CS 519: Scientific Visualization

Vector Field Visualization: Derived Quantities Glyphs

Eric Shaffer

Some slides adapted Alexandru Telea, *Data Visualization Principles and Practice*

Vector Field Visualization Techniques

1. **Scalar derived quantities**
   - divergence, curl, vorticity

2. **0-dimensional shapes**
   - hedgehogs and glyphs
   - color coding

3. **1-dimensional and 2-dimensional shapes**
   - displacement plots
   - stream objects

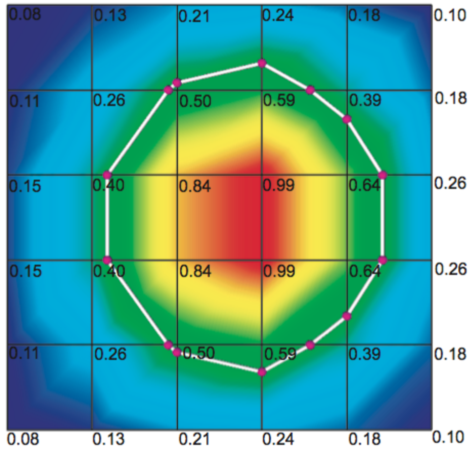
4. **Image-based algorithms**
   - image-based flow visualization in 2D, curved surfaces, and 3D

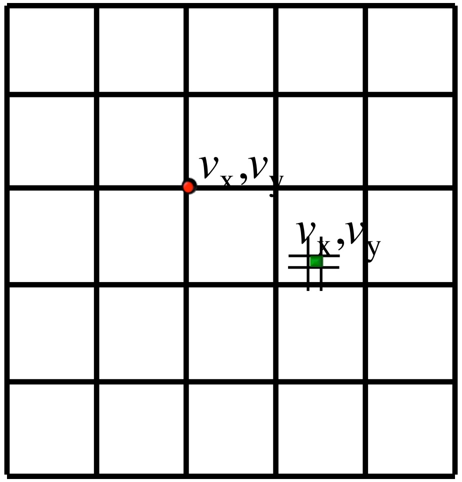
Vector Fields

Input data

- vector field $v : D \rightarrow \mathbf{R}^n$
- domain D 2D planar surfaces, 2D surfaces embedded in 3D, 3D volumes
- for typical scientific application: n=2 (fields tangent to 2D surfaces) or n=3 (volumetric fields)

Challenge: comparison with scalar visualization





Scalar visualization
- challenge is to map D to 2D screen
- after that, we have 1 pixel per scalar value

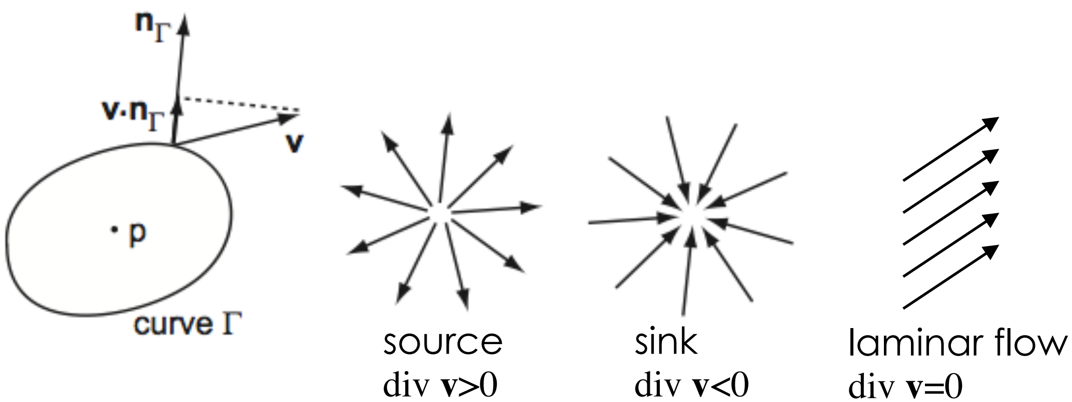
Vector visualization
- challenge is to map D to 2D screen
- after that, we have 1 pixel for 2 or 3 scalar values!

First Technique: Re-use Scalar Visualization Techniques

- compute derived scalar quantities from vector fields
- use known scalar visualization methods for these

Divergence $\text{div } \mathbf{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}$ equivalent to $\text{div } \mathbf{v} = \lim_{\Gamma \to 0} \frac{1}{|\Gamma|} \int_{\Gamma} (\mathbf{v} \cdot \mathbf{n}_{\Gamma}) ds$

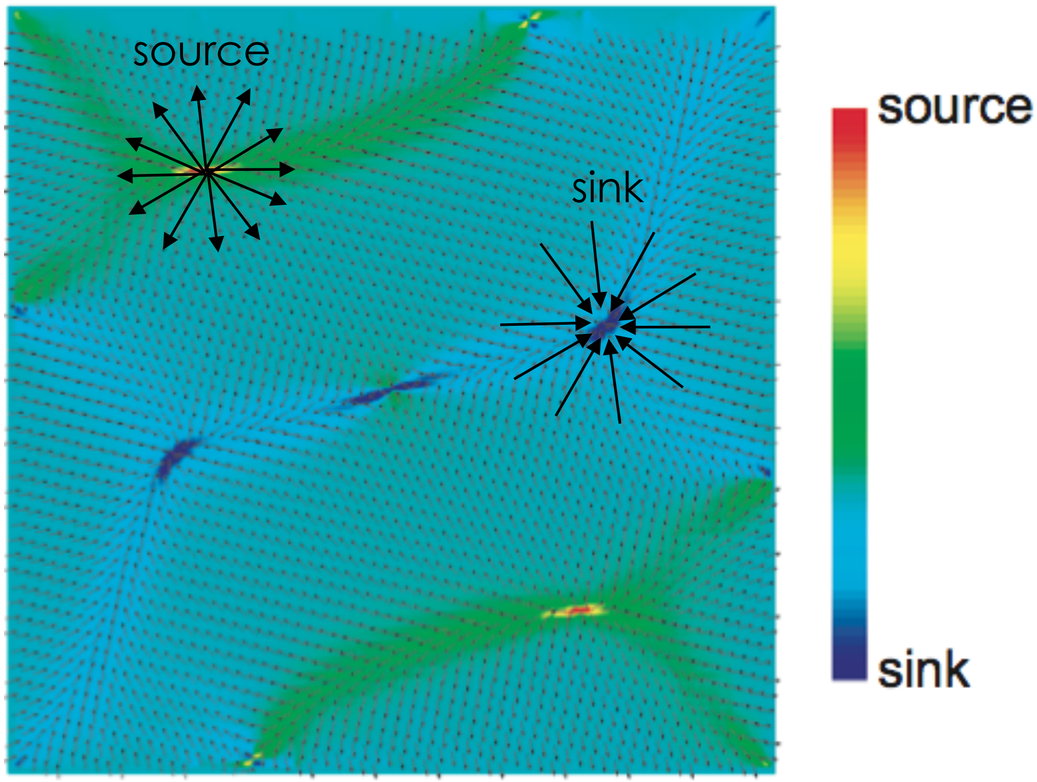
- think of vector field as encoding a fluid flow
- intuition: Degree to which the field converges or diverges at a point
- given a field $\mathbf{v} : \mathbf{R}^3 \to \mathbf{R}^3$, $\text{div } \mathbf{v} : \mathbf{R}^3 \to \mathbf{R}$ is



div $\mathbf{v}$ is sometimes denoted as $\nabla \cdot v$

Divergence

- compute using definition with partial derivatives
- visualize using e.g. color mapping



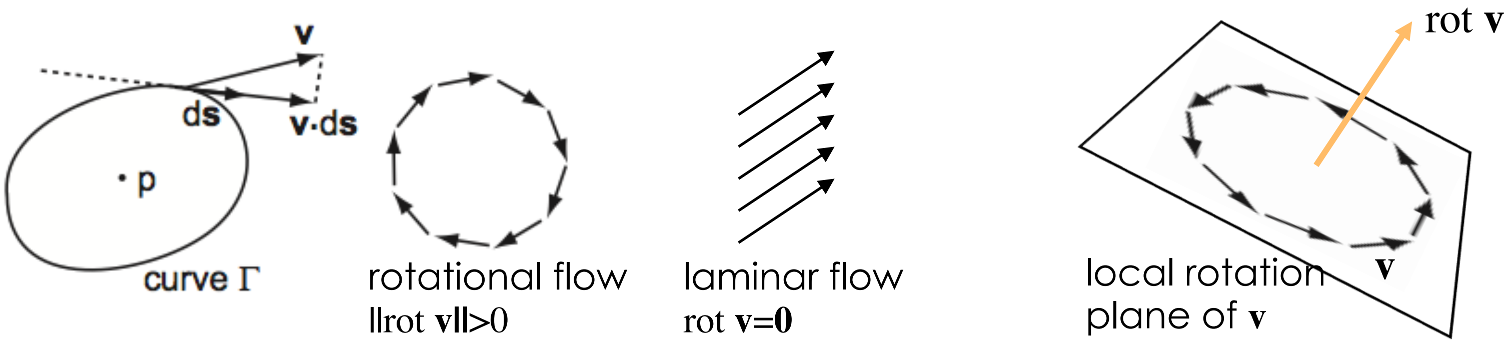
- gives a good impression of where the flow 'enters' and 'exits' some domain

Curl (Vorticity)

Curl (also called rotor)

- consider again a vector field as encoding a fluid flow
- intuition: how quickly the flow ‘rotates’ around each point?
- given a field $\mathbf{v} : \mathbf{R}^3 \to \mathbf{R}^3$, $\mathrm{rot}\ \mathbf{v} : \mathbf{R}^3 \to \mathbf{R}^3$ is

$$\mathrm{rot}\ \mathbf{v} = \left(\frac{\partial v_z}{\partial y} - \frac{\partial v_y}{\partial z}, \frac{\partial v_x}{\partial z} - \frac{\partial v_z}{\partial x}, \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y} \right) \quad \text{equivalent to} \quad \mathrm{rot}\ \mathbf{v} = \lim_{\Gamma \to 0} \frac{1}{|\Gamma|} \int_\Gamma \mathbf{v} \cdot \mathbf{ds}$$

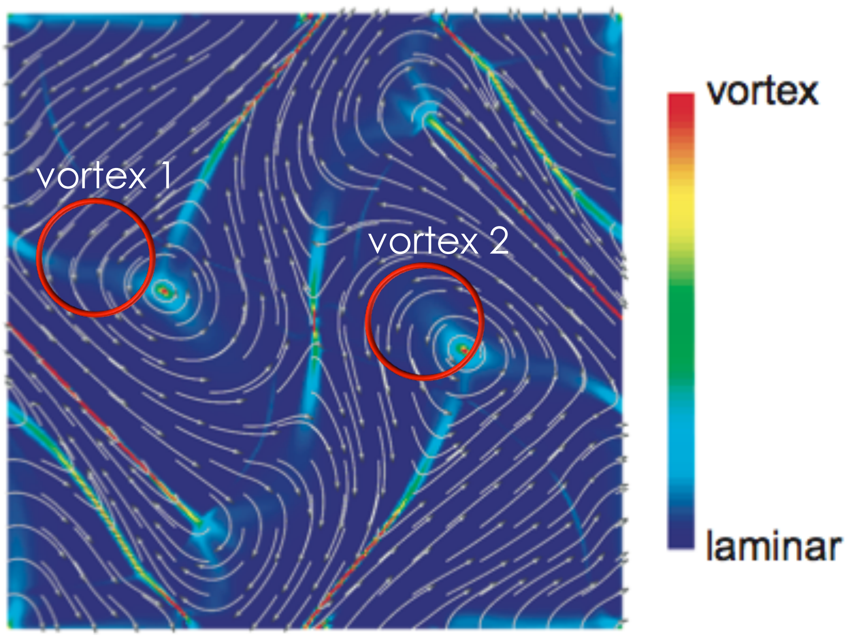


- rot $\mathbf{v}$ is locally perpendicular to plane of rotation of $\mathbf{v}$
- its magnitude: ‘tightness’ of rotation

rot $\mathbf{v}$ is sometimes denoted as $\nabla \times v$

Visualizing Vorticity

- compute using definition with partial derivatives
- visualize magnitude $\|\text{rot } \mathbf{v}\|$ using e.g. color mappin

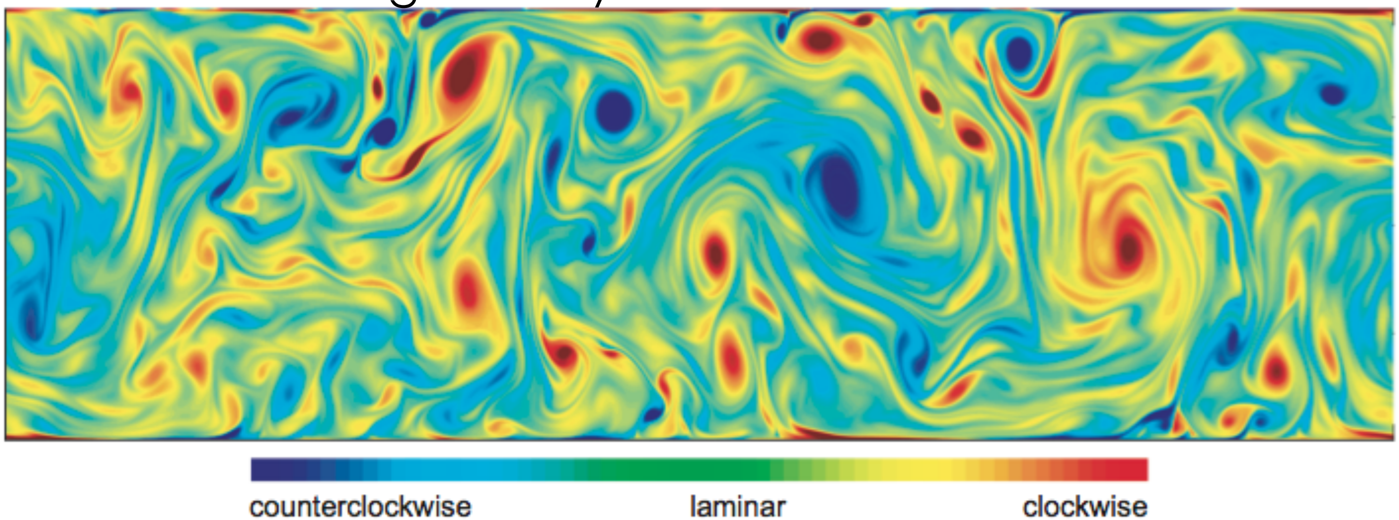


- very useful in practice to find vortices = regions of high vorticity
- these are highly important in flow simulations (aerodynamics, hydrodynamics)

Vorticity

Example of vorticity

- 2D fluid flow
- simulated by solving Navier-Stokes equations
- visualized using vorticity



Observations

- vortices appear at different scales
- see the 'pairing' of vortices spinning in opposite directions

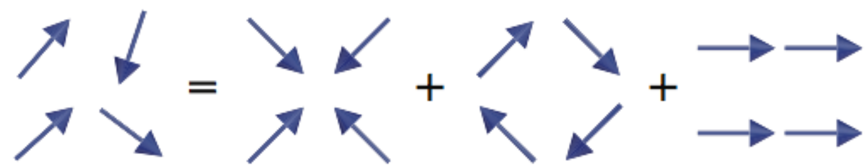
Compute 2D fluid flows in real-time:
A Simple Fluid Solver based on the FFT, J. Stam, J. of Graphics Tools 6(2), 2001, 43-52

Vector Field Decomposition

Helmholtz-Hodge theorem

- any vector field $\mathbf{v}$ can be uniquely decomposed into three components

$$\mathbf{v} = \mathbf{d} + \mathbf{r} + \mathbf{h}$$



$\mathbf{v}$ = $\nabla\varphi$ + $\nabla \times \Psi$ + $\mathbf{h}$

curl-free component $\mathbf{d}$ | divergence-free component $\mathbf{r}$ | divergence-free and curl-free component $\mathbf{h}$

$$\begin{aligned} \nabla \times \mathbf{d} &= 0 \\ \nabla \cdot \mathbf{r} &= 0 \\ \Delta \mathbf{h} &= 0 \end{aligned}$$

$\mathbf{d}, \mathbf{r}, \mathbf{h}$ are computed from two intermediate **potential fields** φ, Ψ

$\mathbf{d} = \nabla\varphi$ curl-free since $\nabla \times (\nabla\varphi) = 0$

$\mathbf{r} = \nabla \times \Psi$ divergence-free since $\nabla \cdot (\nabla \times \Psi) = 0$

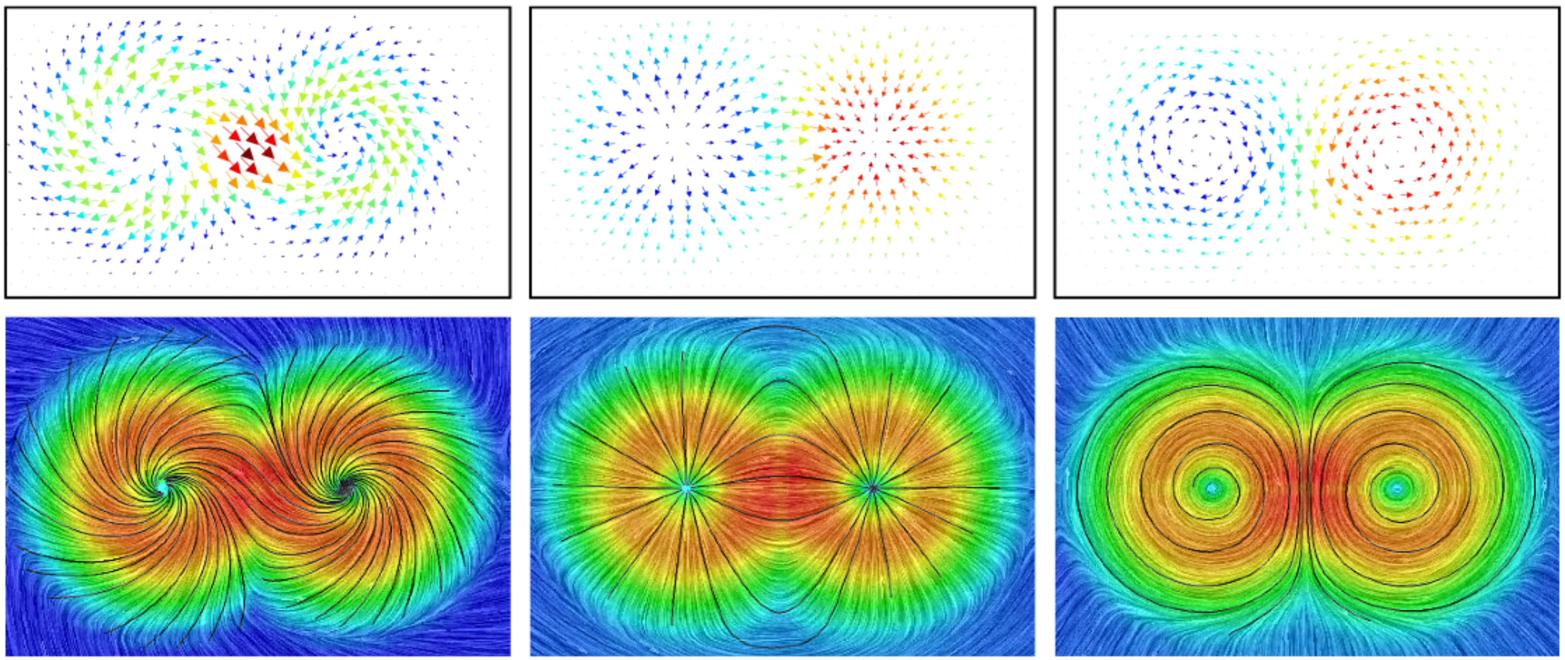
$\mathbf{h} = \mathbf{v} - \mathbf{d} - \mathbf{r}$

For full details, see the paper below

F. Petronetto, A. Paiva, M. Lage, G. Tavares, H. Lopes, H. Lewiner, Meshless Helmholtz-Hodge decomposition, *IEEE TVCG*, 2008

Scalar-based Visualization of Vector Fields

color: vector field magnitude $\|\mathbf{v}\|$ color: divergence $\operatorname{div} \mathbf{d}$ color: vorticity $\|\operatorname{rot} \mathbf{r}\|$



color: vector field magnitude $\|\mathbf{v}\|$ color: magnitude $\|\mathbf{d}\|$ color: magnitude $\|\mathbf{r}\|$

input field $\mathbf{v}$ = curl-free component $\mathbf{d}$ + divergence-free component $\mathbf{r}$

HHD Three Component Decomposition

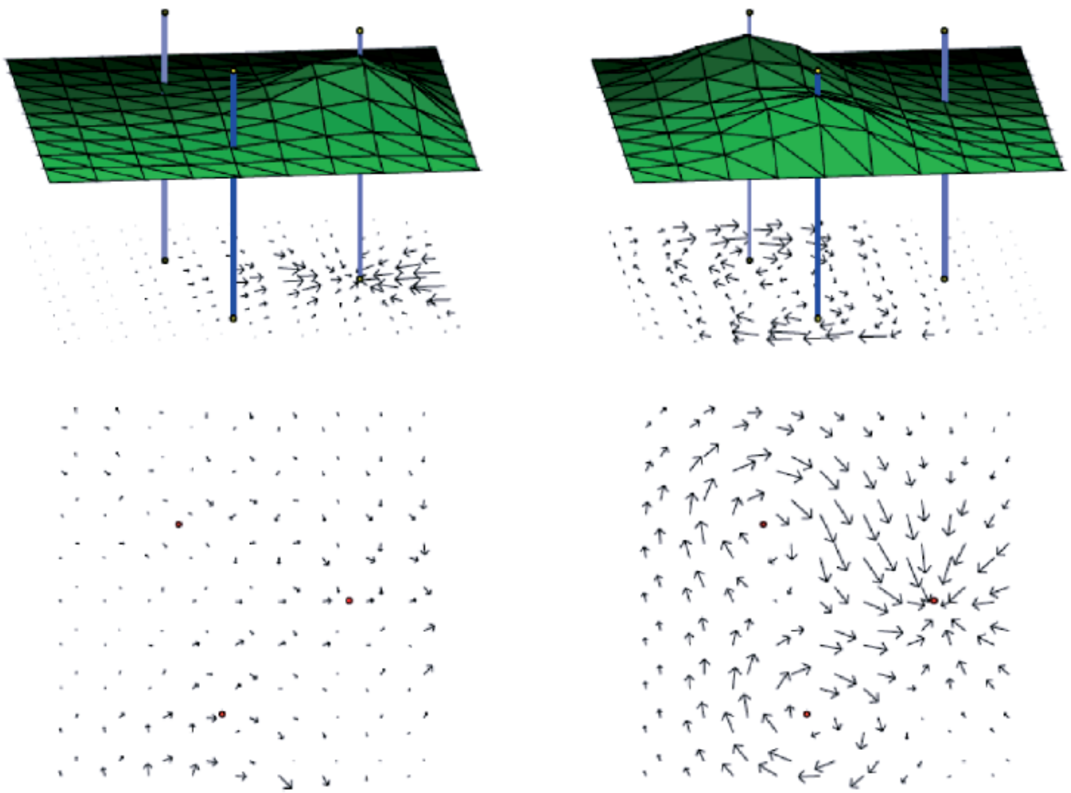


Fig. 4. Results provided by Polthier and Preuß [84]. Top left to bottom right: The irrotational component, $\vec{d}$, with scalar potential D, the incompressible component, $\vec{r}$, with scalar potential R, the harmonic component ($\vec{h}$), and the original vector field ($\vec{\xi}$).

Navier-Stokes Equation

- Momentum equation

$$\mathbf{u}_t = \nu\nabla^2\mathbf{u} - \mathbf{u}\cdot\nabla\mathbf{u} - (1/\rho)\nabla p + \mathbf{g}$$

Change in velocity

Diffusion/ Viscosity

Advection

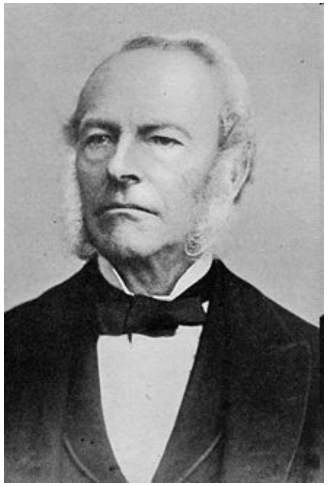
Pressure

Body Forces

- Incompressibility

$$\nabla\cdot\mathbf{u}=0$$

Claude-Louis Navier (1785~1836)



George Gabriel Stokes (1819~1903)

Calculating Derivatives Numerically

- Imagine we are given a closed form for a function
  - For example $f(x,y) = e^{-(x^2+y^2)}$
- Ideally we can then compute derivatives analytically
  - For example $$\frac{\partial f}{\partial x} = -2x(e^{-(x^2+y^2)})$$
- Ideal doesn't always happen
  - Function can be too complicated
  - We might have sampled data instead of a closed form
- In this case, we compute the derivatives numerically

Calculating Derivatives Numerically

- Differentiation is an inherently sensitive problem
  - small changes in input can cause large changes in the output
- Best approach for discretely sampled data :
  - fit a function to the data
  - compute analytical derivatives of the function
- What people often do is compute centered differences

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

Centered Differences

Not really appropriate for sampled data

Can be useful for functions known analytically

Function can be evaluated exactly and at arbitrary resolution

Numerical evaluation may be faster than analytical

h is the step size of a grid or sampling pattern

second-order accurate….error behaves like $O(h^2)$

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

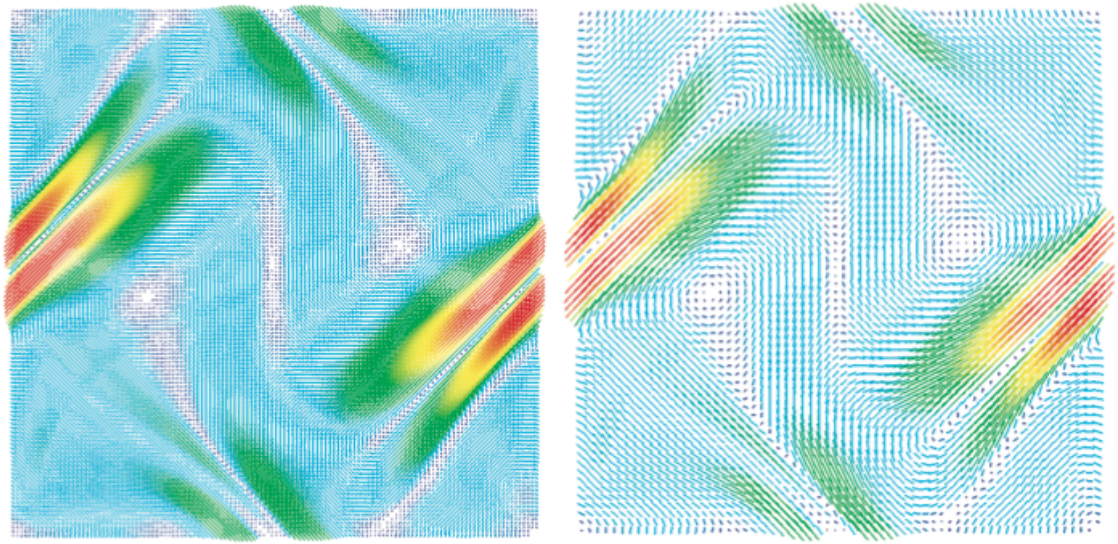
Glyphs

Icons, or signs, for visualizing vector fields

- placed by (sub)sampling the dataset domain
- attributes (scale, color, orientation) map vector data at sample points

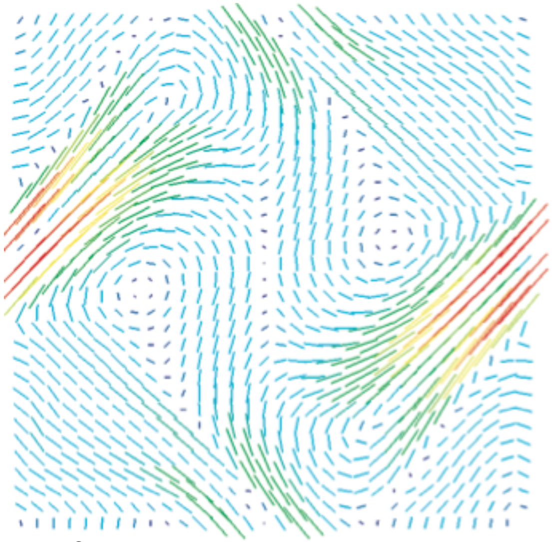
Simplest glyph: Line segment (hedgehog plots)

- for every sample point $x \in D$
  - draw line $(x, x + k\mathbf{v}(x))$
  - optionally color map $\|\mathbf{v}\|$ onto it

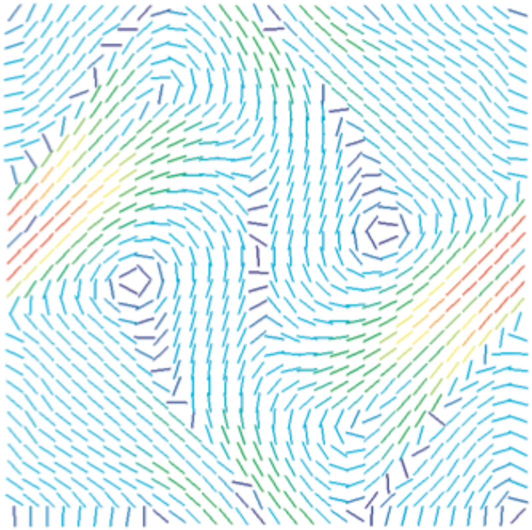


128^2 glyph grid 64^2 glyph grid

Glyphs



32^2 glyph grid



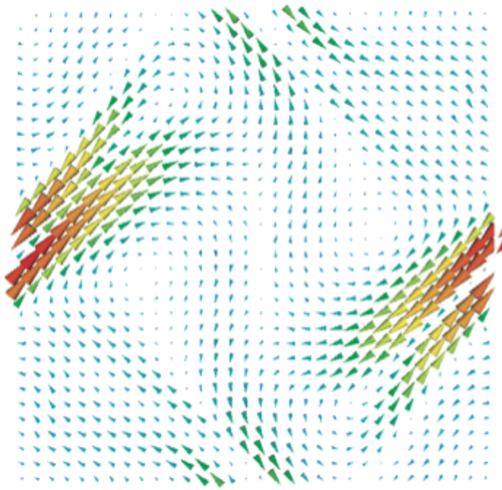
32^2 glyph grid, no line scaling

MHD simulation
256^2 grid

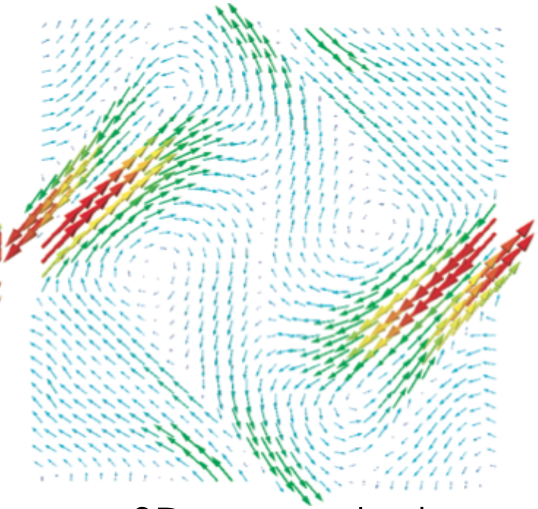
Observations

- trade-offs
  - more samples: more data points depicted, but more potential clutter
  - fewer samples: fewer data points depicted, but higher clarity
  - more line scaling: easier to see high-speed areas, but more clutter
  - less line scaling: less clutter, but harder to perceive directions

Glyphs



3D cone glyphs



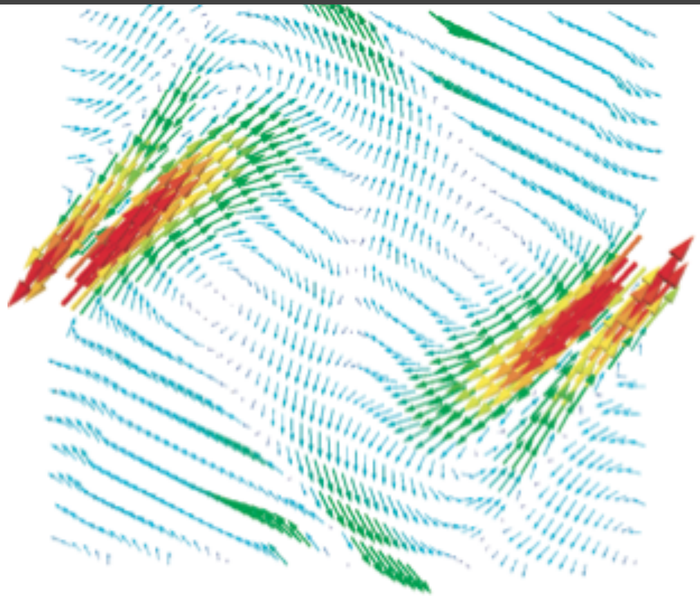
3D arrow glyphs

MHD simulation
256^2 grid

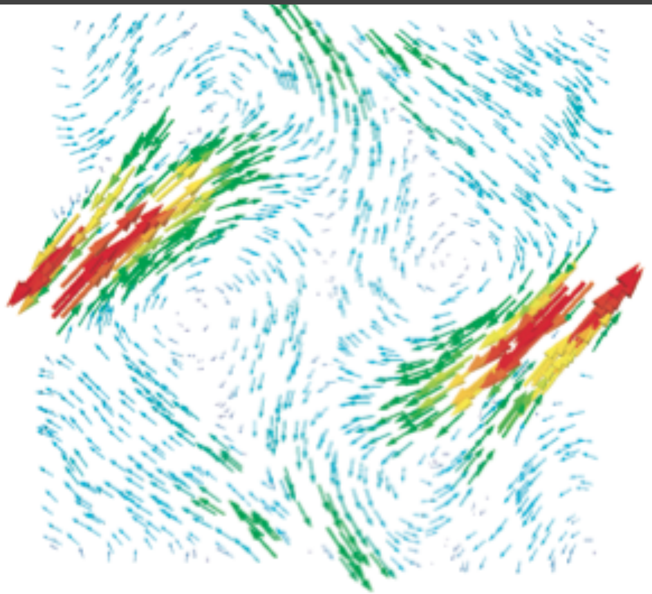
Variants

- cones, arrows, …
  - show *orientation* better than lines
  - but take more space to render
  - shading: good visual cue to separate (overlapping) glyphs

Glyphs



samples on a rotated grid



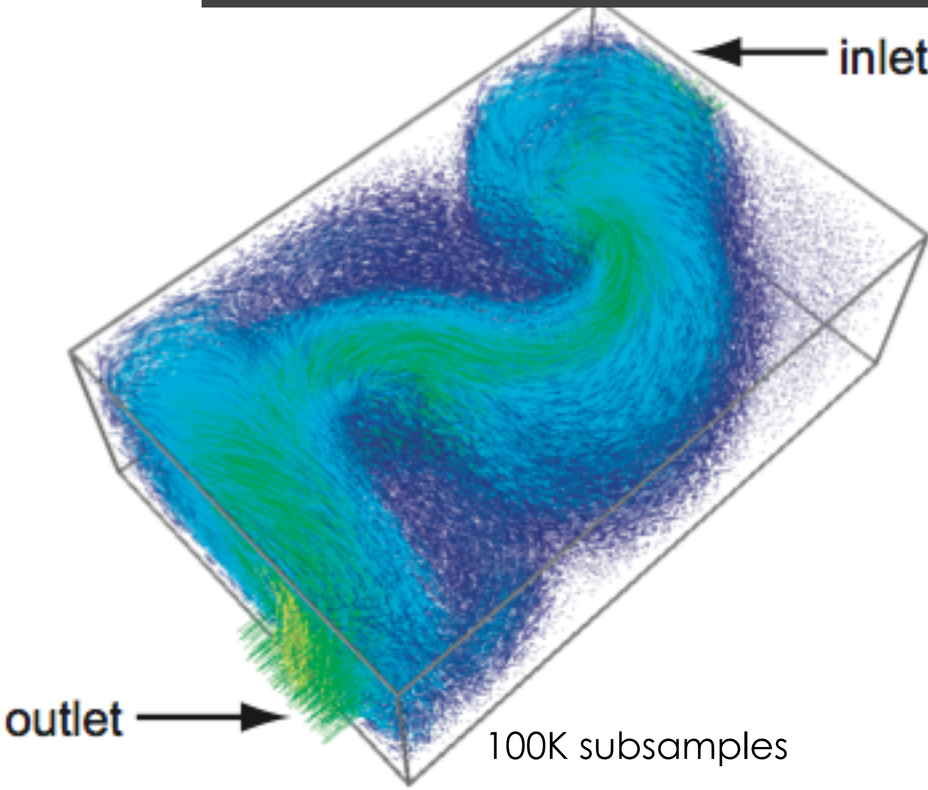
random samples, quasi-uniform density

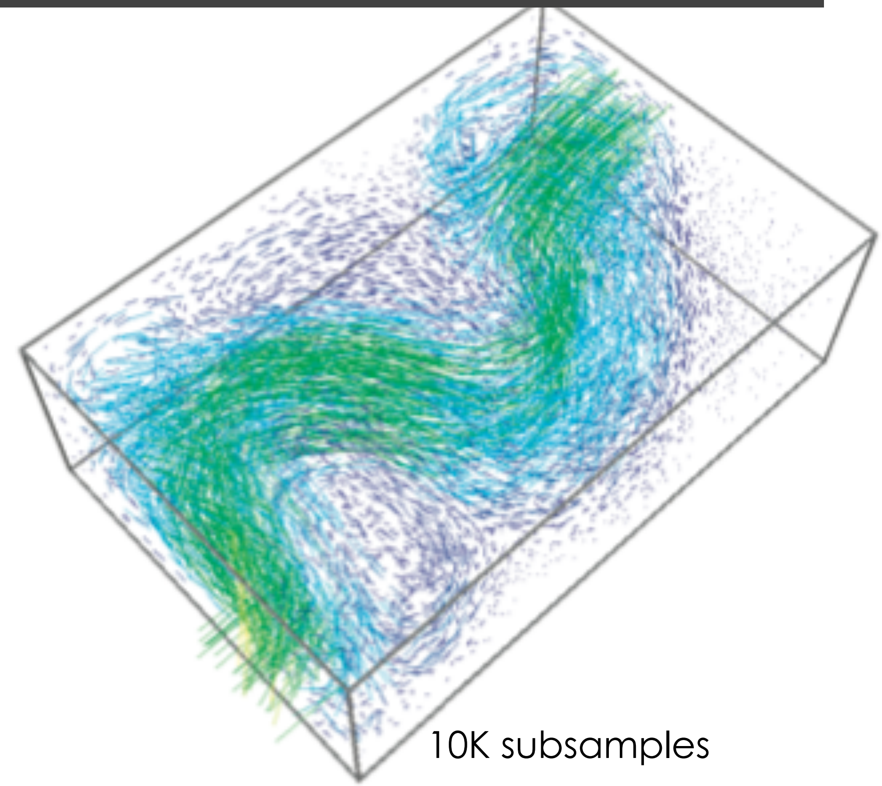
How to choose sample points

- avoid uniform grids!
- random sampling: generally OK

What false impressions does the left plot convey w.r.t. the right plot?

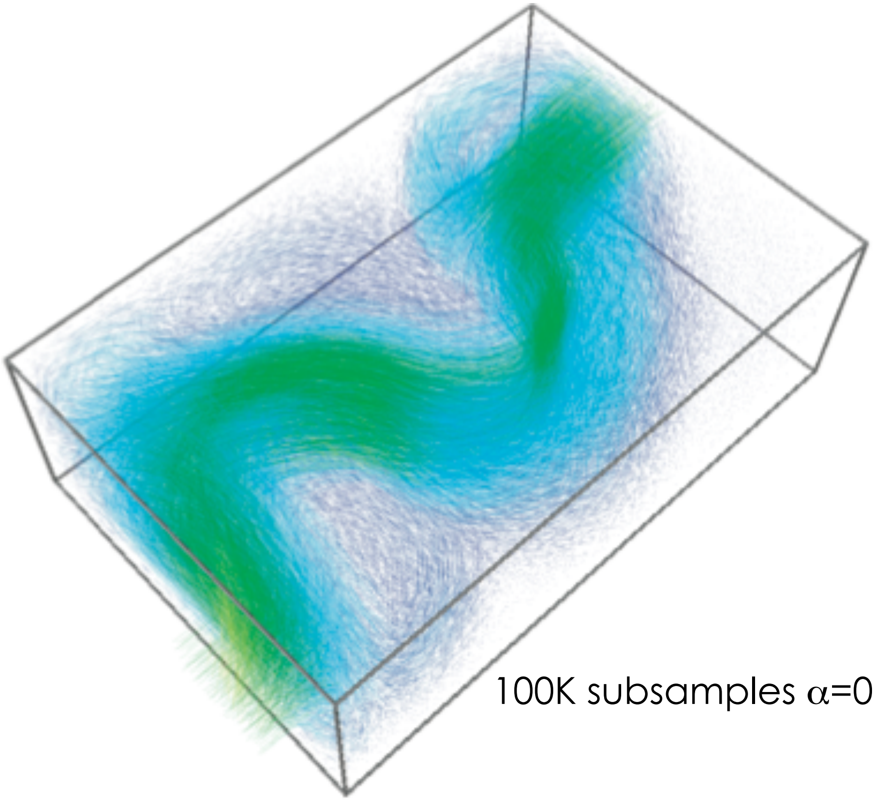
Glyphs in 3D



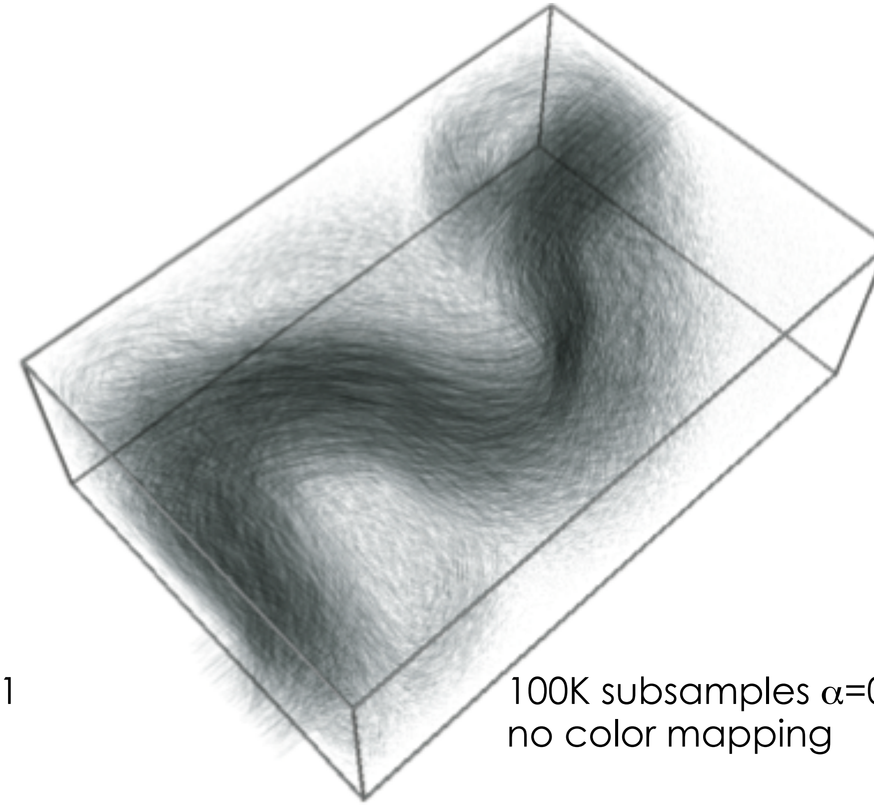


- same idea/technique as 2D vector glyphs
- 3D additional problems
  - more data, same screen space
  - occlusion
  - perspective foreshortening
  - viewpoint selection

Glyphs in 3D



100K subsamples α=0.1

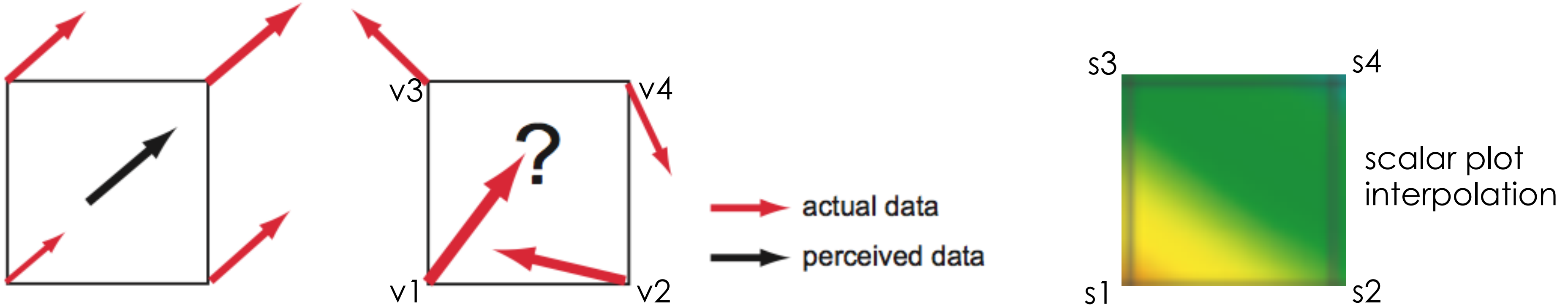


100K subsamples α=0.1
no color mapping

Alpha blending

- extremely simple and powerful tool
- reduce *perceived* occlusion
    - low-speed zones: highly transparent
    - high-speed zones: opaque and highly coherent (why?)

Glyph Problems



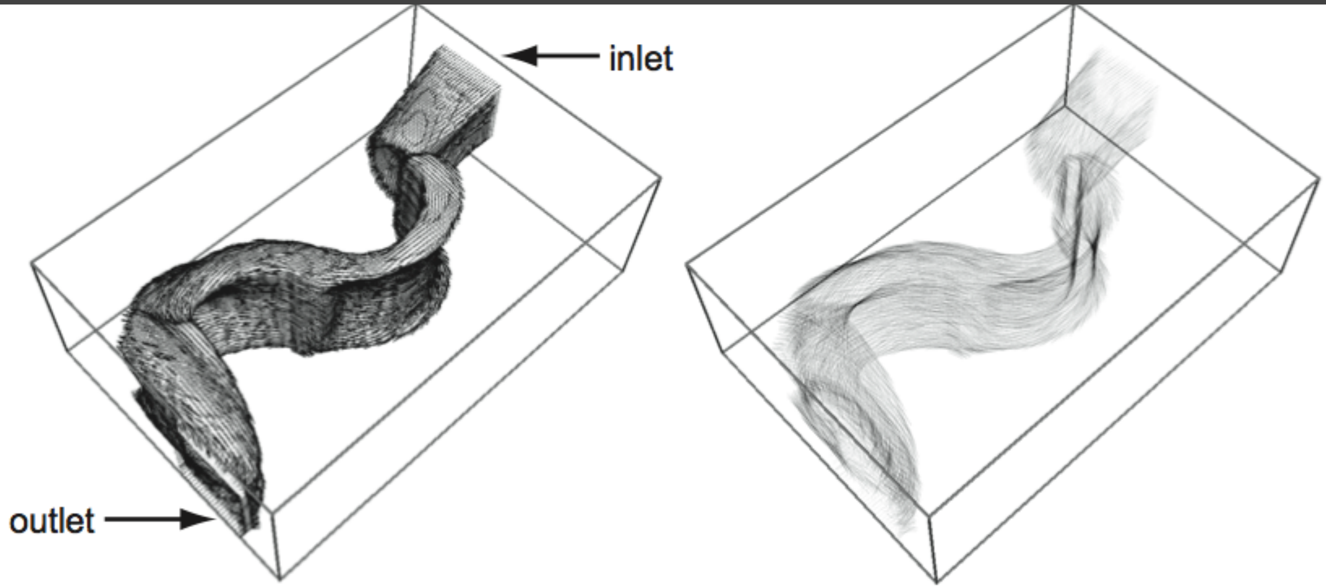
Recall the 'inverse mapping' proposal

- we render something…
- …so we can visually map it to some data/phenomenon

Glyph problems

- **no interpolation** in glyph space (unlike for scalar plots with color mapping!)
- a glyph takes more space than a pixel
- we (humans) aren't good at visually interpolating arrows…
- scalar plots are **dense**; glyph plots are **sparse**
  - this is why glyph positioning (sampling) is extra important

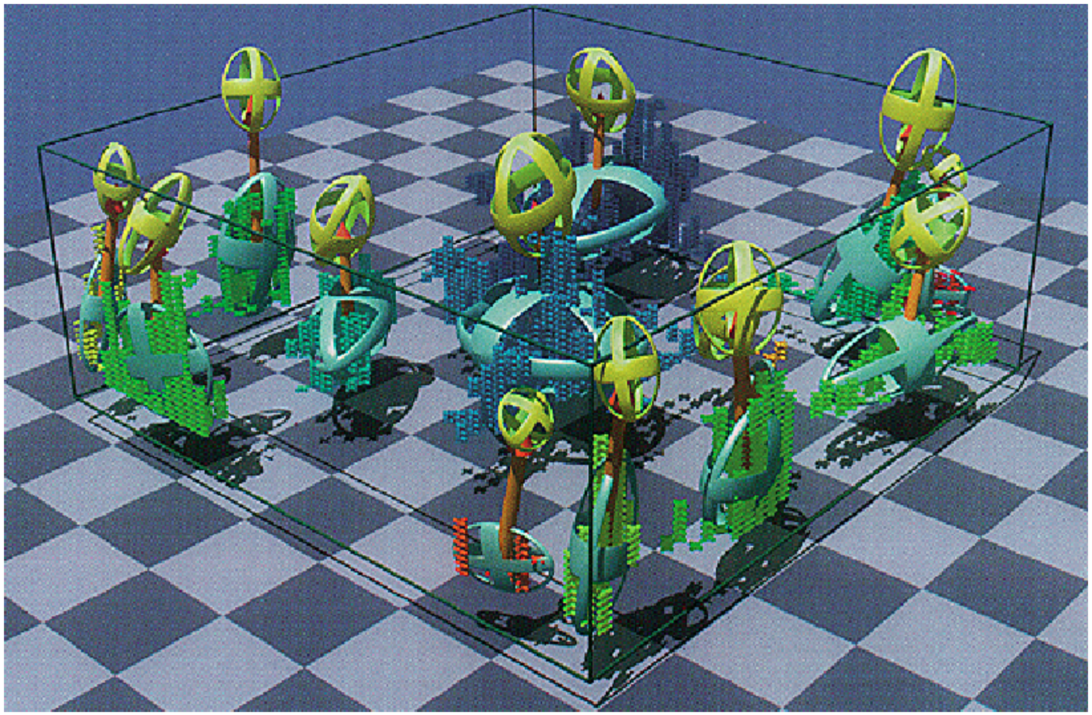
Glyphs on Surfaces



Trade-off between vector glyphs in 2D planes and in full 3D

- find interesting surface
  - e.g. **isosurface** of flow velocity
- plot 3D vector glyphs on it

Glyphs



Average velocity (arrow) and velocity distribution (ellipsoids) for fluid regions with high reaction speed (voxel selection)

- 3*3+3*3+3+1 values per glyph
- nice try, but glyphs are very large → ϖερψ few sample points