Streak Lines as Tangent Curves of a Derived Vector Field

Tino Weinkauf and Holger Theisel

Abstract—Characteristic curves of vector fields include stream, path, and streak lines. Stream and path lines can be obtained by a simple vector field integration of an autonomous ODE system, i.e., they can be described as tangent curves of a vector field. This facilitates their mathematical analysis including the extraction of core lines around which stream or path lines exhibit swirling motion, or the computation of their curvature for every point in the domain without actually integrating them. Such a description of streak lines is not yet available, which excludes them from most of the feature extraction and analysis tools that have been developed in our community. In this paper, we develop the first description of streak lines as tangent curves of a derived vector field – the *streak line vector field* – and show how it can be computed from the spatial and temporal gradients of the flow map, i.e., a dense path line integration is required. We demonstrate the high accuracy of our approach by comparing it to solutions where the ground truth is analytically known and to solutions where the ground truth has been obtained using the classic streak line computation. Furthermore, we apply a number of feature extraction and analysis tools to the new streak line vector field including the extraction of cores of swirling streak lines and the computation of streak line curvature fields. These first applications foreshadow the large variety of possible future research directions based on our new mathematical description of streak lines.

Index Terms—Unsteady flow visualization, streak lines, streak surfaces, feature extraction.

1 Introduction

Flow fields play a vital role in many areas. Examples are burning chambers, turbomachinery and aircraft design in industry as well as blood flow in medicine. A common approach to visualize flows in real-world experiments is to continuously release dye or smoke from a constant position into the flow and examine its behavior: vortices can be observed as patterns of swirling flow. The resulting dye or smoke structures are streak lines, streak surfaces or streak volumes depending on whether the dye has been released from a point, line, or surface.

Recently, streak surfaces have gained attention in the visualization community and a number of contributions have been made to use them in interactive applications [3], compute them with high accuracy and surface quality [14], and to render them in a smoke-like manner to mimic real-world smoke experiments [32]. A generalized notion of streak lines has been used to explore flow features [38]. Level set methods have been applied to interactively visualize streak lines, surfaces and volumes using the dye metaphor [36, 5]. Adaptive streak line computation schemes on contemporary graphics hardware have been explored in [6]. These approaches have proven that streak lines and surfaces are valuable visualization tools for unsteady flows.

There is a clear difference in how we can handle streak lines (and surfaces) in contrast to stream lines and path lines. Whereas all three types of characteristic curves find their applications in integration-based visualization methods, it is only for stream and path lines that we can compute derived properties such as their curvature [28, 35] *without actually integrating* them. This gives rise to a number of feature extraction methods for stream and path lines: Sujudi and Haimes proposed a scheme to extract centers of swirling flow [27]. Peikert and Roth formulated the idea of Sujudi/Haimes using the Parallel Vectors operator and presented a fast and robust extraction technique [17]. Bauer et al. [1] and Theisel et al. [29] proposed different algorithms to track these centers over time in unsteady flows. Weinkauf et al. [34] extended this idea to path lines.

All these feature extraction and analysis methods describe the properties of stream and path lines by employing mathematical formulations that are purely based on the vector field and its derivatives. The integral curves themselves are not required. This is due to the fact that we can describe stream and path lines of an unsteady flow as tangent curves of certain vector fields (Section 2). However, this is not yet the case for streak lines and therefore, these feature extraction and analysis methods cannot be used to explore their properties.

This paper aims at extending the range of applicable techniques for streak lines by introducing the – to the best of our knowledge – first description of streak lines as tangent curves of a derived vector field (Section 3). We call this new vector field the *streak line vector field*: for a 2D time-dependent flow this turns out to be a 4D vector field, and for a 3D time-dependent flow it is a 5D vector field. We explore its properties and give a parametrization of streak lines that allows to uniquely address every single streak line in space-time.

The computation of the streak line vector field involves the spatial and temporal derivatives of the flow map and requires a dense path line integration. We carefully study the accuracy of our approach by comparing the streak lines obtained with our method to streak lines computed using analytic solutions and the classic streak line computation scheme (Section 4).

We present two applications of the streak line vector field. First, an interactive streak line and surface explorer which allows to investigate the space of all streak lines in a different way than it was possible before, including their visualization using LIC-like images (Section 5). Second, we apply and extend feature extraction methods to extract cores around which streak lines exhibit swirling motion as well as to analyze their velocity magnitude and curvature (Section 6).

Notation We consider a n-dimensional ($n = 2,3$) time-dependent vector field $\mathbf{v}(\mathbf{x},t)$ over the domain $D \times T$ where $D \subseteq \mathbb{R}^n$ is the spatial domain and T is a time interval. We write derived $(n+1)$-dimensional variables with a bar like $\bar{\mathbf{p}}$, and derived $(n+2)$-dimensional variables with a double bar like $\bar{\bar{\mathbf{q}}}$. All vectors throughout the paper are column vectors, we often omit the explicit $()^T$ notation.

2 Characteristic Curves of Vector Fields

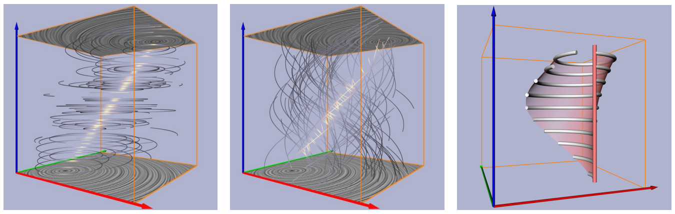
Let $\mathbb{E}^n$ be the n-dimensional Euclidean space. A curve $L \subset \mathbb{E}^n$ is called a *tangent curve* of a n-dimensional vector field $\mathbf{v}(\mathbf{x})$, if for all points $\mathbf{p} \in L$ the tangent vector of L coincides with $\mathbf{v}(\mathbf{p})$. Tangent curves are the solutions of the autonomous ODE system

$$\frac{d}{d\tau}\mathbf{x}(\tau) = \mathbf{v}(\mathbf{x}(\tau)) \quad \text{with} \quad \mathbf{x}(0) = \mathbf{x}_0. \tag{1}$$

For all points $\mathbf{x} \in \mathbb{E}^n$ with $\mathbf{v}(\mathbf{x}) \neq \mathbf{0}$, there is one and only one tangent curve through it. Tangent curves do not intersect or join each other.

- *Tino Weinkauf is with Courant Institute of Mathematical Sciences, New York University, E-mail: weinkauf@courant.nyu.edu.*
- *Holger Theisel is head of the Visual Computing Group at University of Magdeburg, E-mail: theisel@isg.cs.uni-magdeburg.de.*

Manuscript received 31 March 2010; accepted 1 August 2010; posted online 24 October 2010; mailed on 16 October 2010.
For information on obtaining reprints of this article, please send email to: tvcg@computer.org.



(a) Tangent curves of $\bar{\mathbf{s}}$ correspond to the stream lines in $\mathbf{v}$. See eq. (6).

(b) Tangent curves of $\bar{\mathbf{p}}$ correspond to the path lines in $\mathbf{v}$. See eq. (5).

(c) Streak lines (gray tubes) as intersections of a path surface (red) with $t = const$. hyperplanes.

Figure 1. Characteristic curves of a simple 2D time-dependent vector field shown as illuminated field lines (stream and path lines) or gray tubes (streak lines). The red/green coordinate axes denote the (x,y)-domain, the blue axis shows time. From [30] with permission.

Hence, tangent curves uniquely describe the directional information and are therefore an important tool for visualizing vector fields. The tangent curves of a steady vector field $\mathbf{v}(\mathbf{x})$ are called *stream lines*. A stream line describes the path of a massless particle in $\mathbf{v}$.

In a time-dependent vector field $\mathbf{v}(\mathbf{x},t)$ there are four types of characteristic curves: stream lines, path lines, streak lines and time lines. We concentrate on the first three for the rest of this paper. In a space-time point $(\mathbf{x}_0,t_0)$ we can start a *stream line* (staying in time slice $t = t_0$) by integrating

$$\frac{d}{d\tau}\mathbf{x}(\tau) = \mathbf{v}(\mathbf{x}(\tau),t_0) \quad \text{with} \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{2}$$

or a *path line* by integrating

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{v}(\mathbf{x}(t),t) \quad \text{with} \quad \mathbf{x}(t_0) = \mathbf{x}_0. \tag{3}$$

Path lines describe the trajectories of massless particles in time-dependent vector fields. The ODE system (3) can be rewritten as an autonomous system at the expense of an increase in dimension by one, if time is included as an explicit state variable:

$$\frac{d}{dt}\begin{pmatrix}\mathbf{x}\\ t\end{pmatrix} = \begin{pmatrix}\mathbf{v}(\mathbf{x}(t),t)\\ 1\end{pmatrix} \quad \text{with} \quad \begin{pmatrix}\mathbf{x}\\ t\end{pmatrix}(0) = \begin{pmatrix}\mathbf{x}_0\\ t_0\end{pmatrix}. \tag{4}$$

In this formulation space and time are dealt with on equal footing. Path lines of the original vector field $\mathbf{v}$ in ordinary space now appear as tangent curves of the vector field

$$\bar{\mathbf{p}}(\mathbf{x},t) = \begin{pmatrix}\mathbf{v}(\mathbf{x},t)\\ 1\end{pmatrix} \tag{5}$$

in space-time. To treat stream lines of $\mathbf{v}$, one may simply use

$$\bar{\mathbf{s}}(\mathbf{x},t) = \begin{pmatrix}\mathbf{v}(\mathbf{x},t)\\ 0\end{pmatrix}. \tag{6}$$

Figure 1 illustrates $\bar{\mathbf{s}}$ and $\bar{\mathbf{p}}$ for a simple example vector field $\mathbf{v}$. It is obtained by a linear interpolation over time of two bilinear vector fields.

The above space-time formulations for stream and path lines are powerful mathematical tools that facilitate the analysis of spatio-temporal features. Theisel et al. [30] use these formulations to develop tools for stream line and path line oriented topology. Weinkauf et al. [34] devise a criterion for finding the centers of swirling path lines by exploiting (5). Furthermore, the space-time formulations for stream and path lines allow them to introduce a unified notation of swirling motion in steady and unsteady flows.

Such a powerful formulation is not readily available for streak lines as we will see in the following.

A *streak line* is the connection of all particles set out at different times but the same point location. In an experiment, one can observe these structures by constantly releasing dye into the flow from a fixed position. The resulting streak line consists of all particles which have been at this fixed position sometime in the past. Considering the vector field $\bar{\mathbf{p}}$ introduced above, streak lines can be obtained in the following way: apply a path surface[1] integration in $\bar{\mathbf{p}}$ where the seeding curve is a straight line segment parallel to the t-axis, a streak line is the intersection of this path surface with a hyperplane perpendicular to the t-axis (Figure 1c).

Streak lines fail to have a property of stream and path lines: they are not locally unique in space-time, i.e., for a particular location and time there is more than one streak line passing through. As we show in the following section, we need a $(n+2)$-dimensional space to achieve this property for streak lines. Also note, that streak lines coincide with stream and path lines for steady vector fields $\mathbf{v}(\mathbf{x},t) = \mathbf{v}(\mathbf{x},t_0)$ and are described by (1) in this setting.

3 Streak Lines as Tangent Curves

The constructive description of streak lines as intersections of certain stream surfaces with a hyperplane is not suitable to examine their properties in a mathematical framework. In the following we develop a description of streak lines as tangent curves of a derived vector field that lends itself to mathematical analysis and leads to novel approaches for feature extraction as we will see in later sections.

3.1 Flow maps and their derivatives

To describe streak lines, we use the concept of flow maps and its derivatives. The flow map $\phi : D \to D$ describes the spatial location of a particle seeded at $(\mathbf{x},t)$ and integrated over a time interval τ, denoted as $\phi_t^\tau(\mathbf{x}) = \phi(\mathbf{x},t,\tau)$. As a side note, the computation of Finite Time Lyapunov Exponents (FTLE) [9, 7, 20] is essentially based on the consideration of the (spatial) gradient of ϕ. In fact, $\nabla\phi_t^\tau(\mathbf{x}) = \frac{\partial\phi}{\partial\mathbf{x}}$ is a $n \times n$ matrix describing the behavior of particles sent out in a small spatial neighborhood of $\mathbf{x}$.

For the consideration of streak lines, we additionally need the temporal partial derivative $\frac{\partial\phi}{\partial t}$ of ϕ which describes the behavior of particles sent out in the same spatial location but slightly before or after $(\mathbf{x},t)$. To study its properties, we compute the $(n+1)$-dimensional flow function $\bar{\phi}$ of $\bar{\mathbf{p}}$ which is defined as

$$\bar{\phi} : D \times T \to D \times T\ , \quad \bar{\phi}(\mathbf{x},t,\tau) = \bar{\phi}_t^\tau(\mathbf{x}) = \begin{pmatrix}\phi_t^\tau(\mathbf{x})\\ t+\tau\end{pmatrix}. \tag{7}$$

Then the gradient of $\bar{\phi}$ can be expressed as the $(n+1)\times(n+1)$ matrix

$$\nabla\bar{\phi}(\mathbf{x},t,\tau) = \begin{pmatrix} \nabla\phi & \frac{\partial\phi}{\partial t} \\ 0\,..\,0 & 1 \end{pmatrix}. \tag{8}$$

The fact, that the last component of $\bar{\mathbf{p}}$ is 1, ensures that the last line of $\nabla\bar{\phi}$ is $(0,..,0,1)$.

3.2 Description of Streak Lines

We formulate the main property:

Theorem 1 *Given a time-dependent vector field $\mathbf{v}(\mathbf{x},t)$ and its corresponding flow map $\phi_t^\tau(\mathbf{x})$, every streak line of $\mathbf{v}$ is a tangent curve of the $(n+2)$-dimensional vector field*

$$\bar{\bar{\mathbf{q}}}(\mathbf{x},t,\tau) = \begin{pmatrix} (\nabla\phi_t^\tau(\mathbf{x}))^{-1} \cdot \frac{\partial\phi_t^\tau(\mathbf{x})}{\partial t} + \mathbf{v}(\mathbf{x},t) \\ 0 \\ -1 \end{pmatrix} \tag{9}$$

and vice versa. We call $\bar{\bar{\mathbf{q}}}$ the streak line vector field. *It is defined in the domain $D \times T \times \Upsilon$ with $\tau \in \Upsilon$.*

[1] Note, that the extraction algorithms for path surfaces are commonly known as "stream surface algorithms" [10, 26, 8, 23].

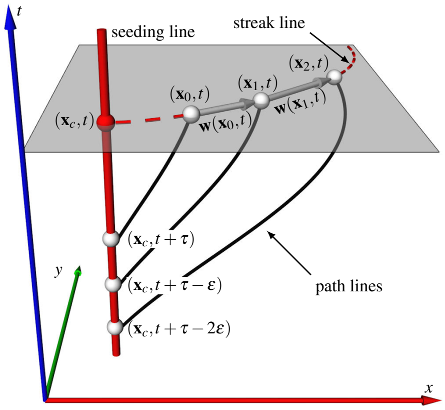


Figure 2. Definition of the vector field $\mathbf{w}(\mathbf{x},t,\tau)$, which is the main ingredient of the streak line vector field $\bar{\bar{\mathbf{q}}}$.

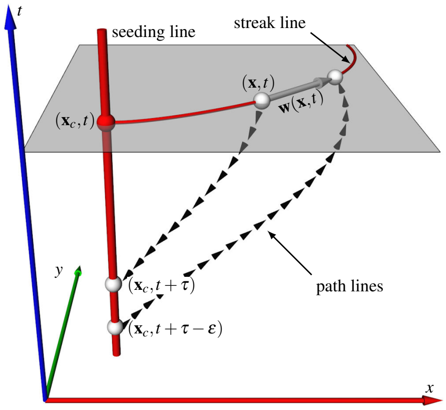


Figure 3. Straightforward, but costly way for computing $\mathbf{w}(\mathbf{x},t,\tau)$ with two consecutive path line integrations.

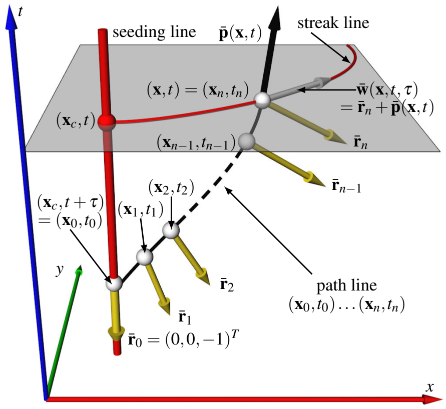

t
seeding line
$\bar{\mathbf{p}}(\mathbf{x},t)$
streak line
$(\mathbf{x},t)=(\mathbf{x}_n,t_n)$
$\bar{\mathbf{w}}(\mathbf{x},t,\tau)$
$=\bar{\mathbf{r}}_n+\bar{\mathbf{p}}(\mathbf{x},t)$
$(\mathbf{x}_c,t)$
$(\mathbf{x}_{n-1},t_{n-1})$
$\bar{\mathbf{r}}_n$
$\bar{\mathbf{r}}_{n-1}$
$(\mathbf{x}_2,t_2)$
$(\mathbf{x}_c,t+\tau)$
$=(\mathbf{x}_0,t_0)$
$(\mathbf{x}_1,t_1)$
path line
$(\mathbf{x}_0,t_0)\dots(\mathbf{x}_n,t_n)$
$\bar{\mathbf{r}}_2$
$\bar{\mathbf{r}}_1$
$\bar{\mathbf{r}}_0=(0,0,-1)^T$
y
x

Figure 4. Computing $\mathbf{w}(\mathbf{x},t,\tau)$ with a single path line integration by keeping track of the direction vector $\mathbf{r}_0$.

We show it only for the 2D time-dependent case, but it easily extends to 3D. Consider a seeding line as shown in Figure 2 which gives rise to a streak line at the time t. Given a point $(\mathbf{x}_0,t)$ on the streak line, we find its seeding location in space-time by integrating $\bar{\mathbf{p}}$ over the time interval τ until we reach the point $(\mathbf{x}_c,t+\tau)$ on the seeding line:[2]

$$(\mathbf{x}_c,t+\tau)=\bar{\phi}(\mathbf{x}_0,t,\tau). \tag{10}$$

The streak line passing through $(\mathbf{x}_0,t)$ is created by seeding particles at $\mathbf{x}_c$ at different times, i.e., on the seeding line. As already established, an integration of $\bar{\mathbf{p}}$ starting from $(\mathbf{x}_c,t+\tau)$ passes through $(\mathbf{x}_0,t)$. To see how the streak line evolves, we seed another particle at $(\mathbf{x}_c,t+\tau-\varepsilon)$ for a small $|\varepsilon|>0$ and integrate $\bar{\mathbf{p}}$ until we reach the time t in the point $(\mathbf{x}_1,t)$:

$$\mathbf{x}_1=\phi_{t+\tau-\varepsilon}^{-\tau+\varepsilon}(\mathbf{x}_c)=\phi_{t+\tau-\varepsilon}^{-\tau+\varepsilon}(\phi_t^{\tau}(\mathbf{x}_0)). \tag{11}$$

To come from $\mathbf{x}_0$ to $\mathbf{x}_1$ by a vector field integration, we define the two-parameter-dependent vector field $\mathbf{w}:\mathbb{R}^4\rightarrow\mathbb{R}^2$

$$\mathbf{w}(\mathbf{x},t,\tau)=\lim_{\varepsilon\rightarrow 0}\frac{\phi_{t+\tau-\varepsilon}^{-\tau+\varepsilon}(\phi_t^{\tau}(\mathbf{x}))-\mathbf{x}}{\varepsilon}. \tag{12}$$

In order to continue the streak line through $(\mathbf{x}_0,t)$ and $(\mathbf{x}_1,t)$, we start the integration of another particle at $(\mathbf{x}_c,t+\tau-2\varepsilon)$ and integrate $\bar{\mathbf{p}}$ until we reach the time t in the point $\mathbf{x}_2$:

$$\mathbf{x}_2=\phi_{t+\tau-2\varepsilon}^{-\tau+2\varepsilon}(\mathbf{x}_c). \tag{13}$$

Assuming a single Euler integration step,[3] we get

$$\begin{aligned}\mathbf{x}_1 &= \mathbf{x}_0+\varepsilon\cdot\mathbf{w}(\mathbf{x}_0,t,\tau) && (14)\\ \mathbf{x}_2 &= \mathbf{x}_1+\varepsilon\cdot\mathbf{w}(\mathbf{x}_1,t,\tau-\varepsilon). && (15)\end{aligned}$$

From this we can formulate the desired 4D vector field $\bar{\bar{\mathbf{q}}}$, with the property that streak lines of $\mathbf{v}$ are tangent curves of $\bar{\bar{\mathbf{q}}}$:

$$\bar{\bar{\mathbf{q}}}(\mathbf{x},t,\tau)=\begin{pmatrix}\mathbf{w}(\mathbf{x},t,\tau)\\0\\-1\end{pmatrix}. \tag{16}$$

It remains to show how to compute $\mathbf{w}$, which will lead us to equation (9) from theorem 1. A straightforward, but costly way is to integrate $\bar{\mathbf{p}}$ from $(\mathbf{x},t)$ over the time interval τ until $\mathbf{x}_c$, and then integrating back from $(\mathbf{x}_c,t+\tau-\varepsilon)$ until we reach the time t again (Figure 3). This involves two consecutive path line integrations.

A less expensive alternative – with a single path line integration – is to integrate in $\bar{\mathbf{p}}$ from $(\mathbf{x}_c,t+\tau)$ to $(\mathbf{x},t)$ while taking care of the direction $\bar{\mathbf{r}}_0=(0,0,-1)^T$. Figure 4 gives an illustration. We know that the change of the direction at a certain point is described by the Jacobian of the vector field. Given a point $(\mathbf{x}_i,t_i)$ and a direction vector $\bar{\mathbf{r}}_i$ there, the new point $(\mathbf{x}_{i+1},t_{i+1})$ and the new direction $\bar{\mathbf{r}}_{i+1}$ are obtained (in Euler integration) as

$$\begin{aligned}(\mathbf{x}_{i+1},t_{i+1}) &= (\mathbf{x}_i,t_i)+\delta_i\,\bar{\mathbf{p}}(\mathbf{x}_i,t_i) && (17)\\ \bar{\mathbf{r}}_{i+1} &= \bar{\mathbf{r}}_i\;+\;\delta_i\,\nabla\bar{\mathbf{p}}(\mathbf{x}_i,t_i)\cdot\bar{\mathbf{r}}_i && (18)\\ &= (I_3\;+\;\delta_i\,\nabla\bar{\mathbf{p}}(\mathbf{x}_i,t_i))\cdot\bar{\mathbf{r}}_i && (19)\end{aligned}$$

where I_3 is the 3×3 unit matrix and δ_i is the current step size. Given the path line from $(\mathbf{x}_c,t+\tau)$ to $(\mathbf{x},t)$ as a polygon $(\mathbf{x}_0,t_0),\ldots,(\mathbf{x}_n,t_n)$ with $(\mathbf{x}_0,t_0)=(\mathbf{x}_c,t+\tau)$ and $(\mathbf{x}_n,t_n)=(\mathbf{x},t)$, we compute

$$\begin{aligned}\bar{\mathbf{r}}_0 &= (0,0,-1)^T\\ \bar{\mathbf{r}}_1 &= (I_3+\delta_0\,\nabla\bar{\mathbf{p}}(\mathbf{x}_0,t_0))\cdot\bar{\mathbf{r}}_0\\ \bar{\mathbf{r}}_2 &= (I_3+\delta_1\,\nabla\bar{\mathbf{p}}(\mathbf{x}_1,t_1))\cdot\bar{\mathbf{r}}_1\\ &\ldots\\ \bar{\mathbf{r}}_n &= (I_3+\delta_{n-1}\,\nabla\bar{\mathbf{p}}(\mathbf{x}_{n-1},t_{n-1}))\cdot\bar{\mathbf{r}}_{n-1}\\ &= \prod_{i=n-1}^{0}(I_3+\delta_i\,\nabla\bar{\mathbf{p}}(\mathbf{x}_i,t_i))\cdot\bar{\mathbf{r}}_0 && (20)\end{aligned}$$

with $\delta_i=t_{i+1}-t_i$. Note that $\bar{\mathbf{r}}_0,\ldots,\bar{\mathbf{r}}_n$ all have a constant third component of -1 which is due to the fact that the last line of $\nabla\bar{\mathbf{p}}$ is always $(0,0,0)$. Finally, we get

$$\bar{\mathbf{w}}(\mathbf{x},t,\tau)=\bar{\mathbf{r}}_n+\bar{\mathbf{p}}(\mathbf{x},t). \tag{21}$$

Note that the last component of $\bar{\mathbf{w}}$ is 0, since the last components of $\bar{\mathbf{r}}_n$ and $\bar{\mathbf{p}}$ are -1 and 1 respectively. This is to be expected since a streak line lives in a $t=const.$ hyperplane. Removing the last component gives almost the desired $\mathbf{w}$, but as a last step, note that $\prod_{i=n-1}^{0}(I_3+\delta_i\,\nabla\bar{\mathbf{p}}(\mathbf{x}_i,t_i))$ corresponds to the gradient of the flow map (cf. [13]):

$$\prod_{i=n-1}^{0}(I_3+\delta_i\,\nabla\bar{\mathbf{p}}(\mathbf{x}_i,t_i))=\nabla\bar{\phi}_{t+\tau}^{-\tau}(\mathbf{x}_c)=\left(\nabla\bar{\phi}_t^{\tau}(\mathbf{x})\right)^{-1}. \tag{22}$$

[2] The illustration in Figure 2 assumes negative τ-values, since it shows the part of the streak line that has been seeded at earlier time steps. The general concept works also for positive τ-values, which describe the part of the streak line seeded at future time steps.

[3] We use the Euler integration step here only as a mathematical tool to deduce the formula, which is justified since $\varepsilon\rightarrow 0$. This is unrelated to our practical computations, which are carried out using a Runge-Kutta integration scheme of 4th order with adaptive step size control.

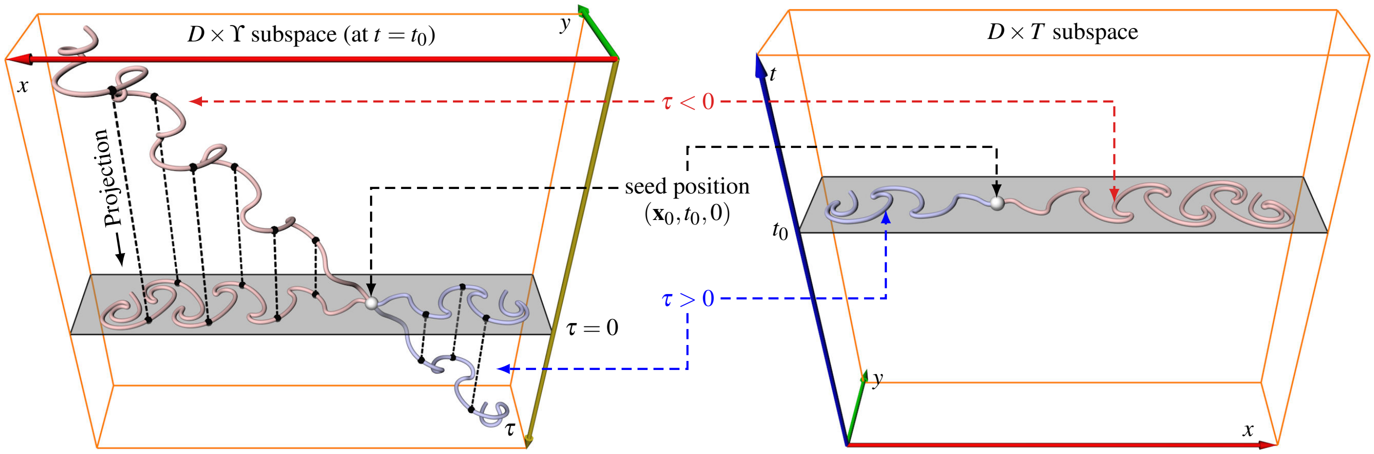


Figure 5. Integration of a streak line using the streak line vector field $\bar{\bar{\mathbf{q}}}$. The data set is a 2D unsteady flow behind a cylinder (explained in Section 5). Shown are the $(n+1)$-dimensional subspaces $D \times \Upsilon$ (left) and $D \times T$ (right). The streak line is seeded at $(\mathbf{x}_0, t_0, 0)$ and integrated forward (red) and backward (blue) in $\bar{\bar{\mathbf{q}}}$. The integration stays in t_0 and can therefore be performed in the $D \times \Upsilon$ subspace (left). A projection of the resulting tangent curve into the spatial domain yields the desired streak line.

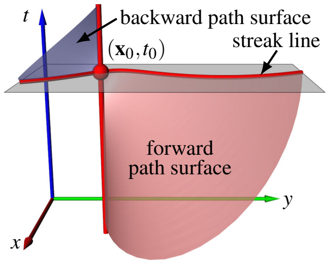


Figure 6. Streak line parametrization: every point in $D \times T$ is assigned with a unique streak line.

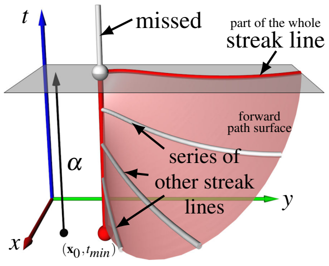


Figure 7. Addressing streak lines from $(\mathbf{x}_0, t_{min})$ misses the backward integrated part.

This yields the desired $\mathbf{w}$ as used in equation (9) from theorem 1:

$$\begin{aligned} \mathbf{w}(\mathbf{x}, t, \tau) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot \left(\nabla \bar{\phi}_t^{\tau}(\mathbf{x})\right)^{-1} \cdot \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} + \mathbf{v}(\mathbf{x}, t) \\ &= \left(\nabla \phi_t^{\tau}(\mathbf{x})\right)^{-1} \cdot \frac{\partial \phi_t^{\tau}(\mathbf{x})}{\partial t} + \mathbf{v}(\mathbf{x}, t). \qquad (23) \end{aligned}$$

3.3 Properties of the Streak Line Vector Field

3.3.1 Streak Line Parametrization

Parametrization is a well-studied concept for curves and surfaces, meaning to find an injective map from a subset of $\mathbb{R}^2/\mathbb{R}^3$ to a curve/surface. This map allows to uniquely address every curve/surface point.

For streak lines, the problem of parametrization has a simple solution: every point $(\mathbf{x}, t) \in D \times T$ can be assigned with a unique streak line and vice versa, i.e., it is a bijective map. We define: given a point $(\mathbf{x}_0, t_0) \in D \times T$, the streak line $\mathbf{S}(\mathbf{x}_0, t_0)$ is constructed as follows: do a path surface integration of $\bar{\mathbf{p}}$ starting from $\{\mathbf{x}_0\} \times T$ in forward and backward direction until it leaves the domain $D \times T$,[4] then intersect the resulting surface with the plane $t = t_0$. See Figure 6. Alternatively, $\mathbf{S}(\mathbf{x}_0, t_0)$ can be computed using $\bar{\bar{\mathbf{q}}}$ as explained in the next section.

Note that, considering the classic streak line computation scheme, one might be tempted to address a streak line by the earliest point $(\mathbf{x}_0, t_{min})$ on the seeding line and a time span α for the integration. The classic computation scheme will then go through a series of other streak lines until it reaches the addressed streak line in $t_{min} + \alpha$. However, this describes only the part of the streak line that can be accessed in forward integration, but it misses the part that can only be reached by backward integration (Figure 7).

To see that backward integration makes indeed sense for streak lines, consider a steady vector field: here, streak lines coincide with stream lines for which the notion of backward integration is very well accepted.

3.3.2 Seeding, Integration and Projection

The streak line vector field $\bar{\bar{\mathbf{q}}}(\mathbf{x}, t, \tau)$ can be used as follows to integrate the streak line $\mathbf{S}(\mathbf{x}_0, t_0)$:

1. Seed at $(\mathbf{x}_0, t_0, 0)$. This uniquely addresses the streak line as described in the previous section.

2. Integrate $\bar{\bar{\mathbf{q}}}$ in forward and backward direction. Since the integration will stay in t_0, it suffices to keep the $n+1$-dimensional subspace $D \times \Upsilon$ in memory.

3. Project the resulting tangent curve into $\mathbb{R}^n$ by removing its t and τ coordinates.

Figure 5 illustrates this by showing the integration in $D \times \Upsilon$ and $D \times T$ for a 2D unsteady flow behind a cylinder.

3.3.3 Performance of Computing Streak Lines

Integrating a streak line in $\bar{\bar{\mathbf{q}}}$ amounts to a simple tangent curve integration, which has a time complexity of $O(n)$ with n being the number of sample points on the streak line. The classic streak line computation method, on the other hand, amounts to a path surface integration, which has a time complexity of $O(n^2)$. Hence, our new method allows to obtain streak lines[5] significantly faster and therefore, it lends itself to a new interaction metaphor: the interactive exploration of the space of streak lines. We will discuss this in more detail in Section 5.

This is not to say that the classic approach could not be used to explore streak lines, but integrating $\bar{\bar{\mathbf{q}}}$ is simply faster and we feel, that it also introduces a new viewpoint on the whole issue of interactive streak line and surface visualization. Of course, we pay for the lower time complexity with higher memory costs and the time for precomputing $\bar{\bar{\mathbf{q}}}$ as detailed in Section 4.

[4] More precisely, it suffices to integrate forward from $\{\mathbf{x}_0\} \times \{t < t_0\}$ in $D \times \{t < t_0\}$ and backward from $\{\mathbf{x}_0\} \times \{t > t_0\}$ in $D \times \{t > t_0\}$, since all other path lines will not cross the plane $t = t_0$.

[5] This is also true for streak surfaces in 3D time-dependent flows: the new method amounts to a stream surface integration, the classic method to a stream volume integration.

3.3.4 Streak Surfaces

Streak surfaces are a set of streak lines seeded from a curve in D. They are typically computed by constantly releasing particles from the seeding curve and integrating *all* released particles along the flow field. During integration the distance between neighboring particles may become too large/small due to diverging/converging flow behavior, which requires the insertion/removal of particles including an update of the mesh connectivity to stay within given resolution and quality constraints. In principal, the *whole surface* has to be checked for its compliance with the constraints after every integration step. Only recently, a number of solutions have been presented to deal effectively and efficiently with this problem [32, 14, 3].

Given our new streak line vector field, we are able to compute the streak surfaces of $\mathbf{v}$ as stream surfaces of $\bar{\bar{\mathbf{q}}}$. Hence, algorithms originally designed for stream/path surfaces [10, 26, 8, 23] can now be applied to compute streak surfaces. The advantage is that one only needs to check the *front line* of the stream surface for insertion/removal of tracers during integration. Figure 12 from Section 5 shows an example.

3.3.5 Relation between FTLE and Streak Line Vector Field

Equation (23) shows that there is no obvious relation between FTLE and streak lines: FTLE is focusing only on the spatial gradients of ϕ while the streak line vector field also uses their time derivatives. To see that this is indeed a difference, consider the example vector field $\mathbf{v}(\mathbf{x},t) = (\cos(t), \sin(t))^T$. It has a zero flow map gradient

$$\nabla\phi \equiv \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{24}$$

while the time-derivative has the following closed form solution, which is generally non-zero

$$\frac{\partial \phi_t^{\tau}(\mathbf{x})}{\partial t} = \begin{pmatrix} \cos(\tau) \\ \sin(\tau) \end{pmatrix}. \tag{25}$$

4 Implementation and Evaluation

4.1 Implementation

We discuss the implementation here only for the 2D time-dependent case, but it extends to 3D in a straightforward fashion. We compute the streak line vector field on a $n_x \times n_y \times n_T \times n_\tau$ 4D grid, where (n_x, n_y, n_T) denotes the grid resolution of the original flow and $n_\tau = n_T$, as follows:

1. For every grid point $(x_i, y_i, t_i, 0)$ of $\bar{\bar{\mathbf{q}}}$, seed 6 path lines in an ε-neighborhood $(x_i \pm \varepsilon, y_i, t_i, 0)$, $(x_i, y_i \pm \varepsilon, t_i, 0)$, $(x_i, y_i, t_i \pm \varepsilon, 0)$.

2. Integrate these path lines using $\bar{\mathbf{p}}$ until they leave the domain $D \times T$.

3. Intersect these path lines with every time step t_j and compute $\mathbf{w}$ as given in (23). Assign the result to the grid point (x_i, y_i, t_i, τ_j) where $\tau_j = t_i - t_j$. If one of the path lines left the domain before reaching t_j, denote this grid point to be invalid (e.g., assign zeros). This applies in particular to all boundary grid points since some of their seeds are already outside of the domain.

It suffices to save the first two components of $\bar{\bar{\mathbf{q}}}$ (which correspond to $\mathbf{w}$) and add its constant third and fourth component later on-the-fly during streak line integration.

4.2 Numerical Accuracy

We evaluated the accuracy of the streak line integration in $\bar{\bar{\mathbf{q}}}$ using a 2D and a 3D flow around a cylinder as well as an analytic vector field. The analytically given, linear 2D time-dependent vector field

$$\mathbf{v}(\mathbf{x},t) = \begin{pmatrix} -(1-t)y - ty \\ (1-t)(x-y) + t(x-y-1) \end{pmatrix} \tag{26}$$

contains a focus critical point moving along the x-axis. Since it is of such a simple nature, we can find a closed solution for $\bar{\bar{\mathbf{q}}}$:

$$\bar{\bar{\mathbf{q}}}(\mathbf{x},t,\tau) = \begin{pmatrix} -1 - y - e^{\frac{\tau}{2}}\left(\frac{1}{3}\sin(\frac{\tau}{2}\sqrt{3})\sqrt{3} + \cos(\frac{\tau}{2}\sqrt{3})\right) \\ x - y - t - \frac{2}{3}e^{\frac{\tau}{2}}\sin(\frac{\tau}{2}\sqrt{3})\sqrt{3} \\ 0 \\ -1 \end{pmatrix}, \tag{27}$$

which we use to compute the ground truth for the streak lines. Furthermore, we compute $\bar{\bar{\mathbf{q}}}$ using the algorithm described above (not by sampling (27)) with different grid resolutions and compare a streak line integrated in this field to the ground truth. We consider the domain $D = [-2.5, 2]^2$ and $T = [0, -10]$ and seed the streak line at $(1.5, 1.5, 0, 0)$. The blue line (——) in Figure 9a shows the distance to the ground truth for a grid resolution with a voxel size of $d_x = d_y = d_{tau} = 0.1$ ($45 \times 45 \times 100$ grid points). More precisely, we show the relative spatial distance with respect to the diagonal of the spatial bounding box, i.e., the absolute distance to the ground truth divided by the length of the diagonal. This distance is plotted over the integration time $\tau = [0, -8]$, which has been normalized in the plot to the interval $[0, 1]$. As it is evident in Figure 9a, the error stays around and below 10^{-4} even for this rather coarse resolution. This shows that our computation scheme for $\bar{\bar{\mathbf{q}}}$ produces reliable results. Note that the slight decrease of the distance with higher τ-values is due to the attracting nature of the focus critical point.

In Figure 9b we computed the relative Hausdorff distance (with respect to the diagonal of the spatial bounding box) between the integrated streak line and the ground truth for different resolutions of $\bar{\bar{\mathbf{q}}}$. We used voxel sizes between $d_x = d_y = d_{tau} = [0.05, 0.5]$. The x-axis in Figure 9b reflects them as normalized resolution $Res = 1 - \frac{d - d_{min}}{d_{max} - d_{min}}$, where 0 corresponds to the coarsest and 1 to the finest resolution. As expected, the Hausdorff distance to the ground truth decreases with increasing resolution and already medium resolutions yield satisfying results.

We found that ε, i.e., the perturbation around the grid point for the flow map computation, greatly influences the accuracy of the streak line vector field. Our first approach was to seed the path lines (step 1 of the algorithm) only at the grid points and use neighboring path lines to compute the flow map. While this reduces the costs for the dense path line integration by a factor of 6, it leads to less accurate results. In general, we use a value for ε that is about an order of magnitude smaller than the voxel size. This yields reliable results even for long streak line integrations and allows to use the same grid resolution for $\bar{\bar{\mathbf{q}}}$ that the original data set is given in – as shown in the following.

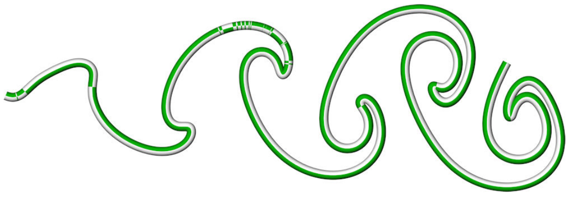
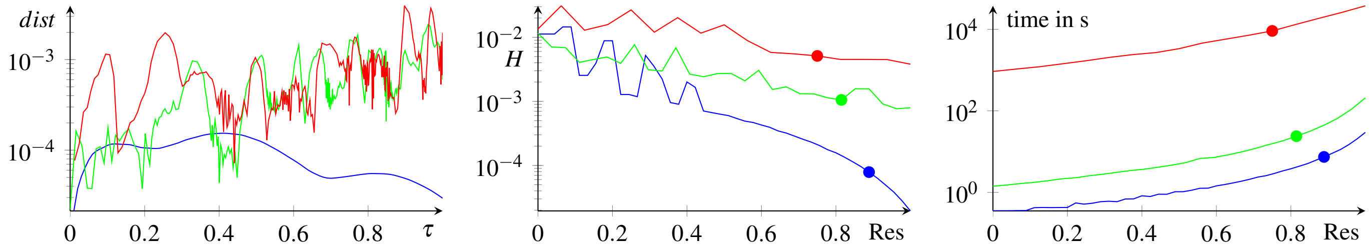


Figure 8. 2D cylinder data set. Visual comparison between the integrated streak line (green) and the ground truth (white). Their point-wise distance is plotted in Figure 9a, their Hausdorff distance in 9b (●).

The green lines (——) in Figure 9 represent a simulated data set of a 2D time-dependent flow behind a cylinder (simulated using the Free Software *Gerris Flow Solver* [18]). A visualization is shown in Figure 10. We computed the ground truth using the classic streak line computation scheme, i.e., by intersecting a path surface of very high resolution with a t-hyperplane. In Figure 9a we plotted the distance to the ground truth for the original grid resolution (see also Table 1) over the integration time $\tau = [0, -2]$. While the distance grows with longer integration times, the error stays below or around 10^{-3}. This is also evident in the visual comparison between the integrated streak line and the ground truth (Figure 8).



(a) Relative distance to the ground truth over integration time.

(b) Relative Hausdorff distance to the ground truth for different resolutions of $\bar{\bar{\mathbf{q}}}$.

(c) Time for computing $\bar{\bar{\mathbf{q}}}$ for a single time step over different resolutions.

Figure 9. Evaluation of the numerical accuracy and pre-computation times using the analytic test case from (26) (——), the 2D cylinder data set (——), and the 3D square cylinder (——). The dots (●●●) represent the resolutions used in Figure 9a and correspond to the original resolutions for the cylinder data sets. The values on the x-axes have been normalized to $[0,1]$ as described in the text.

We also computed the Hausdorff distance for resolutions of $\bar{\bar{\mathbf{q}}}$ with a voxel size in the interval $[3\cdot 10^{-3}, 3\cdot 10^{-2}]$, which is below and above the original voxel size of $8\cdot 10^{-3}$ (Figure 9b). It turns out that even coarser resolutions yield results below 10^{-2}.

The red lines (——) in Figure 9 represent the 3D time-dependent flow around a confined square cylinder (visualized in Figure 12). This is a direct numerical Navier Stokes simulation by Simone Camarri and Maria-Vittoria Salvetti (University of Pisa), Marcelo Buffoni (Politecnico of Torino), and Angelo Iollo (University of Bordeaux I) [4] which is publicly available [11]. It is an incompressible solution with a Reynolds number of 200 and the square cylinder has been positioned symmetrically between two parallel walls. The flow has periodic boundary conditions in spanwise direction.

To evaluate the accuracy of our computations, we used an integration time $\tau = [0, -18]$ (yielding a streak line which is 1.7 times longer than the diagonal of the spatial bounding box) and voxel sizes in the interval $[0.09, 0.25]$ (original voxel size is 0.13). The results are plotted in Figures 9a-b and show low error values similar to the other data sets.

4.3 Memory Requirements

Assuming the time-dependent vector field $\mathbf{v}(\mathbf{x},t)$ with n_T time steps requires the amount of memory M, then the streak line vector field $\bar{\bar{\mathbf{q}}}$:

- takes the same amount of memory M if we stay within a given time step t_0, which allows to explore all streak lines in that time step,
- requires $n_T \cdot M$ memory to cover all time steps.

For all the data sets used in this paper, we found it sufficient to use the same resolution for $\bar{\bar{\mathbf{q}}}$ that the original data set is given in, i.e., it yields reliable results even for long streak line integrations (see also Section 4.2). We sample $\bar{\bar{\mathbf{q}}}$ on a uniform grid in our current implementation. However, as discussed in Section 4.1, the streak line vector field is not defined for high τ-values in some parts of the domain, since the path lines left them earlier during the computation of $\bar{\bar{\mathbf{q}}}$. Hence, it might be beneficial to use adaptive meshes such as an octree to represent $\bar{\bar{\mathbf{q}}}$. This would reduce the memory costs without sacrificing the accuracy. Table 1 shows the memory required to store the streak line vector field for the 2D and 3D cylinder flows.

4.4 Time for Computing the Streak Line Vector Field

Computing $\bar{\bar{\mathbf{q}}}$ for a time step of the 2D cylinder data set shown in Figure 10 took 52 seconds single-threaded on a laptop with an Intel Core 2 Duo T9550 (2.66GHz). Doing the same for a 3D time-dependent flow around a squared cylinder shown in Figure 12 took 160 minutes wall clock time. The times for computing all time steps are detailed in Table 1. The time for computing $\bar{\bar{\mathbf{q}}}$ increases linearly with the number of grid points, since the algorithm integrates and evaluates a single path line for every grid point. Since the number of grid points has an exponential relation to the resolution (voxel size), we see exponential curves in Figure 9c depicting the computation times for different resolutions. Since most of the time is spent with computing the flow map, it seems reasonable to apply faster approximation schemes such as the one discussed in [2]. We leave it to future work to investigate this in detail.

Table 1. Time and memory needed to compute and store $\bar{\bar{\mathbf{q}}}$. The first row for a data set represents $\bar{\bar{\mathbf{q}}}$ for a single time step, the second row for all time steps.

Data set	Resolution (spatial) $\times\tau\times t$	Comp. Time in minutes	Memory in MB
Cylinder 2D	$(338\times 100)\times 250\times 1$	0.87	64.5
	$(338\times 100)\times 250\times 250$	217	16125
Square Cylinder 3D	$(192\times 64\times 48)\times 102\times 1$	160	688.5
	$(192\times 64\times 48)\times 102\times 102$	16320	70227

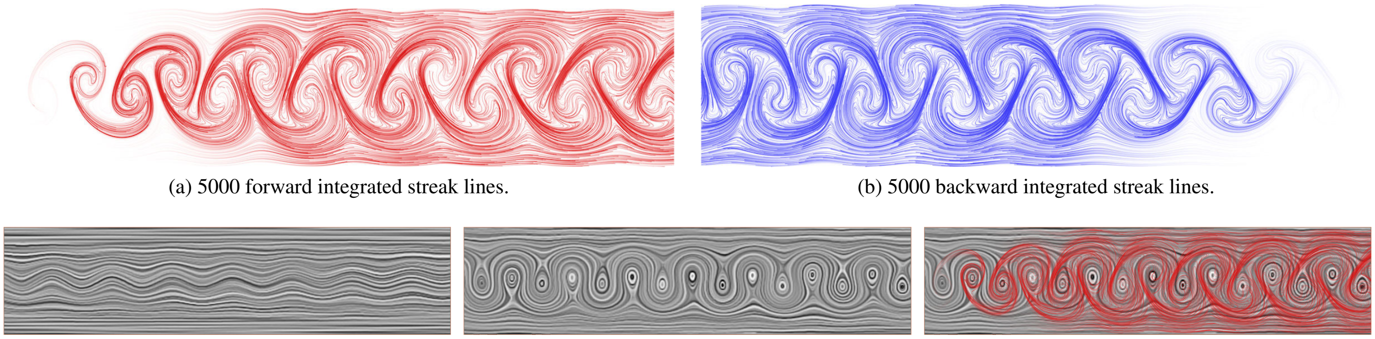
5 Exploration of Streak Lines and Surfaces

Our approach lends itself to a new interaction metaphor: given a spatio-temporal seeding location $(\mathbf{x}_0, t_0)$, the so-defined streak line can be computed more or less instantly, whereas the classic approach requires to develop the streak line over the one-parameter family of streak lines defined by $\{(\mathbf{x}_0, t_{min})\ldots(\mathbf{x}_0, t_0)\}$ in forward integration and $\{(\mathbf{x}_0, t_0)\ldots(\mathbf{x}_0, t_{max})\}$ in backward integration. See also Section 3.3.3. Hence, our approach allows for a faster exploration of the space of streak lines. Figures 10a-b show 5000 forward and backward integrated streak lines of the 2D time-dependent flow behind a cylinder. They have been computed in under a second in the τ-interval $[0, \pm 2.5]$. The pre-computation time for $\bar{\bar{\mathbf{q}}}$ is 52 seconds (see Table 1). To compute the same set of streak lines with the classic approach requires 167 minutes. This clearly demonstrates one of the advantages of integrating streak lines in $\bar{\bar{\mathbf{q}}}$.

Figure 10 shows a part of the domain away from the cylinder where the well-known von Kármán vortex street is well developed. The stream lines of the same time step do not reveal the patterns of the vortex street in the original frame of reference (Figure 10c), but they do after removing the ambient part of the flow (Figure 10d).[6] Figure 10e shows an overlay of the LIC texture and the forward integrated streak lines. It clearly shows, that the structures formed by the streak lines match with the von Kármán vortex street.

Figure 11 shows a LIC visualization of the first two components of $\bar{\bar{\mathbf{q}}}$ at different τ-values for the cylinder data set. These two components correspond to $\mathbf{w}$ (Equation (23)). Note that the LIC does not show streak lines, but rather the local direction of the streak lines at a given τ. One can think of the depicted lines as being built up of instantaneous streaklets. The vortices of the von Kármán vortex street are revealed by this: longer integrated parts of a streak line roll up in

[6] The path lines of this flow do not reveal the vortex street, neither in the original frame of reference nor with removed ambient part. Furthermore, they do not stay in a time step. Therefore, we left them out of this comparison.



(a) 5000 forward integrated streak lines.

(b) 5000 backward integrated streak lines.

(c) Stream lines in the same spatial domain and at the same time step as the streak lines.

(d) Stream lines with removed ambient flow reveal the von Kármán vortex street.

(e) The patterns of the streak lines match with the von Kármán vortex street.

Figure 10. Streak lines and stream lines of a time step in the 2D time-dependent flow around a cylinder. The streak lines are rendered such that they become less opaque the closer they get to their seeding position at $\tau = 0$.

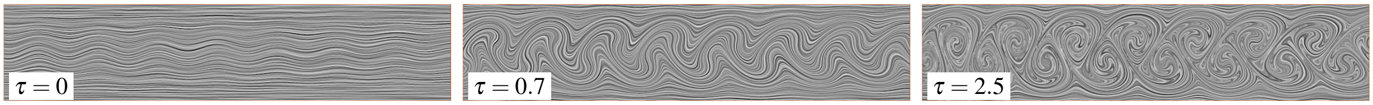


Figure 11. With increasing τ, the first two components of $\bar{\bar{\mathbf{q}}}$ visualized using LIC reveal the von Kármán vortex street of the cylinder data set.

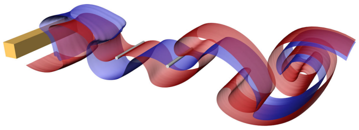


Figure 12. Two streak surfaces in the flow behind a square cylinder; integrated forward and backward starting from the gray lines. Streak surfaces of a 3D flow intersect each other just as streak lines do in 2D.

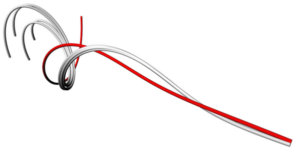


Figure 13. Streak line core (red) of the vector field (26) in the center of spiraling streak lines (gray). Shown is the domain $D \times \Upsilon$ at $t = 0$.

the vortices of this flow as shown in e.g. Figure 8. Hence, the streak lines have a stronger rotational behavior for larger τ-values, which is encoded in $\mathbf{w}$ (the only non-constant components of $\bar{\bar{\mathbf{q}}}$) and exposed in the LIC visualization of Figure 11. Note that following (23), $\mathbf{w}$ corresponds to $\mathbf{v}(\mathbf{x},t)$ for $\tau = 0$, which is also evident in the LIC visualization (compare with Figure 10c).

Figure 12 shows two streak surfaces in the flow around the square cylinder. They have been integrated in forward and backward direction from the depicted seeding lines. The computation time for one of them was one second once the 4D time-slice $D \times \Upsilon$ had been loaded into main memory. Since the streak surfaces show their most intricate structures the longer they have been integrated, the classic approach would require much longer computation times to yield the same result. But more importantly, when using the streak line vector field, the user is able to manipulate the seeding line and gets almost instant feedback in the form of a fully developed streak surface. This way, our new approach provides an interesting, orthogonal alternative to the classic variant: whereas the classic computation scheme focuses on showing the evolution[7] of streak surfaces, the streak line vector field provides quasi-instant results at any given time step.[8]

6 Feature Extraction and Analysis for Streak Lines

The description of streak lines as tangent curves of a derived vector field allows us to apply feature extraction and analysis tools to streak lines that were previously only available for stream and path lines. In the following we show how to extract cores around which streak lines show a spiraling behavior, and we analyze the velocity magnitude and curvature of streak lines by computing derived scalar fields.

6.1 Cores of Swirling Streak Lines

Areas where characteristic curves of a flow exhibit a spiraling behavior are of great interest since they are usually associated with important flow features such as vortices. Sujudi and Haimes developed a method of 3D steady flows to extract core lines around which stream lines swirl [27]. The Parallel Vectors operator is often used to extract these features [17]. For 3D time-dependent flows, one may track these core lines over time [1, 29], which yields surfaces in the 4D space-time domain. However, since the motion in unsteady flows is governed by path lines, it seems reasonable to investigate their swirling motion as done by Weinkauf et al. [34]. This yields surfaces in the 4D space-time domain, too.

In this section we want to extend the same idea to streak lines and extract cores around which they show spiraling behavior. Since the geometry of streak lines is governed by the motion of particles, a spiraling behavior of streak lines indicates the presence of a vortex. For a 2D time-dependent flow, we have to expect surface structures in the 4D domain of the streak line vector field. For a 3D time-dependent flow, it will be volumes in 5D. Our derivation here follows the basic principles of [34] and for the sake of brevity we refer the interested reader to that paper for background information and a more detailed reasoning.

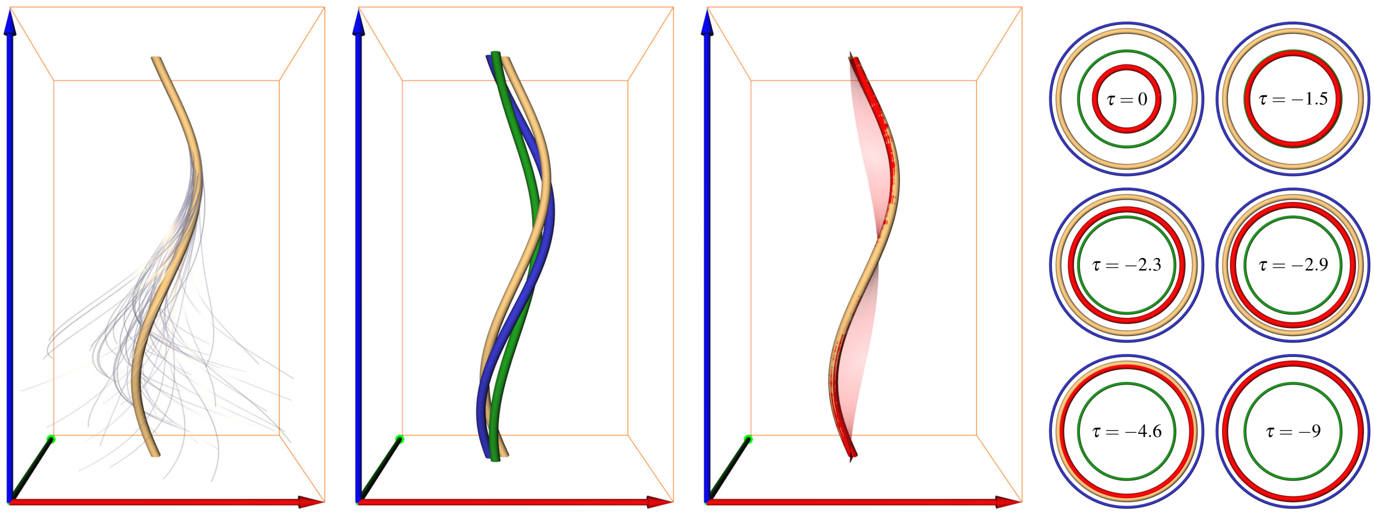
We consider the eigenanalysis of the gradient of the streak line vector field $\bar{\bar{\mathbf{q}}}$, which is a $(n+2) \times (n+2)$ matrix

$$\nabla\bar{\bar{\mathbf{q}}}(\mathbf{x},t,\tau) = \begin{pmatrix} \frac{\partial \mathbf{w}}{\partial \mathbf{x}} & \frac{\partial \mathbf{w}}{\partial t} & \frac{\partial \mathbf{w}}{\partial \tau} \\ 0\,..\,0 & 0 & 0 \\ 0\,..\,0 & 0 & 0 \end{pmatrix}. \qquad (28)$$

It has two eigenvalues of 0 with the corresponding eigenvectors $\bar{\bar{\mathbf{e}}}_t$, $\bar{\bar{\mathbf{e}}}_\tau$. The remaining n eigenvalues are the eigenvalues of $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$, its corresponding eigenvectors are $(\mathbf{e}_i, 0, 0)^T$ where $\mathbf{e}_i$ are the eigenvectors of $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$. Now we treat the cases $n = 2$ and $n = 3$ separately.

[7] In the sense of a series of streak surfaces similar to Figure 7.

[8] Note, however, that our approach can be used to visualize the evolution as well: apply a sequence of integrations in $\bar{\bar{\mathbf{q}}}$ from $(\mathbf{x}_0, t_{min})$ to $(\mathbf{x}_0, t_0)$. Similarly for backward integration. The computational effort is similar to the classic variant; except that our approach requires a long pre-processing time and comes with high memory costs. Hence, when showing the evolution of streak surfaces is the primary focus, the classic variant comes with a smaller footprint.



(a) Path lines approaching the attractor (yellow curve).

(b) Stream (green) and path line cores (blue) do not match the attractor.

(c) Streak line core (red curve and surface) matches the attractor.

(d) With decreasing τ, the streak line core converges. View from top.

Figure 14. The cores of swirling streak lines (red) detect the attractor (yellow) in the *Beads Problem* flow in contrast to the core lines of stream (green) and path lines (blue). The streak line core is a 4D surface shown in (c) together with its intersection at $\tau = -9$, which is a line in space-time that matches the attractor of this flow.

6.1.1 2D Time-Dependent Flows ($n = 2$)

In this case we get for the two eigenvectors corresponding to the 0 eigenvalues

$$\{\bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau\} = \left\{ \begin{pmatrix} -\det(\mathbf{w}_t,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_t)\\ \det(\mathbf{w}_x,\mathbf{w}_y)\\ 0 \end{pmatrix}, \begin{pmatrix} -\det(\mathbf{w}_\tau,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_\tau)\\ 0\\ \det(\mathbf{w}_x,\mathbf{w}_y) \end{pmatrix} \right\}. \tag{29}$$

We search for all locations $(\mathbf{x}, t, \tau)$ where $(\bar{\bar{\mathbf{q}}}, \bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau)$ are linearly dependent and the remaining two eigenvalues of $\nabla\bar{\bar{\mathbf{q}}}$ have imaginary parts. This corresponds to the coplanarity operator in [34] and reads component-wise

$$\lambda_1 \begin{pmatrix} \mathbf{w}_1\\ \mathbf{w}_2\\ 0\\ -1 \end{pmatrix} + \lambda_2 \begin{pmatrix} -\det(\mathbf{w}_t,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_t)\\ \det(\mathbf{w}_x,\mathbf{w}_y)\\ 0 \end{pmatrix} + \lambda_3 \begin{pmatrix} -\det(\mathbf{w}_\tau,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_\tau)\\ 0\\ \det(\mathbf{w}_x,\mathbf{w}_y) \end{pmatrix} = \mathbf{0}. \tag{30}$$

By setting $\lambda_1 = \lambda_3 \det(\mathbf{w}_x, \mathbf{w}_y)$ we can eliminate the fourth component, and the reformulation reads

$$\underbrace{\lambda_2 \begin{pmatrix} -\det(\mathbf{w}_t,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_t)\\ \det(\mathbf{w}_x,\mathbf{w}_y) \end{pmatrix}}_{\mathbf{a}} + \lambda_3 \underbrace{\left(\begin{pmatrix} -\det(\mathbf{w}_\tau,\mathbf{w}_y)\\ -\det(\mathbf{w}_x,\mathbf{w}_\tau)\\ 0 \end{pmatrix} + \det(\mathbf{w}_x,\mathbf{w}_y) \begin{pmatrix} \mathbf{w}_1\\ \mathbf{w}_2\\ 0 \end{pmatrix} \right)}_{\mathbf{b}} = \mathbf{0}. \tag{31}$$

This is a 3D Parallel Vectors problem [17]. The reformulation $\mathbf{a}||\mathbf{b}$ is equivalent to the coplanarity of the vector fields $\bar{\bar{\mathbf{q}}}, \bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau$, and hence $\mathbf{a}||\mathbf{b}$ is satisfied exactly at the cores of swirling streak lines. With this reformulation at hand we can use the powerful extraction techniques available for the Parallel Vectors operator.

6.1.2 3D Time-Dependent Flows ($n = 3$)

In this case we get

$$\{\bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau\} = \left\{ \begin{pmatrix} -\det(\mathbf{w}_t,\mathbf{w}_y,\mathbf{w}_z)\\ -\det(\mathbf{w}_x,\mathbf{w}_t,\mathbf{w}_z)\\ -\det(\mathbf{w}_x,\mathbf{w}_y,\mathbf{w}_t)\\ \det(\mathbf{w}_x,\mathbf{w}_y,\mathbf{w}_z)\\ 0 \end{pmatrix}, \begin{pmatrix} -\det(\mathbf{w}_\tau,\mathbf{w}_y,\mathbf{w}_z)\\ -\det(\mathbf{w}_x,\mathbf{w}_\tau,\mathbf{w}_z)\\ -\det(\mathbf{w}_x,\mathbf{w}_y,\mathbf{w}_\tau)\\ 0\\ \det(\mathbf{w}_x,\mathbf{w}_y,\mathbf{w}_z) \end{pmatrix} \right\}. \tag{32}$$

We consider the 3 remaining eigenvalues of $\bar{\bar{\mathbf{q}}}$ which do not belong to $\{\bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau\}$ and consider only regions where two of them are imaginary. Let $\bar{\bar{\mathbf{e}}}_3$ be the only remaining non-imaginary eigenvector. Then we search for all locations $(\mathbf{x}, t, \tau)$ where $(\bar{\bar{\mathbf{q}}}, \bar{\bar{\mathbf{e}}}_t, \bar{\bar{\mathbf{e}}}_\tau, \bar{\bar{\mathbf{e}}}_3)$ are linearly dependent and the remaining two eigenvalues of $\nabla\bar{\bar{\mathbf{q}}}$ have imaginary parts. A reformulation using the PV operator and an actual implementation of this case are left for future work.

6.1.3 Examples for 2D Time-Dependent Flows

Moving Focus We consider the vector field (26) which we already used in Section 4 to evaluate the accuracy of computing $\bar{\bar{\mathbf{q}}}$. For this, we have as stream line cores (i.e., locations where $\mathbf{v} = \mathbf{0}$) $\mathbf{x}(t) = (t, 0)^T$ and as path line cores (i.e., locations where $\nabla\bar{\mathbf{p}} \cdot \bar{\mathbf{p}} || \bar{\mathbf{p}}$) $\mathbf{x}(t) = (-1+t, -1)^T$. The closed solution for the streak line vector field $\bar{\bar{\mathbf{q}}}$ from (27) leads us to the core of swirling streak lines in this flow:

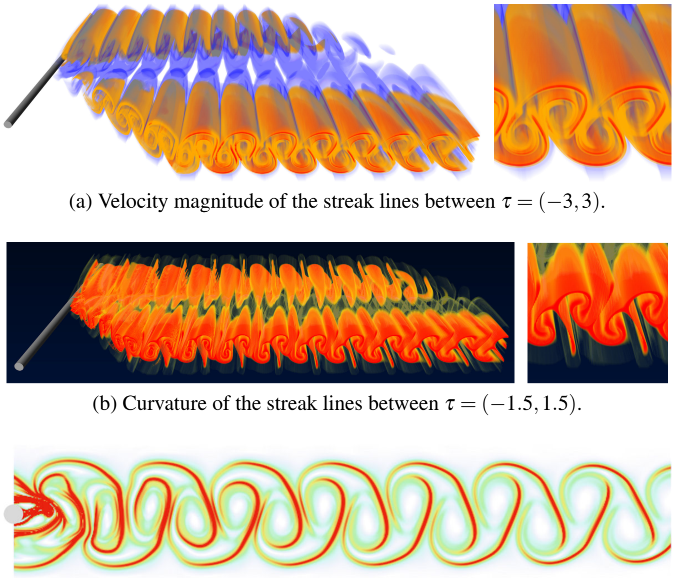
$$\mathbf{x}(t,\tau) = e^{\frac{\tau}{2}} \begin{pmatrix} \frac{2}{3}\sin(\frac{\tau}{2}\sqrt{3})\sqrt{3} + 2\cos(\frac{\tau}{2}\sqrt{3}) - 1 + t\\ -\frac{2}{3}\sin(\frac{\tau}{2}\sqrt{3})\sqrt{3} + 2\cos(\frac{\tau}{2}\sqrt{3}) - 1 \end{pmatrix}. \tag{33}$$

This makes it obvious, how different the streak line core is compared to the cores of stream and path lines. Figure 13 shows the streak line core in the center of swirling streak lines.

Beads Problem Wiebel et al. [37] reported of a biofluid dynamic model where neither classic visualization methods such as LIC or path lines, nor feature extraction methods such as vector field topology or FTLE where able to detect an apparent attractor in the flow, i.e., a point in the flow where particles aggregate. Wiebel et al. used particle density to extract the attractor. Since the simulation is not available to us, we use an analytic variant [16] of this flow which exhibits similar properties:

$$\mathbf{v}(x,y,t) = \begin{pmatrix} -(y - \frac{1}{3}\sin(t)) - (x - \frac{1}{3}\cos(t))\\ (x - \frac{1}{3}\cos(t)) - (y - \frac{1}{3}\sin(t)) \end{pmatrix}. \tag{34}$$

Since this is a time-periodic flow, the attractor can be found using the method of Shi et al. [25]. It is a path line with the following parametric description: $\mathbf{x}(t) = \frac{1}{3}(\sin(t) + \cos(t), -\cos(t) + \sin(t))^T$. We computed $\bar{\bar{\mathbf{q}}}$ using our numerical method in the domain $D \times T \times \Upsilon = [-2,2]^2 \times [0, 2\pi] \times [-9, 0]$ with a resolution of $50^2 \times 100 \times 100$. From this we computed the cores of swirling streak lines and found the following: with decreasing τ, the streak line cores converge to the attrac-



(a) Velocity magnitude of the streak lines between $\tau = (-3, 3)$.

(b) Curvature of the streak lines between $\tau = (-1.5, 1.5)$.

(c) Forward integrated FTLE field ($T^+ = 3$) of the same time step.

Figure 15. Derived scalar fields of the streak lines of the 2D time-dependent flow around a cylinder (gray tube). Shown is the domain $D \times \Upsilon$ for a fixed time step. Two parts of the domain are not defined (front-left and back-right), since the path lines left the domain during the computation of the streak line vector field $\bar{\bar{\mathbf{q}}}$. The images to the right are close-ups.

tor.[9] In other words, our new method is able to detect the attractor reliably in contrast to various other feature extraction methods. Figure 14 shows the streak line core together with the attractor, as well as the cores of swirling stream and path lines. The latter two methods are clearly off the attractor, i.e., do not detect it. We omit the FTLE field since it is, interestingly, constant for this flow and does not reveal any features at all.

We believe that this is a very promising result, since the Beads problem is considered to be one of the major test cases for a successful approach to an unsteady flow topology. However, it has to be left to future investigations whether or not this extends to other challenging flows and streak line cores can serve as a basis for an unsteady flow topology.

6.2 Derived Scalar Quantities

For every point in the $(n+2)$-dimensional streak line vector field there is one and only one streak line through it. This allows to define derived scalar fields describing streak line properties just by considering the derivatives of $\bar{\bar{\mathbf{q}}}$. Integrating the streak lines themselves is not required.

A simple derived scalar field is the velocity magnitude of streak lines, which is simply given as $|\mathbf{w}|$, i.e., as the magnitude of the first two components of $\bar{\bar{\mathbf{q}}}$. Note that we have to omit the other two components of $\bar{\bar{\mathbf{q}}}$, since they are not invariant under domain scaling. Figure 15a shows a volume rendering of $|\mathbf{w}|$ for the 2D time-dependent cylinder flow. The revealed structures bear a high similarity to the streak lines themselves (compare to Figure 10a).

The curvature of stream lines [28, 35] has proven to be a useful tool for vector field visualization: it approaches very high values in the proximity of critical points and together with stochastic seeding algorithms this emphasizes turbulent regions [33]. The curvature of path lines – among other scalar fields – has been used by Shi et al. [24] to select application-relevant path lines within an interactive brushing and focus+context visualization.

To define the curvature of streak lines, we need their first and second derivatives. Since they are tangent curves of $\bar{\bar{\mathbf{q}}}$, their first derivative is $\bar{\bar{\mathbf{q}}}$ itself by definition. Their second derivative is the acceleration vector of the streak line vector field. Again, we omit the last two components of $\bar{\bar{\mathbf{q}}}$ to be invariant under domain scaling, and yield

$$\kappa_2(\mathbf{w}) = \frac{\det(\mathbf{w}, \nabla\mathbf{w}\cdot\mathbf{w})}{\|\mathbf{w}\|^3} \quad , \quad \kappa_3(\mathbf{w}) = \frac{\|\mathbf{w} \times \nabla\mathbf{w}\cdot\mathbf{w}\|}{\|\mathbf{w}\|^3} \qquad (35)$$

for $n = 2$ and $n = 3$ respectively. Figure 15b shows a volume rendering of the streak line curvature in the cylinder data set. While the curvature structures differ from the magnitude, the rolling up of the vortices in the von Kármán vortex street can clearly be observed.

The FTLE field of this flow is shown in Figure 15c. It reveals the vortices in a similar fashion as the velocity magnitude of the streak lines. FTLE is an important tool for the feature-based analysis of flows and has already been considered as a candidate for an unsteady flow topology [21]. Note, however, that the FTLE field for the *Beads Problem* is constant and does not reveal any features.

7 Conclusions and Future Work

We presented the – to the best of our knowledge – first description of streak lines as tangent curves of a derived vector field. This is based on a formulation of the streak line vector field using spatial and temporal derivatives of the flow map. The evaluation of our numerical computation of the streak line vector field has proven that an integration in this new vector field yields streak lines with high accuracy. Furthermore, we have given a parametrization of streak lines that allows to address every streak line by its unique seeding point in space-time.

This novel mathematical description opens the gates to a number of visualization and analysis tools that have been developed in our community, but were previously only available for stream and path lines. Not surprisingly, streak lines and surfaces can be computed almost instantly with our new streak line vector field. As shown in Section 5, this gives rise to a new interaction metaphor for integration-based streak line visualization methods. Furthermore, we extended known feature extraction and analysis tools to work with streak lines. We have given the first mathematical characterization of cores around which streak lines exhibit swirling motion and have shown that these new cores differ significantly from the cores of stream and path lines. This allowed us to find the attractor in the *Beads Problem* flow. Finally, we used the streak line vector field to derive scalar fields describing important properties of streak lines such as their velocity magnitude and curvature.

We already mentioned a number of possible future research directions throughout the paper. Most importantly, the role of streak line cores for an unsteady flow topology has to be investigated in detail. The practical issues around their extraction for 3D time-dependent flows deserve attention as well.

A number of algorithms are available [31, 12, 15, 19] to create evenly-spaced stream line visualizations. Is something similar also possible for streak lines? A first approach to this has been presented in [22]. The new streak line vector field should make it at least simpler to generate visualizations where a set of streak lines is selected for minimal overlap but maximal coverage.

Finally, one out of four characteristic curves is left: time lines. Can they be represented as tangent curves? One has to consider the following fact: streak lines coincide with stream and path lines for steady vector fields, but time lines do not. In particular, the seeding curve for a time line can be of arbitrary shape. This might be the major obstacle that has to be tackled first in order to do for time lines what we did in this paper for streak lines.

Acknowledgments

Tino Weinkauf is supported by a Feodor Lynen research fellowship of the Alexander von Humboldt foundation. Holger Theisel is partially supported by the German Ministry of Education and Science (BMBF) within the ViERforES project (no. 01IM08003C) and by the SemSeg project under the EU FET-Open grant 226042.

[9] Note that streak line cores of a 2D time-dependent flow are 4D surfaces. Hence, we have to intersect them along τ to get a line in space-time.

References

[1] D. Bauer and R. Peikert. Vortex tracking in scale space. In *Data Visualization 2002. Proc. VisSym 02*, pages 233–240, 2002.

[2] S. L. Brunton and C. W. Rowley. Fast computation of finite-time Lyapunov exponent fields for unsteady flows. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(1):017503, 2010.

[3] K. Bürger, F. Ferstl, H. Theisel, and R. Westermann. Interactive streak surface visualization on the GPU. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 15(6):1259–1266, 2009.

[4] S. Camarri, M.-V. Salvetti, M. Buffoni, and A. Iollo. Simulation of the three-dimensional flow around a square cylinder between parallel walls at moderate Reynolds numbers. In *XVII Congresso di Meccanica Teorica ed Applicata*, 2005.

[5] N. Cuntz, A. Kolb, R. Strzodka, and D. Weiskopf. Particle level set advection for the interactive visualization of unsteady 3D flow. *Computer Graphics Forum (Proc. Eurovis)*, 27(3):719–726, 2008.

[6] N. Cuntz, A. Pritzkau, and A. Kolb. Time-adaptive lines for the interactive visualization of unsteady flow data sets. *Computer Graphics Forum*, 28(8):2165–2175, 2009.

[7] C. Garth, F. Gerhardt, X. Tricoche, and H. Hagen. Efficient computation and visualization of coherent structures in fluid flow applications. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6):1464–1471, 2007.

[8] C. Garth, H. Krishnan, X. Tricoche, T. Tricoche, and K. I. Joy. Generation of accurate integral surfaces in time-dependent vector fields. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 14(6):1404–1411, 2008.

[9] G. Haller. Distinguished material surfaces and coherent structures in three-dimensional fluid flows. *Physica D*, 149(4):248–277, 2001.

[10] J. Hultquist. Constructing stream surfaces in steady 3D vector fields. In *Proc. IEEE Visualization '92*, pages 171–177, 1992.

[11] International CFD Database, http://cfd.cineca.it/.

[12] B. Jobard and W. Lefer. Creating evenly-spaced streamlines of arbitrary density. In *Proceedings 8th Eurographics Workshop on Visualization in Scientific Computing*, pages 57–66, Boulogne, 1997.

[13] J. Kasten, C. Petz, I. Hotz, B. Noack, and H.-C. Hege. Localized finite-time Lyapunov exponent for unsteady flow analysis. In *Proc. Vision, Modeling and Visualization*, pages 265–274, 2009.

[14] H. Krishnan, C. Garth, and K. Joy. Time and streak surfaces for flow visualization in large time-varying data sets. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 15(6):1267–1274, 2009.

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

[16] R. Peikert, 2009. private communication.

[17] R. Peikert and M. Roth. The parallel vectors operator - a vector field visualization primitive. In *Proc. IEEE Visualization 99*, pages 263–270, 1999.

[18] S. Popinet. Free computational fluid dynamics. *ClusterWorld*, 2(6), 2004.

[19] O. Rosanwo, C. Petz, S. Prohaska, I. Hotz, and H.-C. Hege. Dual streamline seeding. In *Proc. IEEE Pacific Visualization*, pages 9–16, 2009.

[20] F. Sadlo and R. Peikert. Efficient visualization of Lagrangian coherent structures by filtered AMR ridge extraction. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6):1456–1463, 2007.

[21] F. Sadlo and D. Weiskopf. Time-dependent 2-D vector field topology: An approach inspired by Lagrangian coherent structures. *Computer Graphics Forum*, 29(1):88–100, 2010.

[22] A. Sanna, B. Montrucchio, and R. Arinaz. Visualizing unsteady flows by adaptive streaklines. In *Proc. WSCG'2000*, Plzen, Czech Republic, 2000.

[23] D. Schneider, A. Wiebel, and G. Scheuermann. Smooth stream surfaces of fourth order precision. *Computer Graphics Forum (Proc. EuroVis)*, 28(3):871–878, 2009.

[24] K. Shi, H. Theisel, H. Hauser, T. Weinkauf, K. Matkovic, H.-C. Hege, and H.-P. Seidel. Path line attributes - an information visualization approach to analyzing the dynamic behavior of 3D time-dependent flow fields. In H.-C. Hege, K. Polthier, and G. Scheuermann, editors, *Topology-Based Methods in Visualization II*, Mathematics and Visualization, pages 75–88. Springer, 2009. Topo-In-Vis 2007, Grimma, Germany, March 4 - 6.

[25] K. Shi, H. Theisel, T. Weinkauf, H. Hauser, H.-C. Hege, and H.-P. Seidel. Path line oriented topology for periodic 2D time-dependent vector fields. In *Proc. Eurographics / IEEE VGTC Symposium on Visualization (EuroVis '06)*, pages 139–146, Lisbon, Portugal, 2006.

[26] D. Stalling. *Fast Texture-based Algorithms for Vector Field Visualization*. PhD thesis, FU Berlin, Department of Mathematics and Computer Science, 1998.

[27] D. Sujudi and R. Haimes. Identification of swirling flow in 3D vector fields. Technical report, Department of Aeronautics and Astronautics, MIT, 1995. AIAA Paper 95-1715.

[28] H. Theisel. *Vector Field Curvature and Applications*. PhD thesis, University of Rostock, 1995.

[29] H. Theisel, J. Sahner, T. Weinkauf, H.-C. Hege, and H.-P. Seidel. Extraction of parallel vector surfaces in 3D time-dependent fields and application to vortex core line tracking. In *Proc. IEEE Visualization 2005*, pages 631–638, 2005.

[30] H. Theisel, T. Weinkauf, H.-C. Hege, and H.-P. Seidel. Topological methods for 2D time-dependent vector fields based on stream lines and path lines. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):383–394, 2005.

[31] G. Turk and D. Banks. Image-guided streamline placement. In *Proc. Siggraph '96*, pages 453–460, 1996.

[32] W. von Funck, T. Weinkauf, H. Theisel, and H.-P. Seidel. Smoke surfaces: An interactive flow visualization technique inspired by real-world flow experiments. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2008)*, 14(6):1396–1403, 2008.

[33] T. Weinkauf, H.-C. Hege, B. Noack, M. Schlegel, and A. Dillmann. Coherent structures in a transitional flow around a backward-facing step. *Physics of Fluids*, 15(9):S3, 2003. Winning Entry from the Gallery of Fluid Motion 2003.

[34] T. Weinkauf, J. Sahner, H. Theisel, and H.-C. Hege. Cores of swirling particle motion in unsteady flows. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization 2007)*, 13(6):1759–1766, 2007.

[35] T. Weinkauf and H. Theisel. Curvature measures of 3D vector fields and their applications. *Journal of WSCG*, 10(2):507–514, 2002.

[36] D. Weiskopf. Dye advection without the blur: A level-set approach for texture-based visualization of unsteady flow. *Computer Graphics Forum (Eurographics 2004)*, 23(3):479–488, 2004.

[37] A. Wiebel, R. Chan, C. Wolf, A. Robitzki, A. Stevens, and G. Scheuermann. Topological flow structures in a mathematical model for rotation-mediated cell aggregation. In *Proc. Topo-In-Vis 2009*, page to appear, Snowbird, Utah, U.S.A., 2009.

[38] A. Wiebel, X. Tricoche, D. Schneider, H. Jaenicke, and G. Scheuermann. Generalized streak lines: Analysis and visualization of boundary induced vortices. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 13(6):1735–1742, 2007.