CWI Quarterly Volume 7 (2) 1994, pp. 131 – 146

Visualization Techniques for Vector Fields with Applications to Flow Data

Frits H. Post, Theo van Walsum, Willem C. de Leeuw, Andrea J.S. Hin
Delft University of Technology,
Faculty of Technical Mathematics and Informatics,
Julianalaan 132, 2628 BL Delft, The Netherlands.

This paper describes three different methods for visualization of vector fields: selective visualization, statistical visualization, and turbulent particles. The first method enables the user to focus on the interesting parts of a data set by creation, processing, and display of selected parts. The second technique visualizes statistical quantities about a region of interest rather than raw data. For statistical simulations of turbulent flows, the third method uses particles to show the erratic motions caused by turbulence. A brief technical description of each method is given, and practical examples from fluid dynamics demonstrate their utility. The first two methods are suitable to vector fields in general and illustrate how data reduction for visualization is achieved. The third is specialised for statistical turbulent flow simulations, where simplification in the simulation produces a reduced dataset, which requires a special visualization technique to partially reconstruct turbulent motion.

1. Introduction

The production of numerical data by experiments, observations, and numerical simulations, has vastly increased in the last decades. This has also created new problems of consuming these data, and scientific visualization is emerging as a new field of research to help solve these problems. Scientific visualization can be defined as the use of computer generated images and interactive techniques for exploration, interpretation, and presentation of scientific data.

Fluid dynamics is an application area with high and urgent demands in visualization. In fluid dynamics research, the patterns resulting from insertion of ink, smoke, or particles have long been observed and recorded. Recently,

computational fluid dynamics (CFD) research has flourished with improved numerical modelling and solution techniques, and with today's powerful computational resources these permit three-dimensional time-dependent flow simulations on a scale that was hard to imagine only ten years ago.

The dominant data-type produced by flow simulations is a vector field, most commonly a velocity field. The data are usually defined on the nodes of a computational grid used in a numerical simulation. The visualization of these data presents an interesting challenge to scientific visualization research. Common techniques such as arrow plots and streamlines work well with 2-D fields, but they are less suitable for 3-D data. The fundamental reason for this is that there is no natural visual representation for 3-D vector fields. The task is to construct new visual representations that are meaningful to human observers. Scientific visualization researchers have responded by developing a range of new techniques for visualization of vector field and flow data. Overviews are given elsewhere [1, 2].

In this paper we will present three different visualization techniques that have recently been developed at Delft University of Technology. The three techniques are described in more detail in separate papers [3, 4, 5]. All three techniques are in different ways concerned with reduction and simplification of the data or the complexity of the simulation. The amounts of data are too large and the simulations are too complex to allow display of 'raw data'.

The first technique allows the user to select and visualize only parts of the data that are considered interesting, as defined by the application area and the specific research problem. This approach, called 'selective visualization', is based on the idea that the researcher is able to specify a criterion of interest, expressed as a selection expression, which is then used to filter the data and extract regions where the interest criterion is satisfied. The selected regions are then used for further processing in the visualization process.

When an region of interest has been defined, the second technique uses statistics and iconic visualization to characterise the vector field in this region. With this technique, a vector field can be explored using aggregate statistical properties about that region, such as mean, variance, or distribution function. The visualization will direct the user to regions where interesting patterns occur, to move the region of interest, and to focus by reducing the size of the region.

While this technique provides aggregate data about a given region, the third technique does the reverse: it visualizes an instantiation of statistical distribution data from a turbulent flow. The statistical flow simulation produces data fields of mean velocity and turbulence intensity. For visualization, particles are used to show both the smooth motions of the mean flow and the random-walk motions of the turbulent flow.

The aim of the first two techniques is to reduce the amount of data to be visualized, leading to visualizations that are more clear and focused. With the third technique, a simplification has been made during the numerical flow simulation, by using statistical methods and avoiding the detailed modelling of small-scale turbulent motions.

In the following sections we will give a brief technical description of each of the three visualization techniques, and demonstrate their utility with practical examples. At the end of the paper, we will present some conclusions and discuss current and future work.

2. Selective Visualization

There are several techniques to reduce the amount of data in the visualization process. Contraction operations reduce the dimensionality of a data field: instead of a velocity vector one uses a scalar variable for visualization, e.g. a vector component or the vector magnitude. Slicing operations reduce the spatial dimensions of the domain on which data are to be displayed. Other techniques are probing techniques, where the data is displayed on a position or area selected by the user.

A more general approach to slicing and probing techniques can be conceived. We report a method for selective visualization of scientific data. First, a technique to create *selections*, or selected regions in a dataset, is described [6], and then we will describe how these selected regions can be used to selectively visualize data [3]. We will conclude with two applications of selective visualization on fluid flow data sets.

1.2. Selection of regions

The selective visualization approach uses selections to create better visualizations. Selections contain information on which data is considered relevant or interesting to the scientist; this information is used in the visualization mapping, as is explained in the next section. This section describes how regions are selected, and how selections are represented.

The data to be visualized is assumed to be represented on the nodes of a computatonal grid, and the selection technique also operates on these grid nodes. We represent a selection as a Boolean dataset on the grid, where each Boolean value indicates whether a grid node belongs to the selection or not. A selected cell can be defined as a cell of which at least n (some threshold value) of its corner nodes are selected, and a selected region as a set of connected selected grid nodes.

Using this representation, it is relatively simple to select regions that satisfy complex criteria based on local data values: selections can be generated by evaluating a Boolean *selection criterion* for each grid node. If the Boolean criterion evaluates to true, the grid node is selected, if it evaluates to false, it is not selected. The possibilities of such a selection generation depend on the elements that can be used in the Boolean criterion. We allow constants, node coordinates, node indices, data values and logical and comparison operators to appear in a criterion. Also, it is possible to use scalar, vector or gradient (divergence, rotation) functions to generate derived data, or functions to calculate threshold values, such as minimum or mean.

Examples of selection expressions are:

- `select = len(velo) > .5*max(len(velo))` : selects all nodes where

the length of vector `velo` is larger than .5 times the maximum length of `velo`,

- `select = (i == 3) && (dot(rot(velo),velo)>1)` : selects all nodes of the slice i=3, where the dot product of `velo` and the rotation of `velo` is larger than 1.

These expressions offer a general and flexible way of specifying regions of interest in a dataset.

2.2. Visualization mapping with selections

Once the selections have been determined, they can be used in the visualization mapping. The most straightforward use of selections is *clipping*. This can be performed by displaying data values only within selections. This enhances the visualization, as uninteresting data (data outside the selection) are not displayed; the attention is entirely focused on the selected data. A large number of existing visualization techniques can be easily extended to incorporate the function of selections. Examples are:

- Iso-surfaces: cell-based iso-surface construction algorithms like the Marching Cubes [7] can be extended with a test whether the cell is a selected cell, so the part of the iso-surface that is outside a selection can be clipped. An advantage of clipped iso-surfaces is that the area inside the iso-surface can be inspected.
- Orthogonal slices: can be clipped against a selection in the same way as iso-surfaces. Here the advantage is that uninteresting parts of the slices are not visible, and therefore cannot hide other parts of the visualization.
- Arrow plots: displaying a 3-D vector field by drawing arrows at each grid node does not reveal any information because the visualization is cluttered by a huge number of lines. However, by displaying arrows at selections, one can inspect the velocity field at the interesting regions.

A more sophisticated use of selections is *parameter extraction*. As visualization algorithms need parameters (iso-surfaces need an iso-value, streamline generators need starting positions), selections can be used to calculate such parameters. By sampling selections, and feeding these samples in a streamline generator, it is possible to generate streamlines in or through selections. This reduces the need for elaborately probing a fluid flow dataset, one can directly visualize streamlines in interesting regions. In an analogous way, the centre of a selected region can be fed into an orthogonal slicer, thereby generating slices through interesting regions.

A different approach is to visualize selections in a non-standard way. We are currently working on *iconic visualization* of selections using parameterised symbolic representations of selections. By calculating attributes (aggregate values) for selections, and mapping these values onto icon parameters, one can create an abstract visualization, consisting of a number of iconic representations for

selections. Such visualizations are very suitable for processing time-dependent datasets, as the amount of data to be visualized is greatly reduced.

2.3. Selective visualizations in fluid flow applications

Selective visualization has been applied for the comparison of three scalar variables in turbulent flow. In the literature, several criteria are described to classify regions in turbulent flow. One of these criteria, R, has been compared with a combination of two criteria, R_2 and II_D, for a direct numerical simulation of turbulent flow. To check whether the regions of high values for R_2 and II_D coincide with regions of high values for R, we select regions with high value for R_2 (using `select = R2 > 1.`), and visualize II_D within these regions, using colored crosses (see Figure 1a). This visualization can be compared with isosurfaces at a high value of R, as shown in Figure 1b. Visual inspection shows that the region with high values for both R_2 and II_D (the red crosses) are inside the isosurfaces for R, from which it can be concluded that criterion R can be used instead of a combination of R_2 and II_D. An extra check was done by selecting regions with high values for both R_2 and II_D and low values for R. This selection was empty (there where no nodes that satisfied this selection criterion), which confirmed the previous conclusion.

Another example is shown in Figure 2. Here, we want to visualize the spiraling flow pattern that occurs at the step. It is very time-consuming to visualize this pattern by manually probing the flow. However, by selecting regions with high normalised helicity density, and generating starting positions for streamlines from this selection, such a visualization can be generated more or less automatically. The normalised helicity density H_n has been calculated using velocity $\mathbf{u}$ in the following way:

$$H_n = \frac{\mathbf{u} \cdot (\nabla \times \mathbf{u})}{|\mathbf{u}|\,|\nabla \times \mathbf{u}|}$$

where $\nabla \times \mathbf{u}$ is the rotation of the vector field. Streamline starting points were generated in the region where $|H_n| > 0.6$.

3. Statistical visualization

In the previous section we discussed how selective visualization is used to reduce the amount of data to be visualized. In this section we will discuss statistical visualization techniques that are applied to selections of data. Selections are the result of techniques described previously or by interactive techniques. Statistical visualization of the data is independent of the size and shape of the region of interest. Using statistical methods for data representation in combination with techniques to specify a region of interest, a user can investigate the data at different levels. If the information presented at a certain level invites further investigation, the region of interest is reduced, thereby allowing more detailed investigation.

The statistical information is presented to the user by icons. Other iconic techniques for vector data are arrow plots and the tensor probe [8] which visualizes a velocity vector and gradient quantities at a point. Generally, icons are

used to represent local information of a single point. However, we constructed an icon showing the characteristics of a selected spatial domain.

3.1. Statistical data exploration

The process of visualising statistical data of vector fields can be described by a sequence of steps (see Figure 3). First, a region in space is selected: for simplicity an axis-aligned box is used in the following discussion, but the techniques described can be easily generalised to arbitrarily shaped selections as described in section 2. The next step in the pipeline is *projection*. The spatial dimensions in the data are projected to a point, i.e. relative positions within the domain of interest are ignored. After projection we calculate statistical properties: the individual values are replaced by a statistical description. A further reduction of data takes place at this step. The amount of reduction can be varied; the user can choose to generate only the average or more complex characteristics which show more detailed information on the distribution.

The last step in the process is the mapping of the calculated statistical properties to geometric primitives. For different statistical characterisations of the data suitable mappings must be found to obtain an understandable picture of the data. In order to apply statistics to a vector field we consider the distribution of vector values in the region of interest: a function which maps velocities to a probability. This is a three dimensional scalar function. Direct visualization of this function by existing techniques, for example volume rendering or iso-surfaces, is not intuitive and hard to interpret. Therefore we use the distribution function as a base for the calculation of statistical properties. The most important property is the mean of the function. If we assume a rectangular grid with vector values given at grid nodes, and tri-linear interpolation inside the cells, then the average velocity in a cell is the average of the velocities at the corners of the cell. By summation of cell averages, the average over the region of interest is calculated. Special care has to be taken for cells partly inside the region of interest.

Representing a field by an average is useful under certain circumstances, however information about spatial distribution of data is lost. More information is represented by higher order moments of the distribution. The multidimensional equivalent of the second-order moment (standard deviation) is the variance/covariance matrix (VCM). This matrix is a symmetric second-order tensor. Using principal components analysis from multivariate analysis [9], the principal components of the distribution are calculated. Principal components are the eigenvectors of the VCM. The VCM is visualized by an ellipsoid [10] (see Figure 4a). The three axes of this ellipsoid reflect the size and direction of the principal components of the VCM. This ellipsoid can be interpreted as a surface of equal probability for the normal distribution with equal average and variance as the given distribution.

A better approximation of the distribution function is achieved using higher order moments of the distribution function. The results of these calculations are higher order tensors. Interpretation of this data in relation to the actual

vector field is difficult. A better approach in our view is the use of simple fitting or smoothing functions to model the velocity distribution. In case of bimodal or other distributions which are badly modelled by a normal distribution, this is a good alternative. The chosen fitting functions can be of arbitrary complexity, with the distribution function itself as a limiting case. Visualization of the (approximated) distribution function is possible using iso-surfaces of probability density. If simple functions are used this is easier to understand than direct visualization of the distribution function with iso-surfaces.

A different approach to represent the distribution function is by showing a number of samples, instead of a continuous representation. If a box is used as a region of interest, a regular grid of sample points is used to take samples. The values found can be used to generate a 3-D scatter plot of the distribution function in velocity space. The velocity samples are shown as points in space. They can be represented as dots or small spheres. This leads to ambiguity as a result of the projection onto the screen. Therefore we used cones instead; this partly solves the projection ambiguity.

3.2. The flux probe and finding vortices

In this section we will show two examples of visualization techniques based on this representation, applied to flow visualization.

If a polygon is specified as a region of interest, then the average velocity over the polygon shows the average flow through the area of the polygon (see Figure 4a). In incompressible flow the component of this vector perpendicular to the polygon is the mass flux through the polygon. Flux is important in many flow problems such as problems dealing with convection.

Instead of showing velocity, other vector quantities defined over the region of interest can be selected for visualization. Here we will use the curvature vector of a streamline through each point to locate vortex cores in the data. First the curvature vector is calculated in a number of sampling positions in the region of interest, using the method described in [8]. With this value, the osculating circle to the streamline is determined for each sample. The centre of these circles are estimates for the position of the vortex core. By drawing these points in the flow region, vortex cores can be located interactively (see Figure 4b). This only works if the flow in the region of interest is mainly determined by a vortex.

4. Turbulent Particles

In the previous section we described the visualization of a data volume by extracting aggregate values with statistical techniques. For turbulent particles, we reverse the process by creating instances or realisations of particle motion modelled with statistics. Particles have long been used in experimental visualizations [11], and they are now also used in computer graphics visualizations [12, 13] to capture the essence of the fluid motion. The motion of particles in turbulent flow and their distinct behaviour is discussed here.

Most numerical simulations generate a velocity field, from which motion of

particles can be derived with standard tracing algorithms. Indeed, a velocity field generated by a direct numerical simulation provides information on turbulent motion up to the smallest scale modelled. However, direct numerical simulation of turbulence is very expensive, and large-scale 3-D simulations are not feasible even with the most powerful computing resources. Therefore, simulations are often based on averaged equations, the so-called Reynolds-averaged equations. But the velocity field generated by these simulations is not sufficient to represent turbulent motion [14]. These models provide information on turbulence in two separate data fields: a mean velocity field, and a turbulence intensity field. Turbulence intensity is represented by quantities such as turbulent kinetic energy or eddy-diffusivity. Our task is now to visualize the two combined fields as the turbulent motion of fluid particles.

The proposed method uses random-walks for particle motion animation. This random-walk perturbation is more than just a visual effect, as it is based on physical and statistical theory of turbulent flow. The method captures two essential aspects of turbulent motion: dynamics and randomness. Other characteristics, such as the rotational nature of the turbulent motions, are not shown. The visualization shows the distribution of local motion dynamics as it is based on a statistical model of turbulence. A full reconstruction of the actual turbulent motion of individual fluid particles is not possible, as the information for this is not available in the statistical data.

4.1. Random-walk particle motions

The Reynolds decomposition applied in Reynolds-averaged models decomposes a turbulent flow $\mathbf{u}$ as a function of location $\mathbf{x}$ and time t into a mean flow and a fluctuating motion:

$$\mathbf{u}(\mathbf{x}, t) = \overline{\mathbf{u}}(\mathbf{x}, t) + \mathbf{u}'(\mathbf{x}, t)$$

Here, $\overline{\mathbf{u}}(\mathbf{x}, t)$ describes the convective motion, and $\mathbf{u}'(\mathbf{x}, t)$ the turbulent motion. Turbulence can be viewed as a random perturbation of a mean velocity field which can be translated into a spatial displacement of a particle over a time interval.

In Figure 5 the determination of subsequent positions $\mathbf{x}_i$ of a particle at time t_i is shown for a non-turbulent flow and a turbulent flow. In the convective part of the turbulent flow the local mean velocity $\overline{\mathbf{u}}$ is involved. The compound velocity $\mathbf{u}_i$ at $\mathbf{x}_i$ in a flow is the vector sum of a mean velocity $\overline{\mathbf{u}}_i$ and a random perturbation vector $\mathbf{u}'_i$. The direction of the perturbation vector is chosen randomly on a circle (or a sphere in 3-D), while the magnitude of the local turbulence intensity determines the radius of the circle or sphere. To clearly show the effect of the perturbation, the velocity in Figure 5a and the mean velocity in Figure 5b are taken to be constant.

We use a statistical approach to describe the fluctuations of a particle in a turbulent flow, so that the convective motion and random perturbation can be expressed in terms of physical quantities. The particle motion can be described by a random-walk model. These models are based on a fluctuation equation,

which contains a drift coefficient and a random component which leads to perturbed paths of individual particles through a flow. The random walks of a large collection of particles show the effect of turbulent dispersion. This dispersion process is described by a forward-diffusion equation, which is also known as the Fokker-Planck equation (FPE). The FPE is thus closely related to the behaviour of an individual particle as described by the fluctuation equation.

In fluid mechanics, the advection-dispersion equation (ADE) describes the transport of material [14]. The FPE and ADE can be written in a similar form, so that the probability density function of particles in the FPE coincides with the material concentration in the ADE, as described in [5]. Because the FPE and ADE both describe material transport, the similarity between both equations can be used to express the coefficients for the fluctuation equation in terms of physical quantities from the ADE [15]. The drift coefficient of the fluctuation equation is expressed as the mean velocity $\overline{\mathbf{u}}(\mathbf{x})$ plus the spatial variability of eddy-diffusivity $\mathbf{E}(\mathbf{x}, t)$ over a time step. The random component is derived from the eddy-diffusivity field $\mathbf{E}(\mathbf{x}, t)$ and a distribution $d\mathbf{W}$. The distribution consists of three Wiener-Levy processes W_x, W_y, and W_z, each known as Brownian motion, with mean zero and standard deviation t. In the general case the eddy-diffusivity $\mathbf{E}$ is a vector quantity specifying non-isotropic turbulence for the three coordinate directions. In the special case of isotropic turbulence, E_x, E_y, and E_z are everywhere equal, and the eddy-diffusivity field can be represented by the scalar field E.

4.2. The Bay of Gdansk

We used the particles to show a flow in the Bay of Gdansk, a coastal area of Poland connected with the Baltic Sea. Figure 6 shows an overview of the area from the South with the curved coast line meeting the long-stretched peninsula of Hel at the left. A 3-D layered hydrodynamical simulation based on Reynolds-averaged equations has been performed on the curvilinear mesh of the area with low wind from the southern direction. The simulation produced a wind-driven stationary flow field, shown by the particles in the figure. An anisotropic eddy-diffusivity field is produced by solving the turbulence with two coupled conservation equations for kinetic energy and the rate of dissipation, known as a $k - \varepsilon$ model.

We use a particle tracer to determine paths of mean velocity by a conventional time-stepping integration method [16], and to calculate paths of turbulent particles by applying the random walks described above. A particle renderer uses the particle path data to generate animations in real time. The fluctuations are perceived as erratic particle motions, and the spatial variation of turbulent motion is clearly visible.

In Figure 7 a view from the North-East of the Bay is shown. The area has been scaled up by a factor of 250 in the vertical direction. Particles are released close by the coast at regular time intervals from a single source. The white particles act as in a laminar flow by only considering the mean velocity field, and thus create overlapping paths directed into the Bay. The motion of

the pink particles is also affected by eddy-diffusivity. Their paths are perturbed and a substantial number of them tend to float towards an area of circulation. This is a consequence of the displacement of a turbulent particle, relative to the mean velocity path.

5. Conclusions

In the previous sections we have presented three different visualization techniques for vector field data, applied to fluid flow visualization: an approach to selective data visualization, visualization of aggregate statistical quantities of an area of interest, and the use of particles to visualize a turbulent flow simulated by a statistical turbulence model. The techniques represent three different approaches to 3-D vector field and flow visualization, with reduction and simplification as unifying concepts. The first and second techniques are not restricted to fluid dynamics data, while the third is dedicated to a particular class of CFD simulations.

Our current research efforts include several extensions of the work reported here, as well as new directions. The selective visualization approach is extended by development of algorithms to compute aggregate quantitative attributes of selections, allowing pattern matching and iconic visualization. The particle-based turbulence visualization technique is extended to generate scalar fields of particle concentrations, which are visualized using volume ray casting.

Future research will focus on two topics: feature-based visualization and comparative visualization [17]. Feature-based visualization concerns the recognition and extraction of important patterns from the data, using geometric processing to extract geometric objects, or techniques derived from image processing and computer vision. In comparative visualization [18], visual techniques are developed to support comparison of data from different sources describing the same situation. It can be applied to experimental validation of numerical simulations, by comparing experimental and simulated data.

As we believe that a single answer to the problems of 3-D vector field and flow visualization does not exist, scientists will always require a range of different visualization tools. Development of new methods will continue, inspired by the users. Results will also be evaluated by the users, who will eventually decide which techniques are most valuable for their application.

Acknowledgements

We thank Arthur Mynett, Henk van den Boogaard, and Rob Uittenbogaard (Delft Hydraulics) for useful discussions of turbulence modelling, and Ari Sadarjoen for research on particle trace algorithms. We gratefully acknowledge the support from Delft Hydraulics for Andrea Hin's research. Willem de Leeuw's part of this work is supported by the Netherlands Computer Science Research Foundation with the financial support from the Netherlands Organisation for Scientific Research(NWO).

References

1. F.H. Post and T. van Walsum (1993). Fluid flow visualization. In H. Hagen, H. Mueller and G.M. Nielson, editors, *Focus on Scientific Visualization*. Springer Verlag, Berlin.
2. F.H. Post and J.J. van Wijk (1994). Visual representation of vector fields: Recent developments and research directions. In L.J. Rosenblum et al., editor, *Frontiers in Scientific Visualization*. Academic Press Ltd., London.
3. T. van Walsum and F.H. Post (1994). Selective visualization of vector fields. To be presented at the Eurographics '94 conference, Oslo, Norway.
4. W.C. de Leeuw and F.H. Post (1994). A statistical view on vector fields. *Proceedings of the 5th Eurographics Workshop on Visualization in Scientific Computing*, Rostock, Germany.
5. A.J.S. Hin and F.H. Post (1993). Visualization of turbulent flow with particles. In G.M. Nielson and R.D. Bergeron, editors, *Proceedings Visualization '93*, pages 46–52. IEEE Computer Society Press, Los Alamitos, CA.
6. T. van Walsum (1993). Content-based grid node selection for vector field visualization. *Proceedings of the 4th Eurographics Workshop on Visualization in Scientific Computing*, Abingdon, UK.
7. W.E. Lorensen and H.E. Cline (1987). Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics*, **21**(4):163–169.
8. W.C. de Leeuw and J.J. van Wijk (1993). A probe for local flow field visualization. In G.M. Nielson and R.D. Bergeron, editors, *Proceedings Visualization '93*, pages 39–45. IEEE Computer Society Press, Los Alamitos, CA.
9. M.G. Kendall (1965). *A Course in Multivariate Analysis, Introduction*. Chas. Griffin.
10. D. Silver, N. Zabusky, V. Fernandez and Gao M (1991). Ellipsoidal quantification of evolving phenomena. In N. Patrikalakis, editor, *Scientific Visualization of Physical Phenomena*, pages 573–588. Springer Verlag.
11. W. Merzkirch (1987). *Flow Visualisation, second edition*. Academic Press Inc., New York.
12. J.J. van Wijk (1993). Flow visualization with surface particles. *IEEE Computer Graphics and Applications*, **13**(3):18–24.
13. P.G. Buning (1989). Numerical algorithms in CFD post-processing. In *Computer Graphics and Flow Visualization in Computational Fluid Dynamics*. Von Karman Institute Lecture Series 1989-07, Brussels, 1989.
14. H. Tennekes and J.L. Lumley (1972). *A First Course in Turbulence*. MIT Press, Cambridge.
15. H.F.P. van den Boogaard, M.J.J. Hoogkamer and A.W. Heemink (1993). Parameter identification in particle models. *Stochastic Hydrology and Hydraulics*, **7**(2):109–130.
16. A. Sadarjoen, T. van Walsum, A.J.S Hin and F.H. Post (1994). Particle tracing algorithms for 3d curvilinear grids. *Proceedings of the 5th*

Eurographics Workshop on Visualization in Scientific Computing, Rostock, Germany.

17. L. Hesselink, F.H. Post and J.J. van Wijk (1994). Research issues in vector and tensor field visualization. *IEEE Computer Graphics and Applications*, **14**(2):76–79.
18. H.-G. Pagendarm and F.H. Post (1994). Comparative visualization – approaches and examples. *Proceedings of the 5th Eurographics Workshop on Visualization in Scientific Computing*, Rostock, Germany.

(a) (b)

FIGURE 1. Relation between three scalar variables in turbulent pipe flow; a) II_D is coloured in regions where R_2 is high; b) an iso-surface for R is added. Data courtesy: J.M.J. den Toonder, Faculty of Mechanical Engeneering, Delft University of Technology.

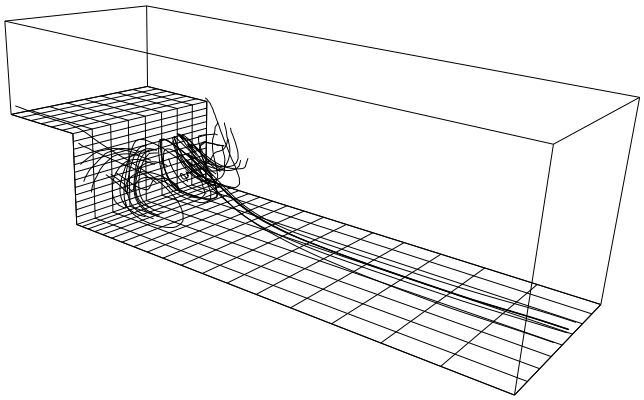

FIGURE 2. Selective visualization of flow in a backward facing step geometry: streamlines through regions were the absolute value of the helicity density is high. Data courtesy: Faculty of Technical Mathematics and Informatics, Delft University of Technology.

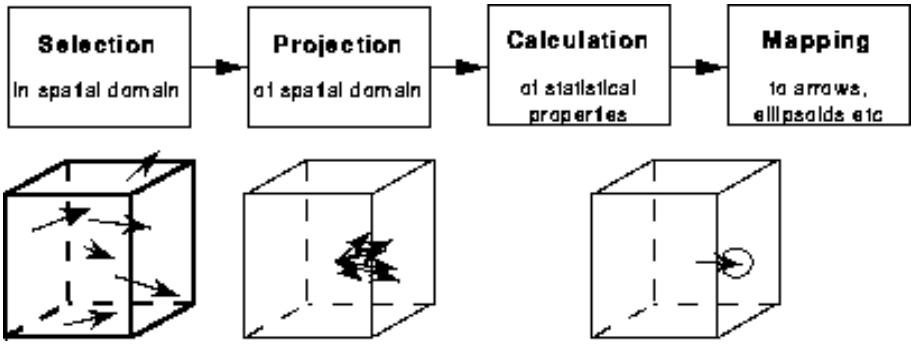

FIGURE 3. Generation of a statistical representation

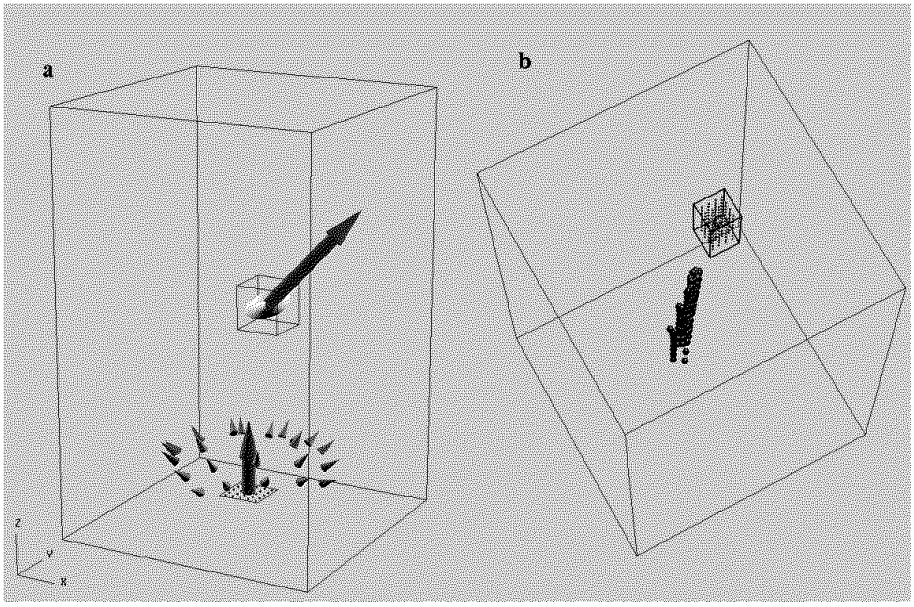

FIGURE 4. Examples of the use of statistical probes: (a) A flux probe (bottom) and a box shaped region of interest showing the average velocity arrow (red), the variance-covariance matrix (yellow) and some samples (green); (b) the vortex probe. Data courtesy: R. Wilhelmson, NCSA

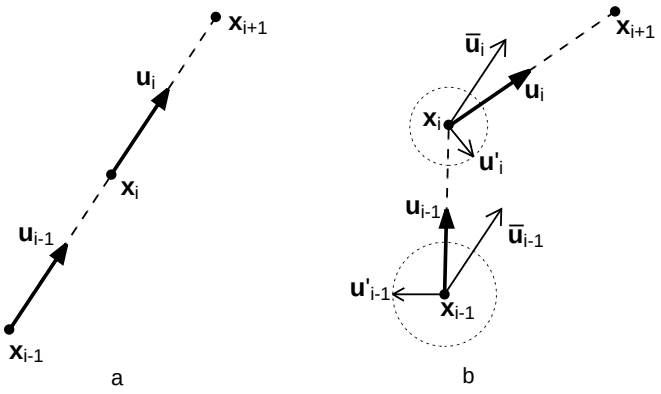

FIGURE 5. Particle path integration: a) in a non-turbulent flow and b) in a turbulent flow.

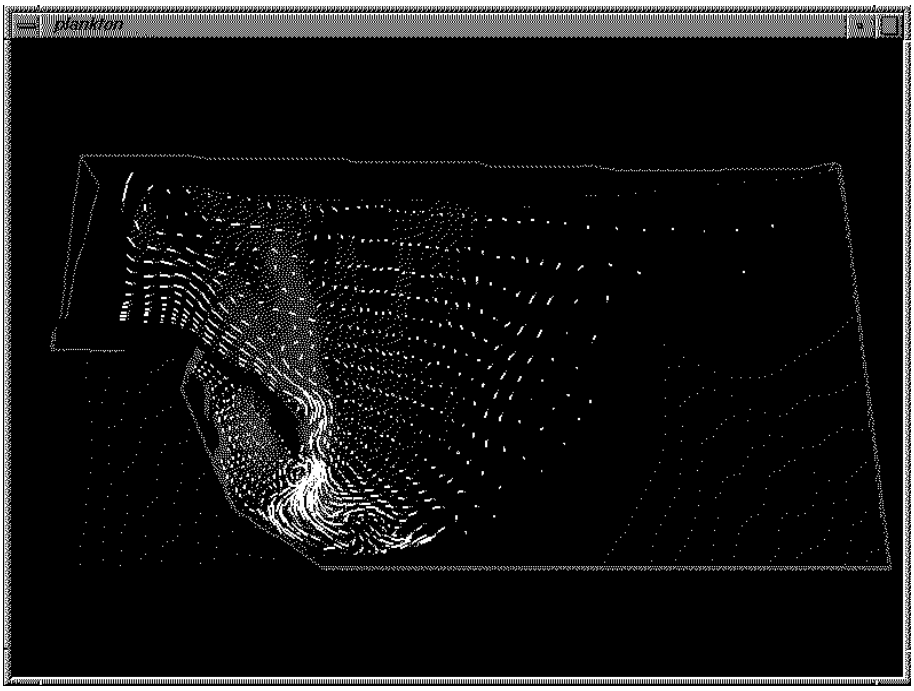

FIGURE 6. A view of the Bay of Gdansk from the South with particles showing the stationary mean velocity field. Data courtesy: M. Robakiewicz, Academy of Sciences, Gdansk.

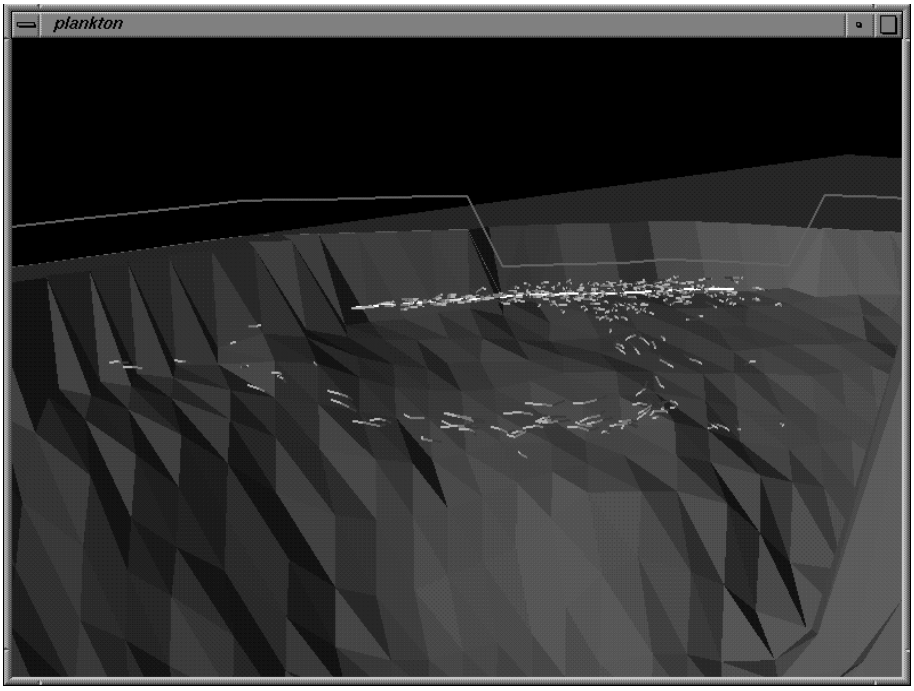

FIGURE 7. A view of the Bay from the North-East, with a single particle source.