A Streamrunner and Streamcomets for Highly Interactive Visualization of CFD Simulation Data on Versatile, Unstructured Grids

Robert S. Laramee and Helwig Hauser
{Laramee, Hauser}@VRVis.at
VRVis Research Center, Austria
www.VRVis.at

Abstract

Visualization of CFD simulation data on unstructured, 3D grids poses several challenges. The wide range of real-world data set sizes and the geometric versatility within individual, unstructured CFD simulation models present special challenges to the engineers analyzing flow simulation results. Users also face perceptual problems such as occlusion, visual complexity, lack of directional cues, and lack of depth cues. These perceptual problems combined with the versatility of real-world CFD simulation data models motivates highly interactive and versatile visualization solutions.

We present two new integral objects, the streamrunner and streamcomets, that address these challenges. The streamrunner gives the user interactive control over the evolution of streamlines from the seed point until reaching their full length. The streamrunner control minimizes occlusion and visual complexity and maximizes directional and depth cues for flow visualization in 3D. The streamcomet shifts even more interactive control to the user by letting the user control the position of the comet's head along the integral path, the comet tail length, the diameter of the comet head and tail, and optionally the comet's animation speed. We also propose the use of the streamrunner and streamcomets for unsteady flow visualization. Combined with our other interactive 3D flow visualization tools, these new integral objects represent two further steps in supplying a strong demand for more interactive visualization solutions for CFD simulation data on complex, unstructured grids.

Keywords
flow visualization, vector field visualization, streamlines, interaction, occlusion, visual complexity, depth cues, directional cues, CFD simulation data, unstructured 3D grids

1 INTRODUCTION

Demand for visualization solutions for CFD simulation data has grown rapidly in the last decade. This is due, in part, by the interest of manufactures in minimizing the time taken for their production cycle. This objective is realized with the use of software simulation tools to analyze design decisions rather than constructing real, heavy-weight objects.

Here at the VRVis research center we collaborate with AVL in order to provide flow visualization solutions for analysis of their CFD simulation result data. AVL (www.avl.com) is an internationally recognized leader in providing automotive design and CFD simulation solutions to its partners in the automotive industry. AVL works with other internationally recognized companies such as Toyota, DaimlerChrysler – Mercedes-Benz, and Pierburg Instruments GmbH. AVL's own engineers as well as engineers at industry affiliates use flow visualization software to analyze and evaluate the results of their automotive design and simulation on a daily basis. The analysis of an engineer includes tasks such as searching for areas of extreme pressure, looking for symmetries in the flow, searching for critical points, and comparing simulation results with measured, experimental results.

The variety of visualization software users spans the disciplines of mechanical engineering, software engineering, computer science, management, and marketing to name just a few. Working directly within the AVL environment, we at VRVis have the privilege of receiving feedback directly from the engineers, managers, and business marketing executives about the current features in their flow visualization software products. The pervading message we hear consistently is: users are interested in more interactive control of the flow visualization results. This is a classic theme in the realm of scientific visualization [8].

1.1 A WIDE RANGE OF SIMULATION DATA SETS

The users have very good reasons for requesting more interaction control over the visualization results. One reason is the large variety of simulation data sets that undergo analysis. AVL has a large, varied collection of data sets ranging from small geometries such as small fluid conduits and cylinders used for connection objects to mid-range size geometries such as chain drives, water pumps, cooling jackets, air boxes, air vents, intake manifolds, intake ports, catalytic converters, to large geometries such as automotive cabin interiors, automotive exteriors, airplane exteriors, an air flow tunnel, and even a hospital room with an air flow simulation. The full list of simulation data sets is much longer. The geometric sizes of these grids differ by *six* or more orders of magnitude as well as the sizes of the underlying polygons. Hence, the tools used to visualize the simulation results also need to span this range of sizes correspondingly. Furthermore, we speculate that this difference will only increase in the future.

1.2 THE VERSATILITY OF UNSTRUCTURED CFD SIMULATION MODELS

Another reason the users request more interaction control over the visualization results is due to the *versatility* of individual data sets. By versatile, we mean embracing a wide variety of components, features, and levels of resolution. To illustrate this idea, we can look at figure 1 showing two intake ports – small valves in a car engine that allow air into the engine's cylinders. By looking at an overview of the intake port grid, we observe what appears to be four adaptive levels of resolution: (1) for the flow source on the left and the cylinder on the lower, right, (2) another level of resolution for the connecting pipes in the middle, (3+4) and two levels of resolution for the intake port components themselves. However, when we zoom in to have a closer look at the intake ports (figure 2) we are somewhat surprised to find five adaptive levels of resolution used to evaluate the intake ports themselves: (a) two levels for the top of the ports, (b) approximately the same two levels of detail plus an added

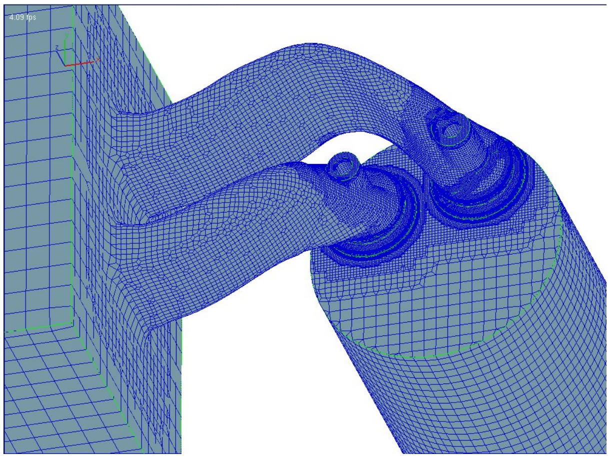

Figure 1: The CFD simulation grid of an intake port. This image illustrates the versatility of a typical, unstructured, CFD simulation grid containing a flow source from the left, two connecting pipes in the middle, two intake ports at the ends of the pipes, and a combustion chamber cylinder on the right *(intake port data courtesy of AVL).*

layer of finer resolution grid cells for a few of the rings around the base of the ports. When we examine the geometric sizes of the individual facets in the intake port grid, we discover that facets in the flow source (on the left in figure 1) are approximately 1000–2000 times larger than the finest resolution facets at the base of the intake ports.

We don't often see such complex, unstructured grids in the flow visualization research literature, however, these grids are a daily experience in the industrial CFD simulation and visualization community. Our goal is to provide flow visualization solutions that are equally as versatile and adaptive as the grids themselves. We approach this challenge by providing the user with highly interactive tools including the streamrunner and streamcomets.

1.3 PERCEPTUAL CHALLENGES IN 3D FLOW VIZ

The majority of flow visualization research literature addresses 2D visualization techniques. This is partly because flow visualization on boundary surfaces and in 3D presents additional perceptual challenges such as occlusion, lack of directional cues, lack of depth cues, and visual complexity. Figure 3 illustrates these problems in the wire-frame context of the two connecting pipes from the intake port shown in figure 1.

All of the CFD simulation models at AVL are unstructured and three dimensional. Although engineers often use 2D cuts through the 3D meshes during their analysis, there is a strong interest in 3D and boundary surface visualization techniques that address the perceptual problems mentioned above. We also know that there is strong evidence to support the notion that users acquire a better understanding of 3D data sets using 3D visualization techniques as opposed to 2D visualization techniques [27].

The rest of this paper is organized as follows: In section 2 we discuss related research in both 2D and 3D flow visualization with a strong interaction component as well as related work in visualization of CFD simulation data. In section 3 we describe the streamrunner feature in the context of two practical examples. Section 4 presents streamcomets in detail with examples of their use. Then we propose the extension of the streamrunner and streamcomets to

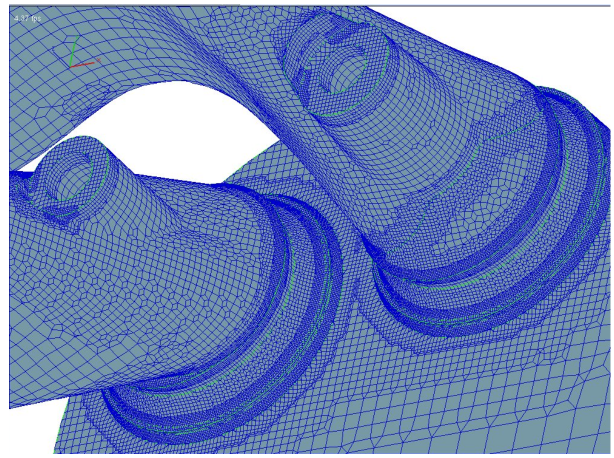

Figure 2: A close-up view of the intake ports in the same CFD simulation grid as shown in figure 1. We can see multiple, adaptive resolution levels of unstructured grid cells.

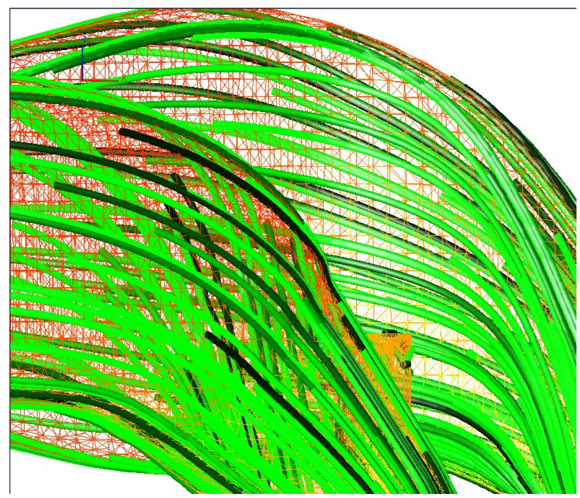

Figure 3: This image illustrates some of the perceptual challenges facing 3D flow visualization including occlusion, visual complexity, and lack of directional cues.

unsteady flow visualization. Finally, we conclude with some initial results and ideas for future work.

2 RELATED WORK

In our review of the flow visualization research literature [7, 16] we do not see very many integration-based visualization techniques whose main emphasis is on user interaction and even *fewer* that address the versatility aspect of CFD simulation grids. However, here we review flow visualization literature with a strong interaction component.

2.1 INTERACTIVE FLOW VIZ

Of the different tools used to visualize flow, streamlines are a very popular choice due to their intuitive semantics and ease of implementation. Jobard and Lefer present an excellent 2D technique that preserves the density of their evenly spaced streamlines [9] while the user is able to zoom in and out of a vector field [10]. Their algorithm also supports enrichment, i.e., providing more details in specific areas of interest via the placement of more streamlines.

De Leeuw and van Wijk introduce a probe for local flow field visualization and analysis [2]. The user can interactively position the probe at a certain location in the 3D flow field and visualize local velocity, curvature, rotation, shear, convergence, divergence, and acceleration. We don't propose the use of such a probe for an initial exploration of a large, complex, unstructured CFD simulation data set. This tool would be extremely useful after a global understanding of the flow field is obtained.

Zöckler et al. [28] present interactive 3D flow visualization with real-time illuminated streamlines. The user-interaction component of their research consists of the use of "draggers" provided by the Open Inventor graphics toolkit. Each dragger defines a rectangular or spherical volume in which streamlines are seeded. However, this method still suffers from occlusion and visual complexity.

Fuhrmann and Gröller [6] use dashtubes with volume filling properties, reduced occlusion, animation of flow for clear direction, and fast rendering. There are a few ways in which their techniques are related to ours since they address the same perceptual problems associated with 3D visualization that we do. They also add strong user interaction techniques via the use of magic lenses and magic boxes, similar to the tools used by Zöckler et al. [28]. However, their presentation severely lacks application to practical data sets. In fact we see no clear illustration of their technique being applied to a real data set. We also supply more interactive degrees of freedom to the visualization via new integration-based glyph representations.

Rezk-Salama et al. [17] present an interactive technique for 3D flow visualization using LIC in combination with 3D texture mapping. Primary user interaction is realized using a clipping plane that is translated interactively. However, their method results in an over complex 3D flow visualization as well as occlusion. Furthermore, the use of a clipping plane makes this interaction technique inherently 2D since the resulting flow visualization using LIC is on a 2D rectangle.

2.2 RELATED RESEARCH IN CFD SIMULATION DATA VISUALIZATION

Here we briefly outline other important research literature that targets the visualization of CFD simulation data but does not place specific emphasis on interaction techniques.

Ebert et al. present three alternative volume rendering approaches for visualizing CFD simulation data [5]. Roth and Peikert discuss techniques for locating vortices within CFD vector field data [18, 19]. Their work is targeted specifically at turbomachinery design with unstructured grids.

Shen et al. describe techniques to accurately compare CFD simulation data with experimental data in the context of a wind tunnel [23]. Rather than comparing data on an image level, their technique compares data at the data set level. Schulz et al. describe flow visualization techniques of fluid dynamics simulations of *Cartesian* grids of multiple resolutions [22]. User interaction is provided by freely movable probes such as rakes, 2D slices, and a cube. Streaklines, stream ribbons, and glyphs are some of the integral objects used in the visualization.

Silver and Wang [24] illustrate several nice examples of their feature tracking algorithm on structured and unstructured CFD data sets over several time steps. Sadarjoen and Post present a vortex detection algorithm that utilizes the geometric properties of streamlines [20]. They illustrate their technique on CFD simulation data sets.

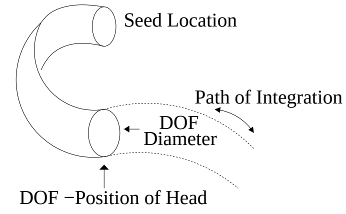

Figure 4: The streamrunner promotes 2 interactive degrees of freedom: (1) the position of the object head along the path of integration and (2) the diameter of the streamtube.

Doleisch and Hauser present a focus+context visualization technique for CFD simulation data in 3D [3]. They minimize visual complexity via the use of a non-discrete degree of interest function. User interaction is provided via different views of the CFD simulation data including a scatter plot view and a histogram view.

3 THE STREAMRUNNER

The streamrunner feature, already acknowledged by the HCI community [14], addresses the problems of occlusion and scene complexity directly by giving the user interactive control over the evolution of streamlines from the time they are seeds until they terminate. A streamline may terminate when it reaches a boundary in the geometry, reaches a region of zero velocity, or reaches a maximum length set by the user. The two interactive degrees of freedom (DOF) afforded by the streamrunner are illustrated in figure 4: (1) the position of the stream's head along the integral path and (2) the diameter of the the integral object, in this case the tube diameter.

3.1 THE STREAMRUNNER FOR FLOW VIZ IN 3D AND ON SURFACES

Using the streamrunner, the user is able to set the stream evolution to integration step 1 as shown in figure 5 (compare with figure 3). At this point, only the streamline seeds are shown. Individual streamlines are easily distinguished and focused upon early in their evolution because occlusion has been almost eliminated while visual complexity is at a minimum. The streamrunner can then be used to change the current integration step of the streamtubes such that the user can watch the streamlines grow, or *run*, in the direction of the flow. This gives a clear, unequivocal indication of flow direction. With the streamrunner, the user is able to focus on an individual streamline, a group of streamlines, or a particular area of the flow field as they interactively adjust the current integration step. Watching the streams flow in 3D combined with shading, in this case using tubes, also gives added depth cues. The streamrunner also allows the user to trace the evolution of the streamtubes *backwards* in order to see where a tube has come from.

3.2 THE STREAMRUNNER FOR DENSE, INTEGRATION-BASED FLOW VIZ ON SLICES

We, as well as other researchers [10], make a distinction between flow visualization using integral objects such as streamlines, streamtubes, streamribbons, and streampolygons [21] and dense, integration-based flow visualization techniques such as Spot Noise [26] and LIC [1, 25]. However, these two approaches are closely coupled. Conceptually, the path from using integral objects

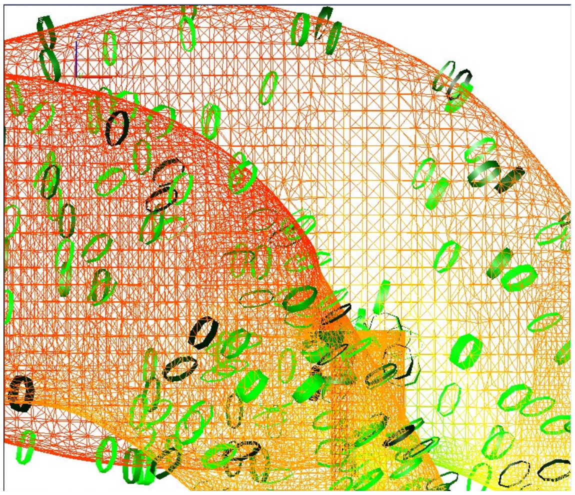

Figure 5: This image shows stream seeds as short pipe segments including a wire-frame context of the connecting pipes in the intake ports data set. In this way occlusion and image complexity are minimized (compare with figure 3).

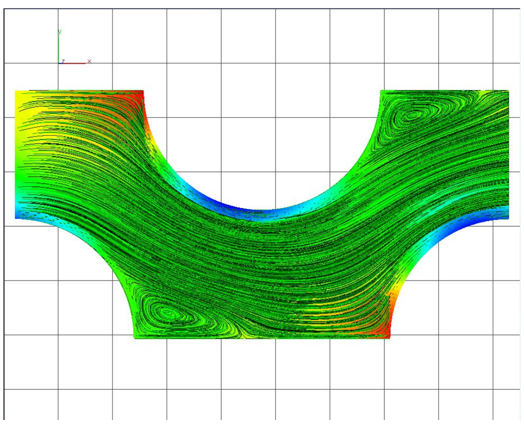

Figure 6: This image illustrates dense, integration-based visualization via a dense streamline seeding. (*CFD data courtesy of AVL*)

to dense, integration-based visualization is obtained via a dense seeding strategy. Densely seeded integral objects result in an image similar to that obtained by dense, integration-based visualization. The inverse path is obtained by using a tool such as a sparse texture for texture advection.

Figure 6 illustrates a dense, integration-based visualization resulting from a dense streamline seeding strategy in the context of a surface cut from a very small, thin fluid conduit where the fluid enters on the left and exits on the right. The streamrunner feature also adds a new dimension of interaction to this type of visualization strategy. By allowing the user to “rewind” the streamlines to seeds, visualization of the streamline seed placement used to generate the dense, integration-based visualization can be accomplished with ease. The seeds for the resulting streamlines in figure 6 are shown in figure 7.

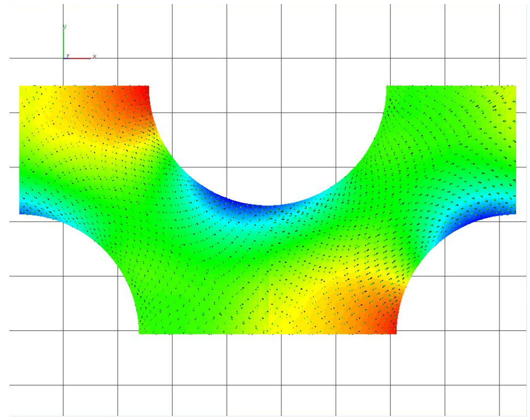

Figure 7: Using the streamrunner to “rewind” the streamlines in figure 6 to seeds, the user is able to visualize the streamline seed placement used in the dense, integration-based visualization with ease.

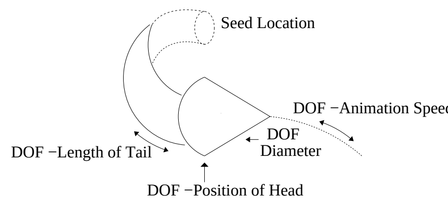

Figure 8: The streamcomet promotes 4 interactive degrees of freedom: (1) the position of the comet head along the path of integration, (2) the diameter of the comet head and tail, (3) the length of the comet tail, and optionally (4) the animation speed of the comets

4 STREAMCOMETS

Streamcomets are a natural extension of the streamrunner and follow a very intuitive metaphor. They offer four interactive degrees of freedom as shown in figure 8. The user is given interactive control over: (1) the position of the head along the integral path, (2) the diameter of the comet head and comet tail, (3) the length of the semi-transparent comet tail, and optionally (4) the animation speed of the comet along the path of integration. Coupled with more interactive degrees of freedom, streamcomets offer the advantage of showing local flow direction and curvature for static images. There is strong evidence to support the notion that flow visualization objects that show the direction of the local vector field improve the user’s ability to identify critical points and understand particle advection paths [13].

Figure 9 gives us an impression of what it is like to use the streamcomets for 3D flow visualization. We include the semi-transparent ring geometry as context information. We also apply a semi-transparent function to the comet tails and give them a glowing effect. The alpha value along each comet tail is a function of the distance to the comet head i.e., the further away from the head, the more transparent the tail.

4.1 ANIMATED STREAMCOMETS

Another useful feature is the option of animating the streamcomets. Conceptually, animating the streamcomets such that comet head position is automatically incremented along the path of integration, acts a visual search function. The viewer is able to use the anima-

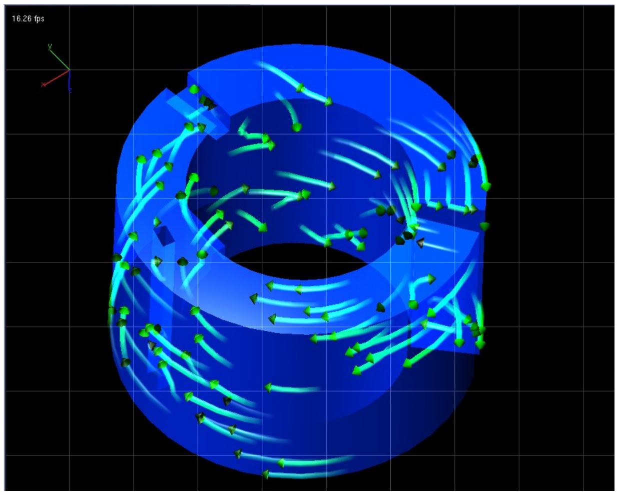

Figure 9: Here streamcomets are rendered in the context of a semi-transparent ring geometry. We add semi-transparency and a glowing impression to the streamcomet tails, whose transparency increases with the distance from the comet head. *(CFD simulation data courtesy of AVL)*

tion to search for optimal comet head positions. This is very useful when the user is (1) not sure where to position head, (2) searching for interesting features in the flow field, and (3) optimizing the other interactive degrees of freedom. We also give the user the option of interactively adjusting the animation speed.

We do not propose the streamrunner and streamcomet as standalone features. They are meant to be combined with other classic, 3D interaction techniques such as rotation, scaling, and translation. We also make use of the focus+context metaphor from information visualization so the user maximizes their global understanding of the vector field. Additional important features we have included are: (1) the option of choosing a non-uniform coloring scheme so colliding integral objects can be more easily distinguished, (2) turning on or off semi-transparent or wire-frame context information and, (3) interactively adjusting the streamline seeding density in the flow field.

4.2 THE STREAMRUNNER AND STREAMCOMETS FOR UNSTEADY FLOW VIZ

Here we propose the logical extension of streamcomets and the streamrunner for unsteady flow visualization. The comet glyph can intuitively encode time attributes for unsteady flow visualization. At the top of figure 10, we see a sample seed point whose geometric location is constant over time and from which streamcomets are injected into the flow, similar to a *streakline* – the line traced by a set of particles that have previously passed through a unique point in the domain [21]. As the comet ages (after being injected into the flow), the size of the head decreases as does a real comet when traveling through space or burning up in the earth's atmosphere. Also the length of the tail encodes the local instantaneous velocity at the comet's current position. The color of the comet head encodes the local temperature and the color of the comet tail reflects another scalar attribute of the flow such as pressure. If we represent comet tails using streamtubes, the local convergence and divergence of the flow may be encoded. If comet tails are represented using streamribbons, local vorticity is encoded. Ideally, the user is able to toggle between the two representations. We claim that the use of streamcomet glyphs for encoding attributes of the flow is more intuitive

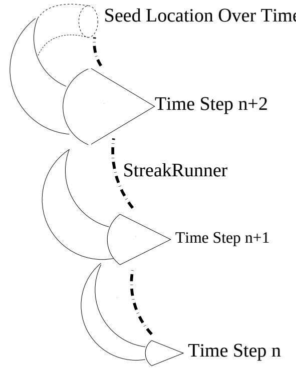

Figure 10: The use of the streamrunner and streamcomets for unsteady flow visualization. Comet heads shrink over time. i.e., the older the comet, the smaller the comet head. The interactive equivalent of a streakline is a streakrunner, which interactively controls the number of discrete time steps along the streakline defined by the series of comet heads.

than using other glyphs such as superquadric shapes [4].

The interactive analogue of a streakline is a *streakrunner*, also shown in figure 10. The streakrunner is an interactive control that defines the number of discrete time steps along the streakline. Such a line is shown in figure 10 connecting the comet heads.

5 ACCURATE VS. PRECISE INTEGRATION

In our experience with versatile, unstructured grids, the distinction between *accurate* and *precise* flow field integration has played an important role. We give the users the option of using an Euler integrator, known to have less accuracy than Runge-Kutta integrators of order 2 or higher, because the users' top priority is often the fastest interaction speed possible. Nevertheless, we still have the requirement of implementing an integrator that is very precise.

We can illustrate this idea with an analogy. Consider the scenario of a bowman shooting an arrow at a target. A very *accurate* shooter will always launch arrows that land near the bulls-eye within a certain deviation such as +/- 30cm. A very *precise* shooter will always launch arrows that land very close to each other, e.g. within +/- 10cm of each other at the same location, but not necessary within 30 cm of the bulls-eye.

Since our integration computation involves edge intersection tests with polygons that are on the order of 10^{-8}–$10^{-9}m^2$, we are required to use high precision methodology. We don't see this distinction of precision vs. accuracy being made in our review of the flow visualization literature [7, 16] and it is a *very* important distinction in other closely related disciplines [11, 12].

6 RESULTS

Figure 11 shows an extreme close-up of streamcomets on the surface of one of the intake ports from figure 11. We emphasize the importance of the user's ability to resize the streamcomets along arbitrary dimensions when zooming in and out of the unstructured data sets. The streamcomets shown here are far too small to view other parts of the intake ports such as the connecting pipes and the cylinder. Furthermore, it is important to note that changes to the diameter of the comet heads apply to the entire collection of stream-

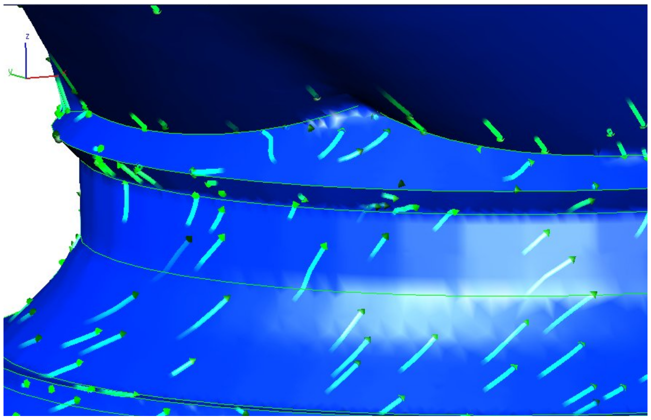

Figure 11: This is an extreme close up view of the streamcomets on the surface of one of the intake ports shown in figure 2.

comets, and are not applied on a per-comet basis. Applying size changes to individual comets would lead to misleading visualization results, e.g., the user may interpret different comet head sizes to be a reflection of scalar properties inherent in the flow field.

Figure 12 gives us another impression of what it is like to use the streamcomets for flow visualization in 3D. Here the streamcomets are shown in the semi-transparent context of the cylindrical combustion chamber from the intake port data set. Giving the user interactive control over the placement of the comet heads, the diameter of the comet heads and tails, the seeding density, and the length of semi-transparent comet tails, affords the user a very good opportunity to see inside the flow field.

7 CONCLUSIONS AND FUTURE WORK

We believe that the added interaction provided by the streamrunner and streamcomets is a very useful tool for flow visualization in 3D and within the domain of versatile, unstructured grids often resulting from CFD simulations. In fact, the streamrunner feature was requested by the head of the business marketing department at AVL. We believe that the fine, interactive control afforded by the streamcomets as well as the intuitive metaphor on which they are based makes them more suitable for 3D flow visualization than previous techniques. Furthermore, their simplicity makes them strong candidates for inclusion in future flow visualization software packages. These integral objects give a brand new level of control over to users investigating a vector field.

Future work could go in several directions including: (1) an implementation prototype of the streamrunner and the streamcomet for unsteady flow visualization including the introduction of a *pathrunner* – the unsteady equivalent of a streamrunner, a streakrunner – the interactive equivalent of a streakline, (2) a formal HCI evaluation of the perceptual effectiveness of the streamrunner and streamcomets for 3D flow visualization – an extension of the pioneering work by Laidlaw et al. [13] to include more flow visualization techniques, and (3) *dynamically annotated* streamcomets, an extension of the excellent work by Loughlin and Hughes [15]. With this methodology, the user would be able to select any streamcomet in the visualization and obtain dynamically updated quantitative information about local flow attributes such as velocity, location, temperature, pressure, shear, acceleration etc. in the form of a digital *Post-it* note. The user is then able to add their own information to

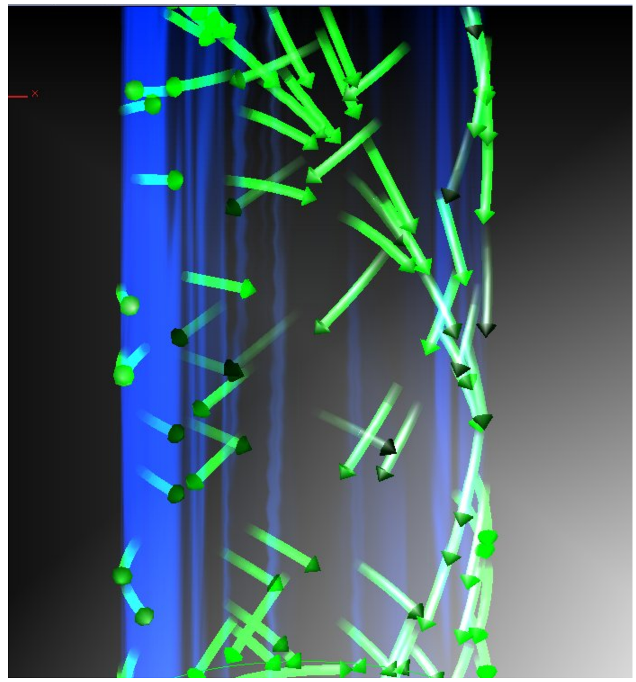

Figure 12: Here the streamcomets reflect the flow on the cylinder portion of the intake port data set 2. The user is afforded a new level of control when investigating fluid flow in 3D with streamcomets.

the analysis and add it to an archive from which they can continue future investigation.

8 ACKNOWLEDGEMENTS

We would like to thank all those who have contributed to financing this research, including AVL (`www.avl.com`) and the Austrian governmental research program called Kplus (`www.kplus.at`). We would also like to thank Hans Peter Blahowsky of AVL, and Karl Wieser of AVL for their very valuable contributions and feedback and a special thanks to Zoltan Konyha of VRVis for his very valuable proof reading.

References

[1] Brian Cabral and Leith C. Leedom. Imaging vector fields using line integral convolution. In *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 263–272, August 1993.

[2] Wim C. de Leeuw and Jarke J. van Wijk. A probe for local flow field visualization. In *Proceedings of the Visualization '93 Conference*, pages 39–45, San Jose, CA, October 1993. IEEE Computer Society Press.

[3] Helmut Doleisch and Helwig Hauser. Smooth Brushing for Focus and Context Visualization of Simulation Data in 3D. In *WSCG 2002, International Conference in Central Europe on Computer Graphics, Visualization and Digital Interactive Media*, pages 147–151, Plzen, Czech Republic, February 2002.

[4] David S. Ebert and Christopher D. Shaw. Minimally immersive flow visualization. In *IEEE Transactions on Visualization and Computer Graphics*, volume 7(4), pages 343–350. IEEE Computer Society, 2001.

[5] David S. Ebert, R. Yagel, J. Scott, and Y. Kurzion. Volume rendering methods for computational fluid dynamics visualization. In *Proceedings of the Conference on Visualization*, pages 232–239, Los Alamitos, CA, USA, October 1994. IEEE Computer Society Press.

[6] Anton L. Fuhrmann and Eduard Gröller. Real-time techniques for 3D flow visualization. In *IEEE Visualization '98*, pages 305–312. IEEE, 1998.

[7] Helwig Hauser, Robert S. Laramee, and Helmut Doleisch. State-of-the-Art Report 2002 in Flow Visualization. Technical report, VRVis Research Center, www.VRVis.at, February 2002. TR-VRVis-2002-003.

[8] William Hibbard and David Santek. Interactivity is the key. In *Proceedings of the Chapel Hill Workshop on Volume Visualization*, pages 39–43, May 1989.

[9] Bruno Jobard and Wilfrid Lefer. Creating evenly–spaced streamlines of arbitrary density. In *Proceedings of the Eurographics Workshop on Visualization in Scientific Computing '97*, volume 7, Boulogne-sur-Mer, France, April 28-30 1997. Eurographics, Springer-Verlag WienNewYork.

[10] Bruno Jobard and Wilfrid Lefer. Multiresolution flow visualization. In *WSCG 2001, International Conference in Central Europe on Computer Graphics, Visualization and Digital Interactive Media*, Plzen, Czech Republic, February 2001.

[11] F. Krückeberg. Arbitrary accuracy with variable precision arithmetic. In *Proceedings of the International Symposium on Interval Mathematics*, volume 212 of *LNCS*, pages 95–102, Freiburg, FRG, September 1985. Springer.

[12] F. Krückeberg and R. Leisen. Solving initial value problems of ordinary differential equations to arbitrary accuracy with variable precision arithmetic. In *Proceedings of the 11th IMACS World Congress on System Simulation and Scientific Computing (Oslo)*, volume 1. IMACS, 1985.

[13] David H. Laidlaw, Robert M. Kirby, J. Scott Davidson, Timothy S. Miller, Marco da Silva, William H. Warren, and Michale Tarr. Quantitative Comparative Evaluation of 2D Vector Field Visualization Methods. In *Proceedings of IEEE Visualization*, San Diego, CA, October 2001. IEEE.

[14] Robert S. Laramee. Interactive 3D Flow Visualization Using a Streamrunner. In *CHI 2002, Conference on Human Factors in Computing Systems, Extended Abstracts*, pages 804–805, Minneapolis, Minnesota, April 20-25 2002. ACM SIGCHI, ACM Press.

[15] Maria M. Loughlin and John F. Hughes. An Annotation System for 3D Fluid Flow Visualization. In *Proceedings of the IEEE Conference on Visualization*, pages 273–280, Los Alamitos, CA, USA, October 1994. IEEE Computer Society Press.

[16] Frits H. Post, Benjamin Vrolijk, Helwig Hauser, Robert S. Laramee, and Helmut Doleisch. Feature Extraction and Visualization of Flow Fields. In *Eurographics 2002*, Saarbrücken Germany, 9–10 September 2002. European Association for Computer Graphics. (to appear).

[17] Christof Rezk-Salama, Peter Hastreiter, Christian Teitzel, and Thomas Ertl. Interactive exploration of volume line integral convolution based on 3D-texture mapping. In *IEEE Visualization '99*, pages 233–240, San Francisco, 1999. IEEE.

[18] Martin Roth and Ronald Peikert. Flow visualization for turbomachinery design. In *Proceedings of IEEE Visualization*, pages 381–384. IEEE CS Press, October 1996.

[19] Martin Roth and Ronald Peikert. A higher-order method for finding vortex core lines. In *IEEE Visualization '98*, pages 143–150. IEEE, 1998.

[20] I. Ari Sadarjoen and Frits H. Post. Geometric methods for vortex extraction. In *Data Visualization '99*, Eurographics, pages 53–62. Springer-Verlag Wien, May 1999.

[21] William J. Schroeder, Kenneth M. Martin, and William E. Lorensen. *The Visualization Toolkit*. Prentice-Hall, Upper Saddle River, NJ 07458, USA, 2nd edition, 1998.

[22] Martin Schulz, Frank Recks, Wolf Bartelheimer, and Thomas Ertl. Interactive visualization of fluid dynamics simulations in locally refined cartesian grids. In *IEEE Visualization '99*, pages 413–416, San Francisco, 1999. IEEE.

[23] Qin Shen, Alex Pang, and Sam Uselton. Data level comparison of wind tunnel and computational fluid data dynamics data. In *IEEE Visualization '98*, pages 415–418. IEEE, 1998.

[24] Deborah Silver and Xin Wang. Tracking features in unstructured datasets. In *IEEE Visualization '98*, pages 79–86. IEEE, 1998.

[25] Detlev Stalling and Hans-Christian Hege. Fast and resolution independent line integral convolution. In *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 249–256. ACM SIGGRAPH, Addison Wesley, August 1995. held in Los Angeles, California, 06-11 August 1995.

[26] Jarke J. van Wijk. Spot noise-texture synthesis for data visualization. In *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 309–318, July 1991.

[27] Colin Ware and Glenn Franck. Evaluating stereo and motion cues for visualizing information nets in three dimensions. *ACM Transactions on Graphics*, 15(2):121–140, April 1996.

[28] Malte Zöckler, Detlev Stalling, and Hans-Christian Hege. Interactive visualization of 3D-vector fields using illuminated streamlines. In *Proceedings of IEEE Visualization '96, San Francisco*, pages 107–113, October 1996.