# STREAMLINE TRACING VISUALIZATION OF FLOW ON SURFACES

Fukun Wu[1,2], Yi Cao[2] and Qingli Dong[1,2]
[1]*CAEP Software Center for High Performance Numerical Simulation, Beijing, China*
[2]*Institute of Applied Physics and Computational Mathematics, Beijing, China*

**ABSTRACT**

To address the visualization problem of surface streamline on complex curved surfaces, we present and implement an efficient surface streamline generation method for visualizing vector field on arbitrary geometry. The intersection calculation on curved surfaces with complex topology is performed to achieve the high-precision underlying vector. Then, this approach adopts an extended Runge-Kutta streamline integration technology for performing streamline tracing on an unstructured mesh, where the adaptive stepsize strategy and intersection acceleration structure are presented for sake of simplicity and efficiency. Finally, this approach is integrated into a visualization software to enhance the visualizing capability. The experimental results demonstrate that our approach can efficiently and accurately generate continuous and consistent geometric surface streamlines on curved surfaces.

**KEYWORDS**

Surface Streamline, Intersection Calculation, Extended Runge-Kutta, Adaptive Stepsize Strategy

## 1. INTRODUCTION

The analysis of flow on polyhedral surfaces is of particular importance for design and optimization of complex geometry in computational fluid dynamics, which plays a key role in visual analysis of electromagnetic shielding effects, surface cur-rents on electronic equipment and flow characteristics inside engine combustion chambers from physical sciences and engineering.

A significant body of research is dedicated to vector field visualization such as LIC (Battke et al. 1997, Cabral and Leedom 1993, Mao et al. 1997), ISA (Laramee et al. 2003), or IBFVS (van Wijk, 2003), in order to support exploration of flow on large and unstructured polygonal meshes. Although texture-like representations allow to in-crease the spatial resolution and to depict small details accurately, it is challenging to visualize the flow feature on CAD models for engineering designer. Covering an image with a set of evenly spaced streamlines is a good way to visualize the flow features. However, image quality enhancement need be achieved by using streamline placement algorithms which optimize the placement of a set of streamlines according to an image-based criterion. In other words, these methods can generate sparse or dense representations of vector fields. However, the generation and advection of texture properties in object space is finally projected to image space.

Directly performing streamline tracing on 3D curved surfaces is interesting due to their visual intuitiveness from the view of designer. The work on streamline tracing visualization has been concentrated on Runge-Kutta integration methods which cannot adapt to the constraints of complex geometric surfaces due to the intrinsic properties. It is difficult to generate the continuous and consistent streamline on curved surfaces with the challenge of the robust intersection testing and numerically stable methods. Polthier and Schmies (Polthier and Schmies 2006) introduced straightest geodesics in geometry to solve the initial value problem for geodesics on polyhedral surface. However, their algorithm only uses intrinsic geometric properties of the polyhedral surfaces, not considering underlying discrete triangulation of surfaces in vector field.

This paper proposes an efficient streamline generation algorithm for the analysis of flow on curve surfaces. Different from previous work, we introduce the high precision polygon intersection and interpolation algorithms to achieve vector field extraction on surfaces with complex topology which keep the underlying detail of geometric surfaces in vector field. Then, we present surface streamline integration technology to perform streamline tracing on unstructured mesh where the fourth-order Runge-Kutta integrator is extended to 2D surfaces.

## 2. RELATED WORK

Visualizing vector fields defined on curved surfaces or manifolds is of particular im-portance for engineering designer in computational fluid dynamics and has received comparatively much attention. Due to the complex topology of CAD geometries with holes and discontinuities, to use a technique based on surface parametrization would be especially difficult. The research on the visualization of vector fields defined on surfaces focuses on texture-based approaches (Laramee 2004, Stalling and Hege 1997, Li et al. 2008). In the early 1990s, Spot Noise (van Wijk 1991) and LIC (Cabral and Leedom 1993, Forssell and Cohen 1995) were presented to generate dense representations based on textures which are limited to curvilinear surface in 2D. An extension of LIC for arbitrary surfaces in 3D were proposed by Mao et al. (Mao et al. 1997) where the convolution of noise image with filter kernels are performed only at visible ray-surface intersections. See Stalling et al. (Stalling and Hege 1997) for more comprehensive overviews of LIC techniques applied to surfaces. In order to overcome computation time hurdle introduced by LIC, two representative approaches, ISA (Laramee et al. 2004) and IBFVS (van Wijk 2003), were proposed to generate dense representation of flow on complex surfaces at fast frame rates. Although texture properties are advected on boundary surfaces in 3D, they essentially realized texture advection in image space in 2D by projecting the surface geometry and its associated vector field to image space and then applying texturing.

Covering an image with a set of streamlines is also a good way to visualize the flow features. A significant body of research is dedicated to using image-based approach to generating streamlines with different streamline seeding strategies for vector fields defined on curved surfaces (Li and Shen 2007, Mattausch et al. 2003, McLoughlin 2010, Spencer et al. 2009, Verma et al. 2000, Ye et al. 2005). These methods can generate sparse or dense representations of vector fields, and image quality enhancement can be achieved by using streamline placement algorithms based on an image criterion. However, the generation and advection of texture properties is still confined to image space.

It is interesting for engineering designer to use geometric streamline approach due to their visual intuitiveness. This can help them analyze the flow feature of arbitrary positions on curved surfaces in order to aid the design of computational fluid dynamics models. However, it is very difficult to perform streamline tracing on an unstructured mesh with the challenge of the robust intersection testing and numerically stable methods of handing special cases (Reshetnyak 1989). Geodesic curves in geometry solve the initial value problem on polyhedral surfaces on smooth surfaces and play a key role to streamline generation (Polthier and Schmies 2006, Spencer et al., Alexander and Bishop 1996). For instance, Polthier and Schmies presented an efficient straightest geodesics method to generate streamlines on polyhedral surface and arbitrary manifolds, which allows to move uniquely on a polyhedral surface in a given direction along a straightest geodesic. Their work only uses geometric properties of the polyhedral surface and not considers the underlying discrete triangulation of the surface. Inspired by their work, we achieve the highly detailed underlying vector by surface extraction and present a novel and conceptually simple method of generating streamlines on curve surfaces.

## 3. ALGORITHM

The streamline generation algorithm proposed in this paper is applied to vector fields defined on complex curved surfaces. The key to performing streamline tracing is to accurately calculate the integral value of the streamline on curved surfaces.

### 3.1 Surface Extraction

From the view of engineering designer in computation fluid dynamics, the surface streamline generation algorithm is essentially different from the traditional image-based streamline visualization algorithms

(Laramee et al. 2004, Spencer et al. 2009). It is limited by the geometry model with high-order, discontinuous and multi-scale characteristics, and requires the high-precision acquisition of the vector field on the geometric surface. Here, our algorithm extends the previous polygon clipping approaches presented by Greiner (Greiner and Hormann 1998) and Vatti (Vatti 1992) to 3D curved surfaces to achieve the high-precision surface extraction which keep the underlying detailed structures of geometric surface on vector field.

Considering the fact that vector field may be composed of structured or unstructured grids, this paper applies the grid-related interpolation algorithms to calculate the vector field on extracted surfaces. For unstructured grids, we adopt the trilinear interpolation method to calculate the vector on arbitrary point on curved surfaces. For structured grids, domain-related interpolation methods are applied.

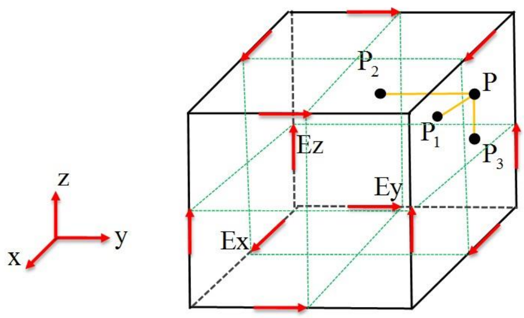



Figure 1. Geometric structure of vector field E defined on the hexahedral cell

Figure 1 shows an example of the structured grid generated by the numerical solver. The physical quantity $E$ in vector field is registered as the center of the grid cell, but the components of the vector $E$ are located at the edge center of the grid cell. In order to calculate the vector value of any node $P$ on the complex geometric surface, three reference planes, as shown in the area enclosed by the three groups of green dashed lines in Figure 2, are introduced for vector field defined on structured grids. The projection points of the point $P$ on the three reference planes are $P_1$, $P_2$, and $P_3$ respectively. $P_1$ is used to calculate the x-axis component $E$x of the vector field E associated with $P$. $P_2$ and $P_3$ are used to calculate y-axis component $E$y and z-axis component $E$z respectively.

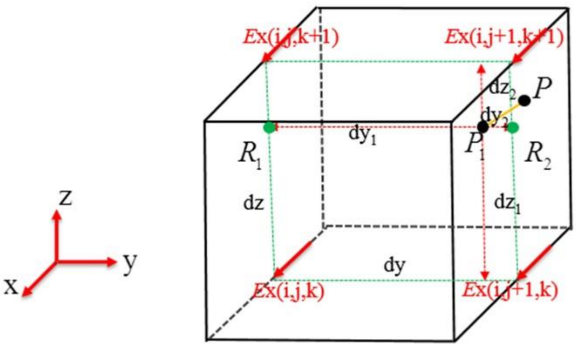



Figure 2. Geometric structure of vector field E defined on the hexahedral cell

Figure 2 shows the example of calculating the projection point $P_1$, and the complete linear interpolation formula is:

$$Ex(P_1) = \frac{dy_1}{dy} * Ex_{R_1} + \frac{dy_2}{dy} * Ex_{R_2} \tag{1}$$

where,

$$Ex_{R_1} = \frac{dz_1}{dz} * Ex(i, j, k) + \frac{dz_2}{dz} * Ex(i, j, k+1)$$
$$Ex_{R_2} = \frac{dz_1}{dz} * Ex(i, j+1, k) + \frac{dz_2}{dz} * Ex(i, j+1, k+1) \quad (2)$$

Substituting the $Ex_{R_1}$ and $Ex_{R_2}$ in Equation 2 into Equation 1 can obtain the component of $Ex(P_1)$. $Ey(P_2)$ and $Ez(P_3)$ can be obtained in the same way.

## 3.2 Surface Streamline Generation

Through extracting vector field using curved surfaces, we can obtain a high precision surface vector field which provides the pre-processing data for streamline generation. Then, according to the initial seed point specified interactively by engineering designer, a streamline moving along the geometric surface needs to be generated. However, due to the limitation from the complex curved surfaces, the classical Runge-Kutta integrators is no longer valid. Focusing on this central challenge, this paper adopts a new and extended Runge-Kutta method based on surface projection for integration calculation, with the aid of an adaptive stepsize strategy and streamline-surface intersection acceleration structure, to accomplish the continuous and consistent surface streamline generation.

### 3.2.1 Runge-Kutta Integral

The core idea of the surface streamline generation algorithm is to track and calculate the motion trajectory of these seed points on curved surfaces in vector field. The trajectory is a curve, namely, a streamline. For each streamline $x(t)$, its motion equation satisfies:

$$\frac{dx_t}{dt} = V(t, x) \quad (3)$$

Where $V(t, x)$ is the value of vector field at the position $x(t)$, and for the static field, $V(t, x) = V(x)$.

Considering the case that surface streamlines are constrained by geometric shapes, this paper defines an extended Runge-Kutta streamline integral formula as

$$x(t+h) = x(t) + (Proj_{\Omega}(\int_{t}^{t+h} V(t, x)dt)) \quad (4)$$

Where Ω represents the geometric surface and ProjΩ represents the projection of the integral formula on the geometric surface. The choice of *h* is more important. If h is too small, the calculation overhead will be too large. If *h* is too large, it will bring greater errors and reduce the accuracy of streamline generation.

For generating continuous surface streamline in vector field, we need to iteratively calculate the points of the streamline according to the Equation 4, and finally obtain a complete streamline that can show the move status of flow field. We adopt the fourth-order Runge-Kutta method used for concrete calculation. This method is used as the basis of solving the derivative and initial values in differential equations. Specifically, the next step value $x_{i+1}$ of the streamline is determined by the current value $x_i$ plus the projection on the surface of the product of the step interval *h* and an estimated slope, formulated by

$$x_{i+1} = x_i + Proj_{\Omega}(h(K_1 + 2K_2 + 2K_3 + K_4)/6) \quad (5)$$

Where

$$
\begin{aligned}
&K(1)=V(t_i, x_i)\\
&K(2)=V(t_i+\frac{h}{2}, x_i+\frac{hK_1}{2})\\
&K(3)=V(t_i+\frac{h}{2}, x_i+\frac{hK_2}{2})\\
&K(4)=V(t_i+h, x_i+hK_3)
\end{aligned} \tag{6}
$$

### 3.2.2 Stepsize Adaptive Adjustment

When iteratively performing streamline integration, the stepsize is comprehensively weighed according to the calculation accuracy and efficiency, and it has a decisive influence on the local error. When using a fixed step size, the complexity of the projection calculation becomes a time-consuming problem. As shown in Figure 3, $s_1$, $s_2$, and $s_3$ represent three polygonal cells on the geometric surface, and $p_1$, $p_2$, and $p_3$ represent the projection points of streamline integration on surface. The structural diversity of curved surfaces increase the complexity of generating surface streamlines after projection.

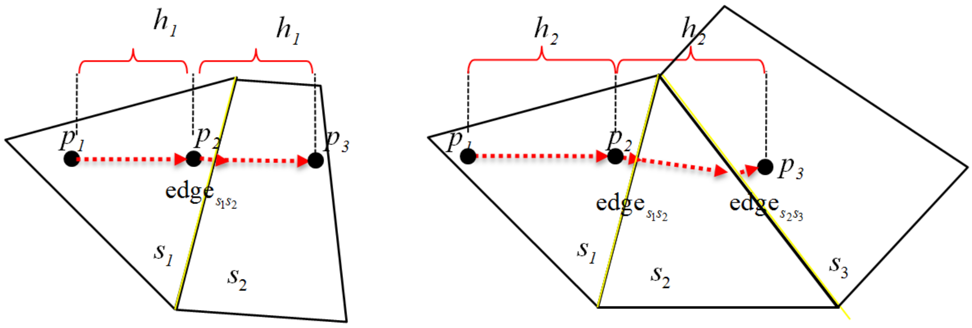



Figure 3. A Surface projection under two types of integration with fixed steps h1 (left) and h2(right)

For solving this problem, we construct an adaptive stepsize method based on the grid size and shape to calculate and generate surface streamline as shown in Figure 4.

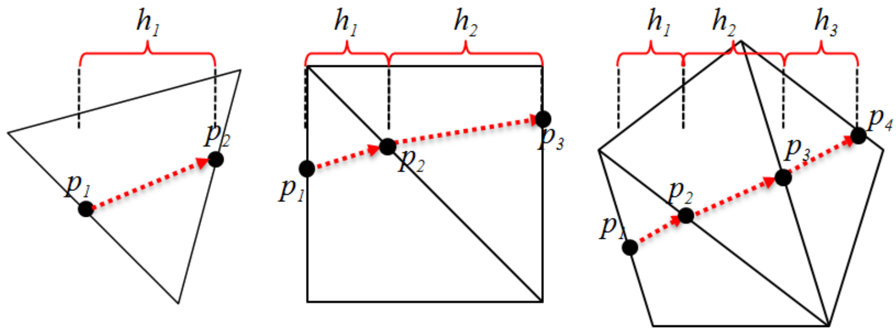



Figure 4. A Surface projection under two types of integration with fixed steps $h_1$ (left) and $h_2$(right)

According to the type of polygon cells on curved surface, different stepsizes are processed during surface integration. For a triangle, the integral projection directly crosses the internal field of the triangle, and the step length is only $h_1$. The quadrilateral or pentagon is triangulated separately to form two step sets $h_1h_2$ and $h_1h_2h_3$, where there is still only one integration step inside each triangle. This algorithm avoids the complicated calculations when the streamlines are projected on the geometric surface, and at the same time takes into account the grid shape and the changes of cell volume. When visualizing streamlines, we synthesize surface streamlines orderly according to the adaptively generated projection pointsets.

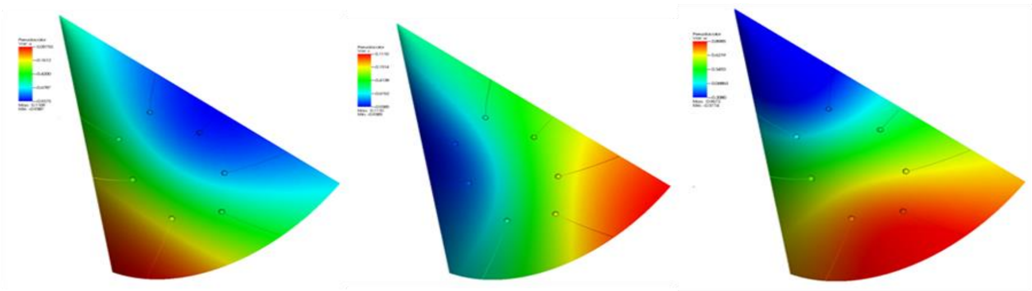
# 4. EXPERIMENTAL RESULTS AND ANALYSIS

We implemented our surface streamline generation algorithm and integrate it into a general visual analysis platform, namely VisIt (VisIt 2005), in the form of plugin-in component. We also demonstrated its validity by applying our method to representative datasets for visualizing the surface field of objects with regular and irregular geometry. All of the images in this work were produced on a Dell T7600 workstation with a 2.40 GHz Intel Xeon CPU E5-2609.

For geometric models with arbitrary topology, the high-precision extraction algorithm proposed in this paper can obtain the surface vector field with the same mesh density and accuracy as original 3D vector field, meanwhile maintaining the profile of the geometric model. Table 1 provides a concise overview of performance by extracting surface vector field using cylinder and sphere models, where the consumed time can be further reduced with the increase of the cpu cores.

Table 1. Performance comparison of extracting two types of dataset using cylinder and sphere under 6 CPU cores

| Type of dataset | Num of cells | Time(cylinder) | Time(sphere) |
|---|---|---|---|
| Structured grid | 36000 | 0.257 s | 0.189 s |
| Unstructured grid | 45618700 | 3.872 s | 3.591 s |



a. variable = u    b. variable = v    c. variable = w

Figure 5. Visualization of coloring surface streamline using different physical variable. The comparison with the underlying scalar field shows the expected better approximation quality

A very important feature in streamline rendering is to support the visualization of streamline colors based on arbitrary physical field values. Figure 5 shows the surface streamline coloring effects based on the four types of scalar fields u, v, and w, with the corresponding scalar field coupled in the background. It can be concluded that our surface streamline generation algorithm can effectively describe the flow characteristics of the vector field and the changing trend of physical quantities.

The complexity of the geometric surface models varies with numerical devices simulated in the field of engineering. As shown in Figure 6, this paper selects another common geometric surface model for surface streamline generation and verification. Figure 6a shows the high-precision vector field obtained after surface extraction. The upper half of the vector diverges upward, and the lower half of the vector diverges downward, that is, there should be two completely different directions of surface flow on the entire surface. Figure 6b shows the continuous and consistent surface streamlines generated using 14 seed points, which is the same as the trend of the surface flow field, where the color is generated based on the physical field u. Figure 6c is a schematic diagram of the coupling of the pseudocolor diagram and surface streamlines. Figure 6c verifies the flexibility and accuracy of the algorithm in this paper when coloring the surface streamline. In particular, the above experiments further prove that the proposed surface streamline method is not limited by the shape of the geometric model and the characteristics of the vector field.

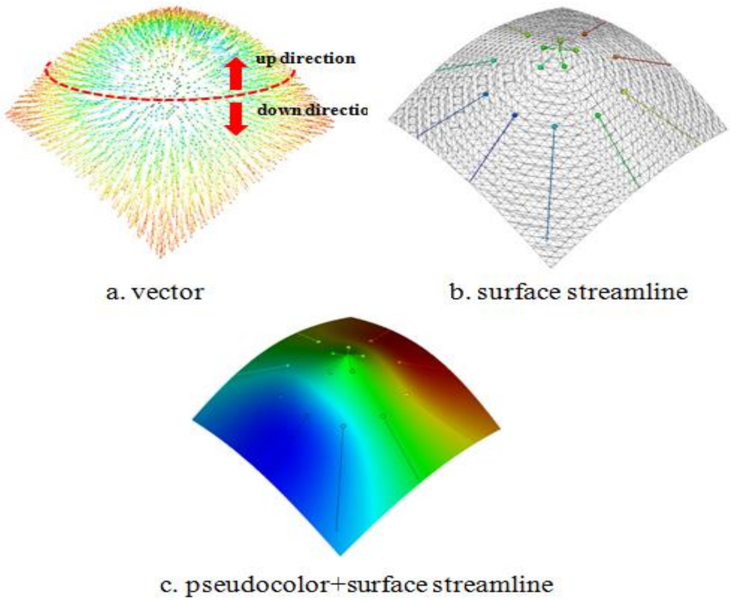



Figure 6. Proximity comparison: (top row) between the underlying direction of motion in vector field and the generated streamlines using our algorithm, (bottom row) between scalar field and our coloring method. The underlying structure of the flow mesh is now more clearly reflected by the streamlines

In addition, a comparison with Tecplot's streamline generation results as illustrated in Figure 7 also demonstrates the expected high approximation quality of our proposed algorithm.

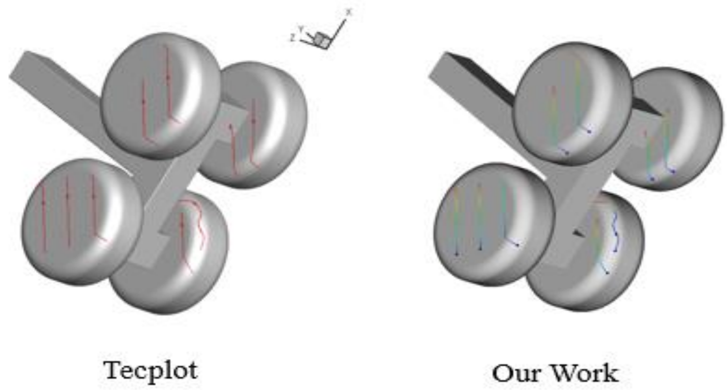



Figure 7. Comparison of generating surface streamlines on curved surface using Tecplot and ours

## 5. CONCLUSION

In this paper, we have proposed a streamline generation algorithm for vector field defined on curved surfaces. This algorithm achieves the underlying discrete feature of curved surfaces by applying the polygon intersection and the grid-related high-precision interpolation algorithms. Then, we present the surface streamline integral technology, combined with the adaptive stepsize strategy and acceleration structure, in order to support continuous and consistent surface streamline generation, which solves the bottleneck problem of performing streamline tracing directly on curved surfaces. We have shown that this work can aid engineering designer for the design and optimization of major devices by analyzing flow on curved surfaces.

## REFERENCES


Alexander, S. and Bishop, R. L., 1996. Comparison theorems for curves of bounded geodesic curvature in metric spaces of curvature bounded above. *Differential Geometry and its Applications*, Vol. 6, No. 1, pp. 67–86.

Battke, H. et al, 1997. Fast line integral convolution for arbitrary surfaces in 3D. *Visualization and Mathematics*, pp. 181–195. Springer, Berlin, Heidelberg.

Cabral, B., Leedom, L. C., 1993. Imaging vector fields using line integral convolution. *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, California, USA, pp. 263–270.

Forssell, L. K., Cohen, S. D., 1995. Using line integral convolution for flow visualization: curvilinear grids, variable-speed animation, and unsteady flows. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 1, No.2, pp. 133–141.

Greiner, G., Hormann, K., 1998. Efficient clipping of arbitrary polygons. *ACM Transactions on Graphics*, Vol. 17, No. 2, pp. 71–83.

Laramee, R. S., 2004. *Interactive 3D Flow Visualization Using Textures and Geometric Primitives*. Vienna University of Technology, Institute for Computer Graphics and Algorithms, Vienna, Austria.

Laramee, R. S. et al, 2004. The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum*, Vol. 23, No. 2, pp. 203–221.

Laramee, R. S. et al, 2003. Image space based visualization of unsteady flow on surfaces. *Proceedings of Visualization*, Los Alamitos, USA, pp. 131–138.

Li, L., Shen, H.-W., 2007. Image-based streamline generation and rendering. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 3, pp. 630–640.

Li, G. et al, 2008. Flow charts: visualization of vector fields on arbitrary surfaces. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 14, No.5, pp. 1067–1080.

Mao, X. et al, 1997. Line integral convolution for 3D surfaces. *Proceedings of. Eurographics*, Vienna, Austria, pp. 57–69.

Mattausch, O. et al, 2003. Strategies for interactive exploration of 3D flow using evenly spaced illuminated streamlines. *Proceedings of Spring Conference on Computer Graphics*, pp. 213–222.

McLoughlin, T. et al, 2010. Over two decades of integration-based, geometric flow visualization. *Computer Graphics Forum*, Vol. 29, No. 6, pp. 1807–1829.

Polthier, K., Schmies, M., 2006. *Straightest geodesics on polyhedral surfaces*. Association for Computing Machinery, New York, USA, pp. 30-38.

Reshetnyak, Y. G., 1989. Two-dimensional manifolds of bounded curvature. *Encyclopaedia of Mathematical Sciences*, Vol.70, pp. 3–164.

Spencer, B. et al, 2009. Evenly spaced streamlines for surfaces: An image-based approach. *Computer Graphics Forum*, Vol.28, No.6, pp. 1618–1631.

Stalling, D., Hege, H.-C., 1997. LIC on surfaces. *Proceedings of Computer Graphics and Interactive Techniques.* Los Angeles, USA, pp. 51–64.

van Wijk, J. J., 2003. Image based flow visualization for curved surfaces. *Proceedings of Visualization*. Los Alamitos, USA, pp. 123–130.

van Wijk, J. J., 1991. Spot noise-texture synthesis for data visualization. *ACM SIGGRAPH Computer Graphics*, Vol. 25, No.4, pp. 309–318.

van Wijk, J.J., 2003. Image based flow visualization for curved surfaces. *Proceedings of Visualization*. Los Alamitos, USA, pp. 123–130.

Vatti, B.R., 1992. A generic solution to polygon clipping. *Communications of the ACM*, Vol.35, No.7, pp. 56–63.

Verma, V. et al, 2000. A flow-guided streamline seeding strategy. *Proceedings of Visualization*. Los Alamitos, USA, pp. 163–170.

VisIt, 2005. *VisIt User's Manual*. Lawrence Livermore National Laboratory, USA.

Ye, X. et al, 2005. Strategy for seeding 3D streamlines. *Proceedings of Visualization*. Los Alamitos, USA, pp. 471–478.