen, "Illustrative Streamline Placement and Visualization," *Proc. IEEE Pacific Visualization Symposium '08*, pp. 79-86, 2008.

[19] G. Scheuermann, B. Hamann, K. I. Joy, and W. Kollmann, "Localizing Vector Field Topology," *Data Visualization: The State of the Art*, Kluwer Academic Press, pp. 19-36, 2003.

[20] T. Wischgoll and G. Scheuermann, "Detection and Visualization of Closed Streamlines in Planar Flows," *IEEE Trans. Visualization and Computer Graphics*, vol. 7, no. 2, pp. 165-172, 2001.

Keqin Wu received the BE degree and the ME degree in computer engineering from Ocean University of China in 2001 and 2004, respectively. She is currently a PhD candidate in computer engineering at Mississippi State University. She is studying flow visualization and its applications under the joint direction of Dr. Song Zhang and Dr. Robert J. Moorhead II.

Zhanping Liu received the PhD degree in computer science (2000) and the BS degree in mathematics (1992) from Peking University and NanKai University, respectively, P. R. China. He is a research staff member with Kitware, Inc. Prior to this position, he was a research scientist with the Visualization, Analysis, and Imaging Lab of the High-Performance Computing Collaboratory (formerly ERC) at Mississippi State University (2001 ~ 2008) and earlier a post-doctoral associate with the Micro-CT Lab in Department of Radiology at the University of Iowa (2000 ~ 2001). His research interests lie in a wide variety of aspects of scientific visualization, particularly flow visualization involving geometry-based as well as texture-based techniques. More information is available at www.zhanpingliu.org.



Song Zhang received the BS degree in computer science from Nankai University, China in 1996 and the PhD degree in computer science from Brown University in 2006. He has been an assistant professor in the Department of Computer Science and Engineering, Mississippi State University since 2006. His research interests include scientific visualization, medical imaging, and computer graphics. He is a member of the IEEE.

Robert J. Moorhead II received a BSEE from Geneva College in 1980 and a MSEE and his Ph.D. from North Carolina State University in 1982 and 1985 respectively. He is currently the Billie J Bill Professor of Electrical and Computer Engineering at Mississippi State University and Deputy Director of the Geosystems Research Institute. He was previously a Research Staff Member at the IBM T.J. Watson Research Center. His current research interests include computationally demanding visualization and analysis issues, proving the value of visualization, and showing the relative value of different visualization techniques. He has previously conducted research in computer communications and image/video coding. Dr. Moorhead has published more than 100 peer-reviewed manuscripts. He is a senior member of the IEEE.