# Dual Streamline Seeding

Olufemi Rosanwo*     Christoph Petz†     Steffen Prohaska‡     Hans-Christian Hege§     Ingrid Hotz¶

Zuse Institute Berlin, Germany

## ABSTRACT

This work introduces a novel streamline seeding technique based on dual streamlines that are orthogonal to the vector field, instead of tangential. The greedy algorithm presented here produces a net of orthogonal streamlines that is iteratively refined resulting in good domain coverage and a high degree of continuity and uniformity. The algorithm is easy to implement and efficient, and it naturally extends to curved surfaces.

**Keywords:** Streamline placement, Flow visualization, Dual streamlines, Surface vector fields.

**Index Terms:** I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

## 1 INTRODUCTION

CFD based flow visualization and analysis is nowadays a standard technique in many different application areas from technical engineering to life science research. For visual analysis of flow phenomena, among other aspects, tangent vector fields on 2D domains are especially important for at least two reasons. First, analyzing velocity fields in planar slice cuts of a flow domain is very common in practice, as these cuts provide a good overview and can be compared to experimentally observable Particle Image Velocimetry (PIV) measurements. Second, flow phenomena in near wall regions, i.e. close to fixed geometries that act as boundaries, are very important and mainly motivated our research. For instance, the tangent vector field of the shear stress at the curved boundary surface of a flow domain provides insight to important flow phenomena of viscous fluids like vortex formation and flow separation.

Several approaches for the visualization of tangent vector data on 2D surfaces exist, ranging from simple hedgehog plots to advanced feature extraction methods. Among them, streamline based methods are widely accepted for depicting flow at a glance. In contrast to texture based methods like line integral convolution (LIC), streamlines provide sparse visualizations that focus on significant structures and can thus be combined with other visualization techniques, or used to compare vector fields. Additionally, streamlines can be annotated with glyphs showing the flow direction or can be scaled, for example by the velocity magnitude.

The key problem of streamline based methods, which we will refer to as the *streamline placement problem*, consists of finding a good representative set of streamlines depicting the vector field visually comprehensibly and completely. To evaluate the quality of our results and compare them to other approaches, we refer to the following general criteria, proposed by Verma et al. [21]:

**Coverage:** No important features of the vector field should be missed and the streamlines should cover the whole domain.

---

*e-mail:rosanwo@zib.de

†e-mail:petz@zib.de

‡e-mail:prohaska@zib.de

§e-mail:hege@zib.de
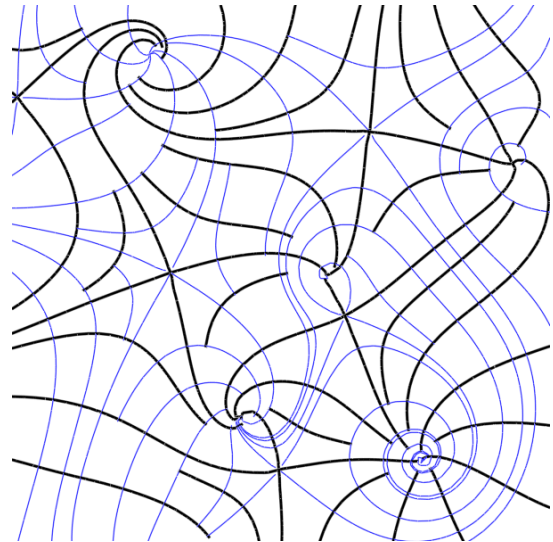
¶e-mail:hotz@zib.de

Figure 1: Overlay of primal (black) and dual (blue) streamlines.

**Uniformity:** The distribution of streamlines should be more or less uniform across the domain.

**Continuity:** Long continuous streamlines are preferred over short ones.

## CONTRIBUTIONS

For solving the streamline placement problem, we propose a greedy algorithm, based on the idea of dual vector fields to guide the seeding process. Our main contribution resides in the introduction of *dual streamlines* that run orthogonal to the given vector field as opposed to *primal streamlines* that run tangential. Since seeds for new streamlines are only placed along these curves, the search space for seed placement is reduced to a set of curves. This applies to planar and curved surfaces alike. Monitoring interaction between the two sets of streamlines enables us to fully saturate the domain with streamlines, approximate and optimize streamline density distribution and infer a simple, yet effective streamline termination strategy.

Dual Streamline Seeding comes with the following advantages:

- The algorithm has only a single parameter that controls streamline density.

- The approach naturally extends to curved surfaces without the need of surface parametrization.

- All distances used are based on arc length. Thus, we avoid explicit computation of geodesic distances.

- Use of topology highlights important features, even at low streamline density.

## 2 RELATED WORK

Good overviews of flow visualization techniques in general can be found in Laramee et al. [8] and Post et al. [16]. In the following, we focus on streamline based flow visualization on 2D surfaces. A number of techniques with different objectives have been developed. We present them grouped into four categories: image based, direct, feature based and vector field dependent. Our contribution is closely related to the direct approaches.

Image based approaches define an energy function for the streamlines in image space and search for a minimum. Turk and Banks [20] use a low-pass filtered image of the streamlines to measure the deviation from a predefined saturation level across the image. Starting from an initial streamline set, the energy is minimized by iteratively lengthening, shortening, deleting, or moving streamlines in a global optimization process. Due to the large number of possible operations in each step, the method exhibits exponential running time. Visual appeal, uniformity and continuity of the streamlines is superior and serves as the benchmark for our method. Mao et al. [12] extend this approach to parametrizable curved surfaces in 3D. For the computation of the local filter size, they make use of the deformation tensor of the parameter space mapping. Schlemmer et al. [19] introduce an image based approach where the streamline density is steered by a user defined density function. They use a greedy algorithm where new streamlines are added one by one.

Direct methods place new streamlines with a certain heuristic that is expected to provide good overall seeding results without computing a global energy function. Jobard and Lefer [3] achieve good streamline uniformity, but streamline lengths, and thus continuity is unsatisfactory. They start new streamlines at a separation distance apart from all previously computed streamlines until the domain is fully covered. The technique is extended to unsteady flows in [4], its application for multiresolution flow visualization is described in [5]. The method proposed by Liu et al. [11] improves continuity by prioritizing streamline elongation over new streamline insertion. The seeding strategy presented by Mebarki et al. [15] is closest to our approach. In a greedy algorithm, they start new streamlines in the center of the biggest remaining voids, and achieve good continuity and uniformity of the streamlines. Similarly, we search for the largest uncovered areas approximately on dual streamlines and start new streamlines at the midpoint of those segments. Mebarki et al. make heavy use of Delaunay triangulation, which makes their approach hard to apply on curved surfaces in 3D.

Verma et al. [21] present a feature based approach that strives for good coverage of the vector field topology by using fixed seeding templates in the vicinity of the critical points. The template size is determined by a Voronoi decomposition of the domain. Remaining voids are filled by a Poisson random seeding.

Vector field dependent methods focus on coverage of important features by defining similarity measures for streamlines and adding new streamlines at places with great streamline differences. Chen et al. [1] propose a metric based on the distance, shape and orientation of nearby streamlines, Li et al. [9] propose a metric based on a distance transformation of the streamlines.

Most of these seeding techniques are primarily developed for planar 2D domains. Their application on curved surfaces embedded in 3D space is often not obvious, as distance and similarity measures are harder to compute. We propose an algorithm that does not rely on planarity of the domain. Thus, we form a bridge between methods for planar 2D domains and intrinsic 3D methods, as suggested by Salzbrunn and Scheuermann [18], Mattausch et al. [13], Ye et al. [22], and Li and Shen [10].
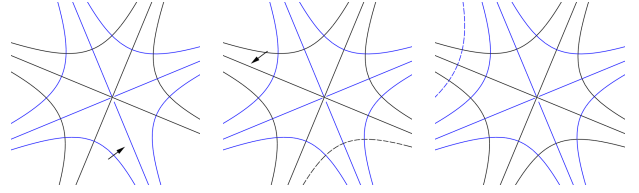


Figure 2: Outline of algorithm behavior. Left: largest segment belongs to blue set (arrow), a new streamline for the black set will be started at its midpoint. Middle: new streamline has been inserted (shown as dotted line). Now, black set contains the largest segment (arrow), whose midpoint will again serve as seed point for the next streamline. Right: result of next streamline insertion. Newly inserted streamline is drawn dotted.

## 3 DUAL STREAMLINE SEEDING

Our algorithm takes as input a surface embedded in 3D, a tangent vector field $v \in \mathbb{R}^3$ and a surface normal field $n \in \mathbb{R}^3$. The two fields are often – but not necessarily – given as a discrete set of tangent data vectors, a set of surface normals and an associated interpolation scheme, such that $v$ as well as $n$ can be evaluated at each point on the surface. Furthermore, user-supplied separation distance $\delta$ defines the minimal distance between a seed and a neighboring streamline and thus controls the resulting streamline density. Our algorithm outputs a set of streamlines, which satisfy the quality criteria listed in Sect. 1.

For solving the streamline placement problem, we manage two sets of streamlines simultaneously:

**Primal Streamlines** are curves inside the domain of $v$ that are tangential to $v$ at every point. These streamlines form our result.

**Dual Streamlines** are curves inside the domain of $v$ that are tangential to the dual field $R(v) := v \times n$, i.e. orthogonal to $v$ at every point. They are auxiliary streamlines, i.e. they are not visible in the resulting image.

Intersections between primal and dual streamlines subdivide a streamline into consecutive segments. These segments are key to our algorithm. We denote the set of primal streamline segments as $P$ and the set of dual segments as $D$. Each set is represented by a priority queue ordered by the arc length of a segment. In the following, the term streamline refers to primal and dual streamlines alike.

Given suitable starting sets for $P$, $D$ (see Sect. 3.1), our greedy seeding algorithm works as follows: In each iteration, the longest remaining segment is detected and a new bidirectional streamline is started from its midpoint in orthogonal direction. Fig. 2 illustrates this process. The algorithm terminates when no segment longer than $2\delta$ remains.

Considering both sets of streamlines at the same time and keeping track of mutual intersections gives a good hint on the overall density distribution. We postulate:

> *Long streamline segments without intersection indicate voids in the dual streamline set.*

This is the central justification for our approach, because it bears a directive for seed placement. The next seed for a streamline in $P$ should be placed at the center of the longest remaining segment in $D$ and vice versa. This procedure enables us to efficiently find voids based on a one-dimensional distance metric (arc length) instead of Euclidean distance in image space or geodesic distance. This makes our approach very suitable for curved surfaces, where other measures are hard to apply correctly or are computationally
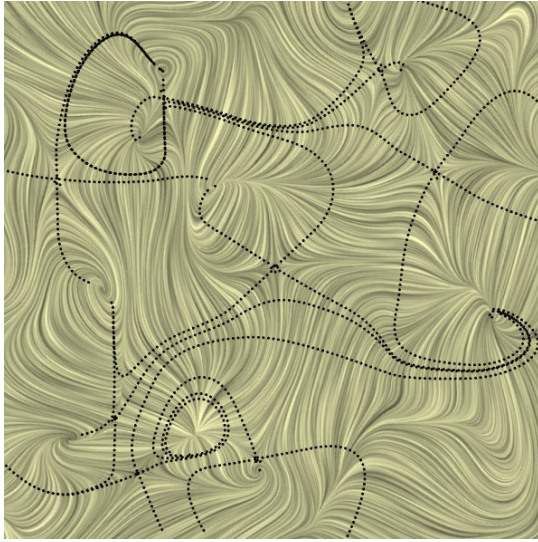
Figure 3: Initial set of dual streamlines, i.e. dual topology shown as dotted lines. The vector field is shown via LIC. Although the initial density distribution is non-uniform, a good starting coverage is achieved and uniformity of primal streamlines is not affected.

expensive. Note that our greedy choice is only *locally* optimal, i.e. only in terms of the given 1D distance metric and the above postulate. The next seed is thus not necessarily put in the largest 2D void, but each additional streamline improves the approximation.

### 3.1 Initialization

To start our iterative seeding process, we need a first set of streamlines as initialization. This starting configuration should guarantee accessibility of all points inside the domain through a series of iterations, as described in Sect. 3.2 irrespective of the vector field structure. Although a non-uniform arrangement in the initial streamlines will be balanced in the course of the algorithm, a good starting configuration also significantly reduces processing time. A structure that matches all these requirements is vector field topology.

We thus compute an initial set of dual streamline segments $D$ by extracting the topological skeleton of $R(v)$, see e.g. [2]. This gives a reasonable starting coverage of the domain and good coverage of all critical points. Fig. 3 shows the result of initialization on an artificial vector field.

In the rare case that this initialization fails to produce any starting curves, i.e. if $v$ possesses no topology, we initialize $D$ by tracing a small number of randomly seeded streamlines in $R(v)$. Actually, a single dual streamline would suffice at this point, but an unfavorable seed location would slow down algorithm convergence. For all examined cases, the use of five of these streamlines is sufficient.

After $D$ has been initialized, $P$ is initialized analogously with the topological skeleton of $v$, which ensures clearly visible separatrices especially at very low densities. Note that initializing $P$ includes intersection tests and streamline termination, as described in the following sections. If the emphasis of topological structure is not desired, $P$ can be initialized empty. Fig. 4 compares these two alternatives.

### 3.2 Iteration

Each iteration consists of two major parts:

1. Extract the longest segment from $P$ or $D$. The midpoint of this segment will serve as the next seed.
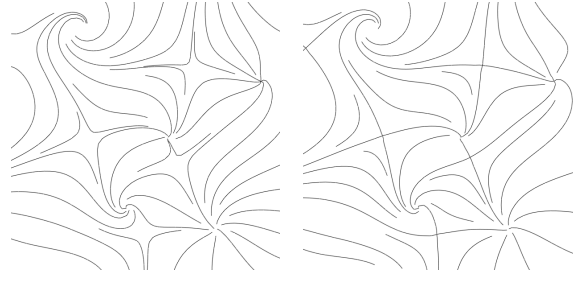


Figure 4: Effects of different initializations. Left: $P$ empty, Right: $P$ initialized with topology of the vector field. The latter option provides clearly visible separatrices and guarantees conservation of perceptibility of topological structure at low densities.

2. Trace a new streamline in forward and backward direction and test for intersections.

For illustration, let $s \in D$ be the longest remaining segment. Now, starting from the midpoint of $s$, a new primal streamline is traced in $v$. Intersections with segments in $D$ split them into smaller segments at the intersection point. The currently traced streamline is also divided into segments at those intersection points. The newly traced segments are added to $P$.

If $s \in P$, the process is analogous, resulting in new streamline segments for $D$ tested for intersection against all segments in $P$.

#### Avoiding clutter

Each streamline-based approach faces the problem of clutter, i.e. streamlines cluster at certain points or along certain lines in the field. These clusters are distracting if they do not coincide with topological features of the field and should thus be avoided.

Similar to the seed criteria, we use the arc length of intersected streamline segments as termination condition. Each time an intersection between the currently traced streamline and another segment is encountered, the minimal distance $d$ to one of the segment's endpoints is determined. If $d$ falls below the termination distance $\delta_{term}$, tracing is stopped. $\delta_{term}$ is set relative to $\delta$, a value of $\frac{\delta}{2}$ worked well for all our examples.

Note that $d$ is measured in arc length. Thus, contrary to standard Euclidean distance metric or similarity distance [1], it is directly applicable on curved surfaces.

### 3.3 Termination

The algorithm terminates if no voids are left in either of the two segment sets. According to our main postulate, this corresponds to an upper bound on the length of all remaining streamline segments. This upper bound is set to $2\delta$ and controls the density of the resulting streamline image. More formally, the termination condition can be formulated as:

$$\forall p \in P : ||p|| < 2\delta \wedge \forall d \in D : ||d|| < 2\delta.$$

## 4 RESULTS

In this section, we present the results of our technique and compare them – where applicable – to other seeding methods on artificial vector fields and CFD data. We implemented the method of Verma et al. [21] and use an implementation of Mebarki et al. [15], which is freely available [14]. For comparison to the techniques of Turk and Banks [20] and Jobard and Lefer [3], we use pictures from [3] by courtesy of the authors. Where possible, we further compare running times.
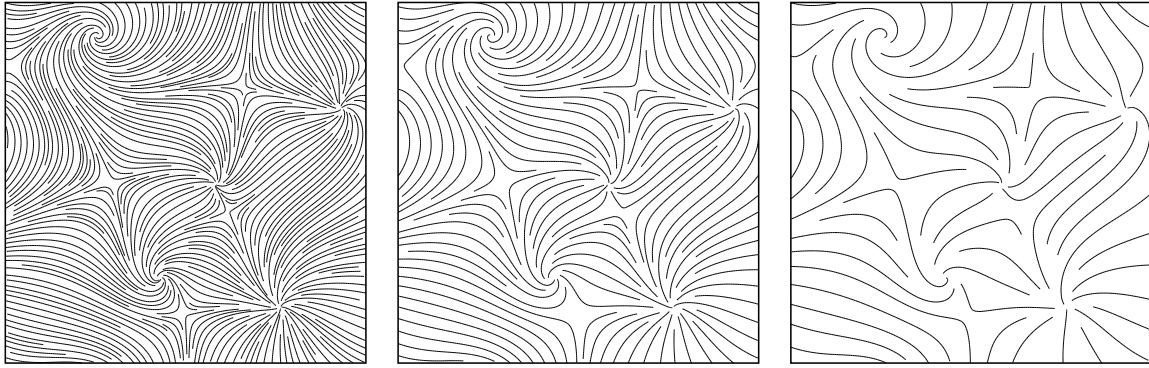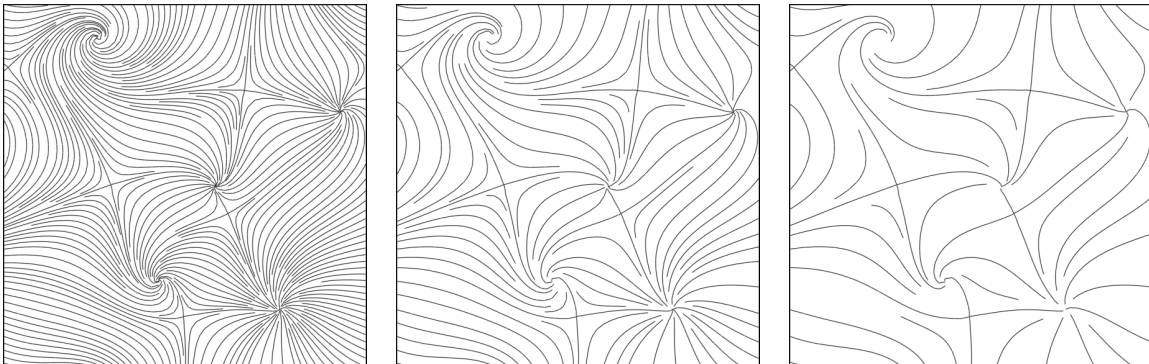
(a) Turk/Banks
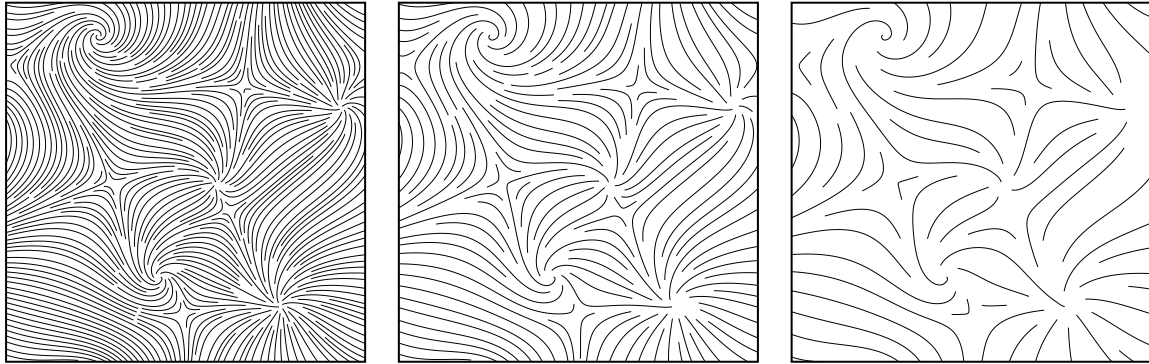


(b) Jobard/Lefer
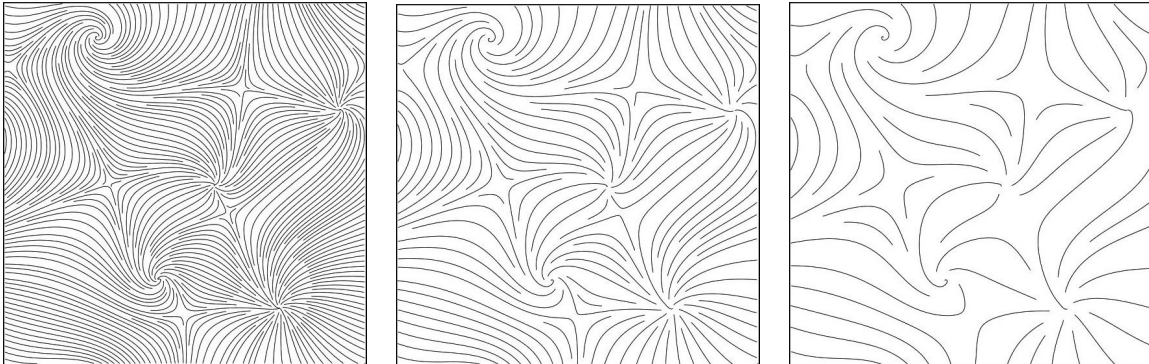


(c) Mebarki et al.



(d) Dual Seeding

Figure 5: Comparison of streamline placement techniques with decreasing densities. Separation distance from left to right: 0.84%, 1.68%, 3.36% of domain width. Pictures on the first two rows were provided by the authors of [3].
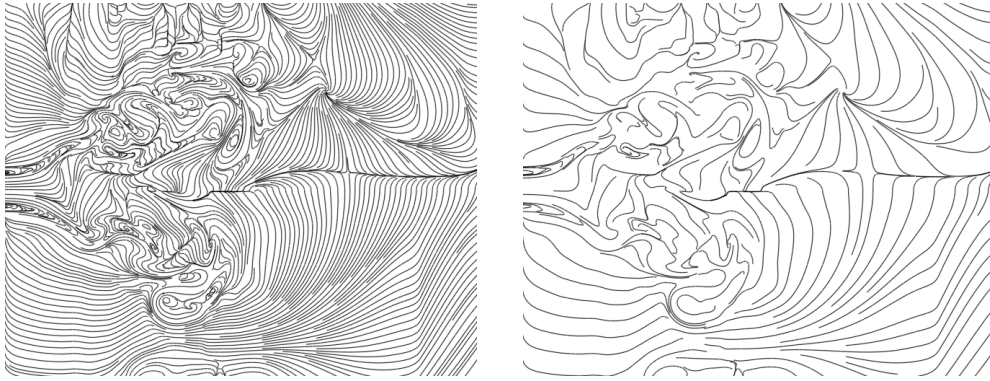
Figure 6: Swirling jet entering fluid at rest. Depicted is a planar slice of a 3D velocity field. Separation distance is 0.5% (left) and 1.5% (right) of domain width. Areas of turbulent and laminar flow are well distinguishable at both densities.

## 4.1 Planar Domains

In the following, we compare Dual Streamline Seeding to four other streamline placing algorithms methods, representing three classes of seeding techniques:

- image-based optimization [20],

- direct seeding [3] and [15], and

- feature-based [21].

As discussed in Sect. 1, quality criteria for streamline placement are uniformity, continuity, coverage. In the following examples, coverage of important flow features is given with all five methods, thus, we will focus on continuity and uniformity. Sect. 5 contains further discussion on coverage. Since there is no standard to compare continuity and uniformity quantitatively, we compare the results visually.

Fig. 5 compares our technique to the listed image based and direct approaches. In terms of continuity, our algorithm gives comparable results to the Farthest Point technique and the method of Turk/Banks and better results than the algorithm of Jobard/Lefer. The latter tends to produce short, separated streamlines along the same path of the flow where the viewer would expect one continuous streamline. This should be avoided to give an impression of flow coherency in a steady flow. Our algorithm does not enforce overall uniformity of streamline density as strictly as the other methods. Especially, clutter control is less regular than with Euclidean distance measure, which is employed by the other approaches. On the other hand, the resulting local deviation of density leads to a better accentuation of the critical points, as discussed in Sect. 5.

Fig. 6 demonstrates results for different separation distances on a planar cut of a 3D vector field. It contains 12,524 data vectors and results from a simulation of a swirling jet entering fluid at rest. Despite the topological complexity of the data set, regions of laminar and turbulent flow can well be distinguished. Neither do the turbulent areas, which are densely covered by dual streamlines, exhibit cluttering, nor are there any voids in areas of laminar flow.

Fig. 7 compares Dual Seeding to the flow-guided seeding strategy of Verma et al. with additional Poisson-distributed seeds to fill the voids. Our algorithm gives better results in terms of continuity and uniformity, especially with high densities. Without additional seeds, the approach of Verma et al. lacks coverage for regions without critical points. On the other hand, the additional seeds further diminish continuity and uniformity.
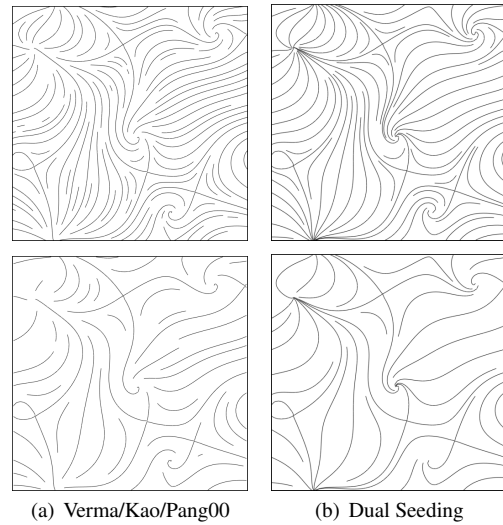


(a) Verma/Kao/Pang00       (b) Dual Seeding

Figure 7: Comparison to feature-based technique by Verma et al. Separation distances: top 2%, bottom 3.5% of domain width. Poisson-distributed seeds employed by Verma et al. introduce significant discontinuities, whereas Dual Seeding results in long continuous streamlines, while still capturing topological features.
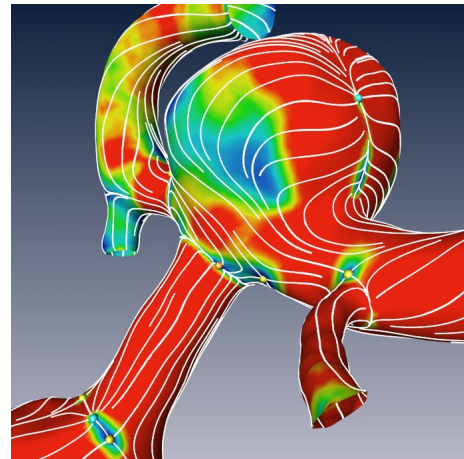


Figure 8: Overlay of wall shear stress streamlines and color-coded pressure distribution on a cerebral aneurysm. Critical points are highlighted by small spheres.
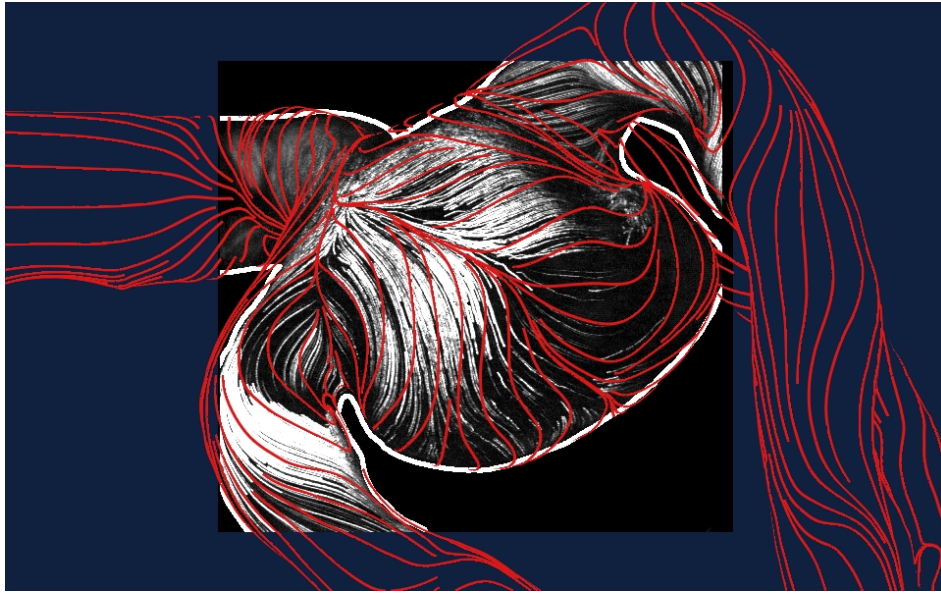
Figure 9: Comparison of experimental flow and CFD results of a cerebral aneurysm. Particle traces of the near wall flow are shown in the background [6], the wall-shear-stress vector field of a roughly aligned CFD model is depicted with red streamlines. The comparison shows a good agreement of the fields.
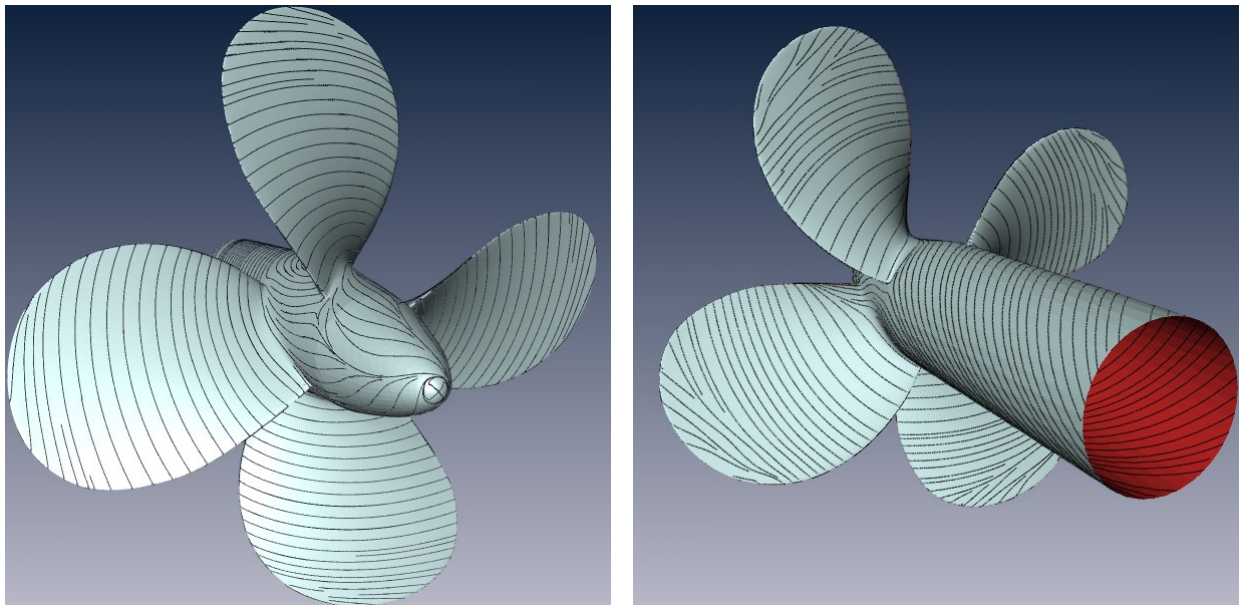


Figure 10: Streamlines of the wall shear stress on a ship propeller. Left: Pressure side, Right: Suction side. Flow separation can be observed at the tip of the blades at the suction side.

| Separation distance | Verma et al. [21] | Mebarki et al. [15] | Dual Seeding |
|---|---|---|---|
| 0.5% | 587ms | 578ms | 154ms |
| 1% | 182ms | 250ms | 71ms |
| 2% | 58ms | 125ms | 39ms |
| 5% | 28ms | 46ms | 21ms |

Table 1: Comparison of running time including initialization. For the examined case, our algorithm is faster by a factor of 2-3.

### 4.2 Non-Planar Domains

Fig. 8 combines a color-coded visualization of pressure with our streamline seeding result and extracted critical points of the wall shear stress. Both fields resulted from CFD simulation of blood flow in a reconstructed aneurysm. The streamlines exhibit the same degree of continuity, uniformity and coverage as the planar examples. Overall flow behavior as well as local topological details are well observable. The structure of the pressure field is still well perceptible due to the sparsity of the streamlines.

We applied our method in Fig. 9 for the validation of a CFD solution with experimental data. A transparent silicon model of a cerebral aneurysm was used for the experimental flow, particle traces of the near-wall-flow were recorded in a gray-scale image with the wall PIV technique by Kertzscher et al. [6]. The same geometry and flow condition was used for the CFD computation. The PIV image and streamlines of the wall shear stress of the CFD solution are superimposed. Due to the sparsity of the streamline visualization, similarities and differences of the two flow fields are well observable.

Another visualization of wall shear stress resulting from CFD simulation on a curved surface is shown in Fig. 10. Flow separation can be observed at the back side of the propeller blades. Although the field barely possesses topological features as starting points, the resulting streamline image still exhibits complete coverage and a great degree of uniformity and continuity.

### 4.3 Timings and Scalability

Timings were measured on a standard AMD64 3GHz computer. We implemented our method for triangulated surfaces. Streamlines are traced triangle-wise locally exact by the method of Kipfer et al. [7], keeping track of the position and arc-length to the seed for each sample. Streamline segments are linked to the triangles as index structure for fast intersection tests. For more implementation details, see [17]. Table 1 compares running times of our method to [15] and [21]. The three techniques were applied to the vector field depicted in Fig. 5, which consists of 50x50 data vectors. Our algorithm is faster than the methods proposed in [15] and [21] by a factor of 2-3. It further exhibits a significantly slower growth than the other two approaches, as Fig. 11 illustrates.

### 5 Discussion

Our algorithm combines the advantages of the approaches of Mebarki et al. [15] and Verma et al. [21] to achieve high quality seed placement according to the mentioned quality criteria, improve on the running time of comparable approaches and depart from the restriction to planar domains.

Analogous to [15], we use a greedy strategy to find the biggest remaining void and place a seed in its center. This favors long streamlines, satisfying the continuity criterion. Contrary to [15], our search space for seed placement is 1-dimensional, increasing efficiency, especially for curved surface domains. Furthermore, streamline distance measurement is induced by the respective dual streamline set, and is thus iteratively refined. This particularly impacts streamline termination, allowing early streamlines to gather, which they usually do at critical points, while streamlines added
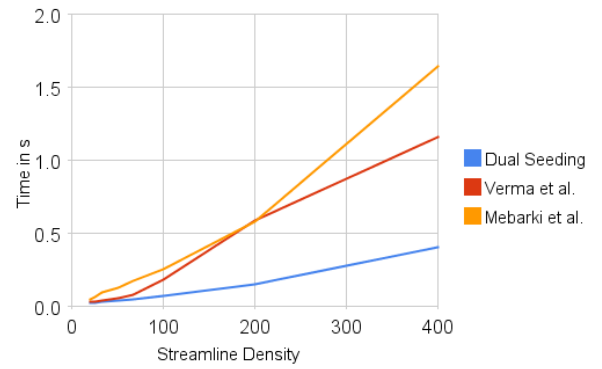


Figure 11: Running times in relation to streamline density, which is the inverse of the separation distance. Dual Seeding exhibits a more moderate growth than the methods proposed in [15] and [21].
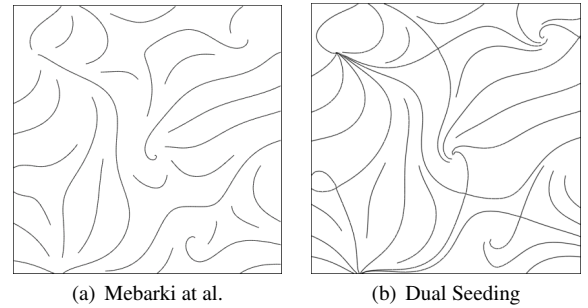


(a) Mebarki at al.　　　　(b) Dual Seeding

Figure 12: Comparison at low streamline density. Our streamline termination algorithm is less strict, allowing streamlines to gather at critical points, which improves their perceptibility.

later on are terminated at farther distances. In effect, critical points are more accentuated at low streamline density, as Fig. 12 demonstrates. This accentuation also improves the perceptibility of critical points compared to the results of Jobard/Lefer and Turk/Banks in Fig. 5. Thereby, our algorithm prioritizes feature coverage over uniformity. Apart from this desirable deviation of density, some undesirable clutter that does not provide additional information remains, as can be observed in Fig. 6 (left). Since streamlines can only be terminated at intersection points, where inter-streamline distance is again approximated, clutter is significantly reduced but not fully eliminated.

By construction, our seed locations on the dual streamlines have similarities to the seeding templates of Verma et al. shown in Fig. 13. The seeding templates run orthogonal to idealized (linear) flow fields in the vicinity of the critical points, and are constructed explicitly for every critical point of the vector field. Due to our topology based initialization and subsequent refinement, seeding on the dual streamlines results in the same emphasis of critical points without the assumption of idealized flow patterns.

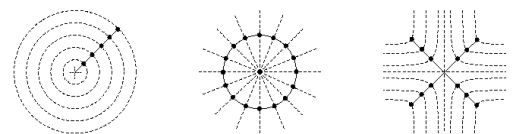Since the two streamline sets mutually control their densities, it



Figure 13: Seeding templates proposed by Verma et al.

is important to initialize $D$ first. During the initialization of $D$, $P$ is empty, and thus the streamline termination criterion never applies, resulting in very long lines that may clutter in the vicinity of critical points or loop, as can be seen in Fig. 1 and Fig. 3. Yet, loops in primal streamlines are detected, as they would intersect the same dual streamline twice. This initialization order ensures that the streamline result is free of narrow spirals that are perceived as distracting clutter.

As presented in the performance comparisons in Tab. 1 and Fig. 11, our algorithm is very efficient and has a moderate growth rate with respect to the streamline density. Efficiency of intersection detection and handling is crucial for our algorithm. We achieved good performance by attaching an index structure to the surface to facilitate intersection detection. The spatial resolution of the index structure can be assessed by the streamline density parameter $\delta$. This way, the search space for streamline intersections is restricted to a local neighborhood, even for dense seedings.

Our method is especially tailored for seeding on arbitrarily shaped curved 2D surfaces and cannot directly be extended to a 3D seeding technique. We rely on the symmetry of the streamline duality. In 2D, the dual (or orthogonal) structure of a line is again a line and we can apply the same method for the addition of subsequent new streamlines. In 3D, the dual structure of a 1D streamline is a 2D surface; the symmetry of our method would be lost. Apart from the structural difference, it is not desirable to seed streamlines in 3D with the same objectives as in 2D, as mutual occlusion is the main problem to solve in 3D.

## 6 CONCLUSION AND FUTURE WORK

We have presented a novel seeding technique especially useful for tangent vector fields on curved surface domains. Unlike other approaches, our method is not based on Euclidean distance, Delaunay-triangulation, Voronoi-decomposition, or other concepts, which are hard to apply efficiently on curved surfaces.

Besides means of streamline tracing, only simple algorithmic concepts are used, which makes our method easy to implement. Still, it is highly efficient, comparable in placement quality and superior in processing time to state-of-the-art methods.

The main contribution of this work resides in the introduction of dual streamlines. By analyzing and selectively refining the resulting net of orthogonal streamlines we explore the domain and control streamline density efficiently.

This approach yields a new measurement for density distribution, i.e. long continuous streamline segments indicate cavities in the dual field. It could be to use that measurement as a basis to apply optimization approaches, such as the one employed by Turk and Banks, instead of complying to the greedy paradigm. This could improve overall uniformity at the cost of simplicity and efficiency.

### REFERENCES

[1] Y. Chen, J. D. Cohen, and J. H. Krolik. Similarity-guided streamline placement with error evaluation. In *IEEE Transactions on Visualization and Computer Graphics*, volume 13, pages 1448–1455, 2007.

[2] J. Helman and L. Hesselink. Representation and display of vector field topology in fluid flow data sets. *Computer*, 22(8):27–36, Aug 1989.

[3] B. Jobard and W. Lefer. Creating evenly-spaced streamlines of arbitrary density. In W. Lefer and M. Grave, editors, *Visualization in Scientific Computing '97. Proc. Eurographics Workshop in Boulogne-sur-Mer, France*, pages 43–56, Wien, New York, 1997. Springer Verlag.

[4] B. Jobard and W. Lefer. Unsteady flow visualization by animating evenly-spaced streamlines. *Computer Graphics Forum*, 19(3), 2000.

[5] B. Jobard and W. Lefer. Multiresolution flow visualization. In *WSCG (Posters)*, pages 34–35, 2001.

[6] U. Kertzscher, A. Berthe, L. Goubergrits, and K. Affeld. Particle image velocimetry of a flow at a vaulted wall. *Proc. Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 222(4):465–473, 2008.

[7] P. Kipfer, F. Reck, and G. Greiner. Local exact particle tracing on unstructured grids. *Computer Graphics Forum*, 22:133–142, 2003.

[8] R. S. Laramee, H. Hauser, H. Doleisch, B. Vrolijk, F. H. Post, and D. Weiskopf. The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum*, 23(2):203–221, 2004.

[9] L. Li, H.-H. Hsieh, and H.-W. Shen. Illustrative streamline placement and visualization. In *Proc. IEEE Pacific Visualization*, pages 79–86, 2008.

[10] L. Li and H.-W. Shen. Image-based streamline generation and rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):630–640, 2007.

[11] Z. Liu, R. J. Moorhead II, and J. Groner. An advanced evenly-spaced streamline placement algorithm. In *IEEE Transactions on Visualization and Computer Graphics*, volume 12, pages 965–972, 2006.

[12] X. Mao, Y. Hatanaka, H. Higashida, and A. Imamiya. Image-guided streamline placement on curvilinear grid surfaces. In *VIS '98: Proc. Conference on Visualization '98*, pages 135–142, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press.

[13] O. Mattausch, T. Theußl, H. Hauser, and E. Gröller. Strategies for interactive exploration of 3d flow using evenly-spaced illuminated streamlines. In *SCCG '03: Proc. 19th spring conference on Computer graphics*, pages 213–222, New York, NY, USA, 2003. ACM.

[14] A. Mebarki. http://www-sop.inria.fr/geometrica/team/Abdelkrim.Mebarki/Streamlines.html.

[15] A. Mebarki, P. Alliez, and O. Devillers. Farthest point seeding for efficient placement of streamlines. In *Proc. IEEE Visualization*, pages 479–486, 2005.

[16] 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*, 22(4):775–792, 2003.

[17] O. Rosanwo, C. Petz, S. Prohaska, H.-C. Hege, and I. Hotz. Dual streamline seeding - method and implementation. Technical report, Zuse Institute Berlin, 2008. ZIB-Report 08-49.

[18] T. Salzbrunn and G. Scheuermann. Flow structure based 3d streamline placement. In H.-C. Hege, K. Polthier, and G. Scheuermann, editors, *Topology-Based Methods in Visualization II*. Springer, to appear 2008.

[19] M. Schlemmer, I. Hotz, B. Hamann, F. Morr, and H. Hagen. Priority streamlines: A context-based visualization of flow fields. In K. Museth, T. Möller, and A. Ynnerman, editors, *Eurographics/ IEEE-VGTC Symposium on Visualization*, pages 227–234, 2007.

[20] G. Turk and D. Banks. Image-guided streamline placement. In *SIGGRAPH '96: Proc. 23rd annual conference on Computer graphics and interactive techniques*, pages 453–460, New York, NY, USA, 1996. ACM.

[21] V. Verma, D. Kao, and A. Pang. A flow-guided streamline seeding strategy. In *Proc. IEEE Visualization*, pages 163–170, 2000.

[22] Ye, Kao, and Pang. Strategy for seeding 3d streamlines. *Proc. Visualization*, 5:60, 2005.