Discrete Differential Fluids

An Application of Discrete Differential Modeling to Fluid Mechanics

Mathieu Desbrun

Overview

Fluids, fluids, fluids
- geometric interpretation of classical models
- discrete geometric interpretation
  - new integration technique

Bigger Picture:
Discrete Differential Modeling
- don't discretize your PDEs
- uncover their inherent geometry
- *then* discretize this geometry

Fluid Models (I)

Euler Equations

$$\rho = \text{const} \qquad \frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\nabla p + f$$

$$\nabla \cdot u = 0$$

pressure | momentum eq. | velocity | body forces | mass eq.

- inviscid fluids (not viscous)
- incompressible
- non-linear PDE, with linear constraint

Fluid Models (II)

Navier-Stokes Equations

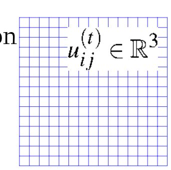
$$\frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\nabla p + f - \nu \Delta u$$

$$\nabla \cdot u = 0$$

- only change: viscosity
  - coefficient ν
- loss of total energy during motion

Algorithm for Simulation

One of many possibilities... (see CFD lit.)
"Stable Fluids" (Stam 99)
- adapted for graphics needs
- regular Eulerian discretization



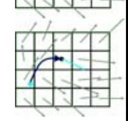
$$u_{ij}^{*} = u_{ij}^{(t)} + \Delta t\, f_{ij}^{(t)} - [\Delta t (u \cdot \nabla) u]_{ij}^{(t)}$$

$$\Delta q_{ij} = \nabla \cdot u_{ij}^{*} \qquad u_{ij}^{(t+\Delta t)} = u_{ij}^{*} - \nabla q_{ij}$$

solve Poisson | not "free of divergence" | velocity | advection

Implementation Issues

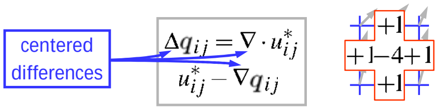
Advection
- discretize? Nah... $[\Delta t (u \cdot \nabla) u]_{ij}^{(t)}$
  - non-linear and nasty
- method of characteristics
  - parcels transported along velocity...
  - let's go backwards in time
    - to know where a "parcel" is *coming from*
    - need to interpolate velocities
    - and resample them
  - unconditional stability! (large time step)



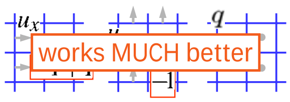
What Where?

Co-located grids

- velocities & pressures at vertices



- staggered grids



The Geometry of Fluids

Euler equations seem clear

- advection + div-free projection ad infinitum
  - numerous follow-up work (Fedkiw *et al.*)
- but what does it mean, geometrically?
  - "total energy" is rather unintuitive
  - is there a notion of momentum preservation?

Yes

- but of course, we need to massage the PDE
- so as to reveal the geometric structure

Geometry Revealed

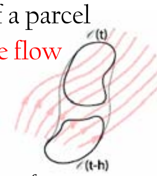
$\frac{\partial u}{\partial t} + (u \cdot \nabla)u = -\nabla p$

Pressure disappears when we take the curl:

$$\frac{\partial \omega}{\partial t} + \mathcal{L}_u \omega = 0 \qquad \omega = \nabla \times u \ \text{(vorticity)}$$
$$div(u) = 0 \qquad u \parallel \partial \mathcal{D}$$

- vorticity measures the "spin" of a parcel
- vorticity is "advected" along the flow
- the circulation around *any closed loop* is constant as it gets advected
  - known as *Kelvin's theorem*
  - call it preserv. of angular momentum if you want

$\Gamma(t) = \oint_{C(t)} u.dl$



Geometry Revealed

So we know:

Integral of vorticity constant on an advected sheet

Additionally, **ω** *defines u*

- if we ignore complex topology for a moment
- *u* = ∇x (Δ^{-1}**ω**) because *u* is divergence free!

Vorticity is the only real variable here and Kelvin's is ***a defining property***

(Navier-Stokes: loss along the way)

Towards a Proper Discretization

Domain discretization = *simplicial complex*

- fluxes through faces for velocity
  - intrinsic (coordinate-free) and **eulerian**
    - » reminiscent of staggered grids…
- net flux for divergence
  - what comes in...must come out
- flux spin for vorticity
  - Torque created on a "paddle wheel"
- valid for any grid...

Enter Discrete Exterior Calculus

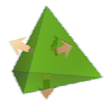
Need for proper link btw flux, vorticity, div

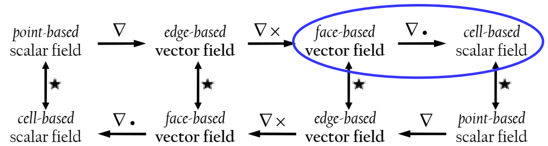
- hopefully matching differential counterparts
- to create a discrete differential structure
  - i.e., structure-preserving discretization
- Fortunately, that's DEC
  - we know how to do all that, right??
  - flux = 2-form
  - div = exterior derivative of flux
  - curl = ★d★ of flux

Divergence Operator

Simply **d** of 2-form

- summing face values of tets
- returning values in tets

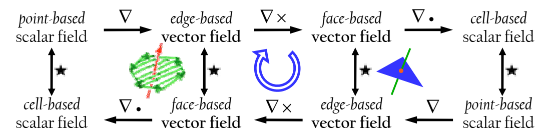




Curl Operator

Curl requires going to the dual

- from faces to dual edges first
- then **d** (sum of dual edge values)
- then back onto primal edges



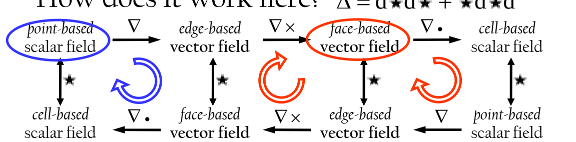
Laplacian Operator

Wait, sum of second partial derivatives… on a PWL field??

- sure, we do it all the time
  - link to FE/weak formulation

Try it for 0-forms at home: you'll get the cot formula…

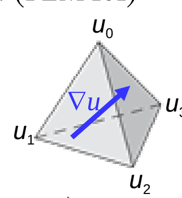
How does it work here? $\Delta = \mathrm{d}\star\mathrm{d}\star + \star\mathrm{d}\star\mathrm{d}$



Gradient Operator (for completeness)

Wait, constant per tet, right? (FEM 101)

- yes
- but can be stored as values on edges, as announced



$$u = \sum_i u_i\, \varphi_i$$

$$\Rightarrow \mathrm{d}u = \sum_i u_i\, \mathrm{d}\varphi_i = \sum_{e_k=\{i,j\}} (u_i - u_j)\, \varphi_{e_k}$$

Integrating Equations of Motion

We have all the computational set-up

- But how do we integrate the motion?

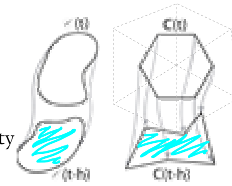
Through preserving important structures?

- Circulation/vorticity preservation
- Crucial for visual impact
  - volutes in smoke
  - vortices in liquids

Discrete Kelvin's Theorem

Simple way to integrate Euler equations:

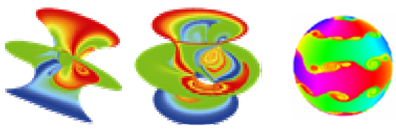
- For each 1-simplex
  - backtrack local loop in current velocity field
  - deduce new circulation
    - i.e., new discrete vorticity
- Find new velocity field
  - simple Poisson equation
    - $u = \nabla\times(\Delta^{-1}\omega)$

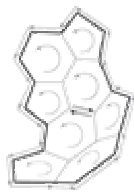


Discrete Kelvin's Theorem

Guarantees circulation preservation for any discrete loop

- big loop = union of small ones
- ... even on curved spaces



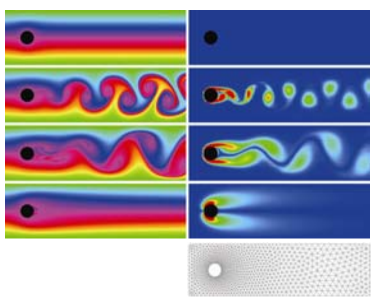


Results

New method

- exact discrete vorticity preservation
- arbitrary simplicial meshes
  - see also paper by Feldman *et al.* this year
- **everything is intrinsic**
- basic operators very simple (super parse)
- great flows for small meshes!
  - computationally efficient even on coarse mesh
  - no need for millions of vortex particles

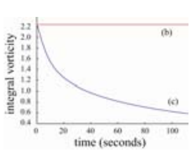
Channel

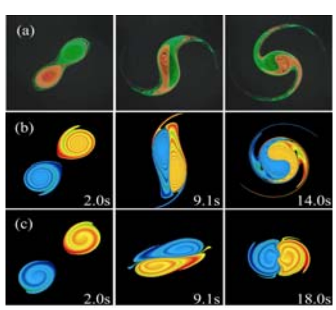


Smoking Bunny

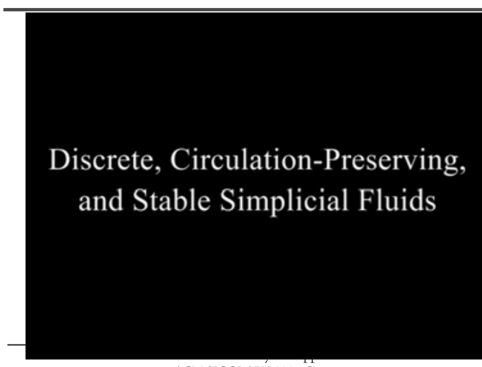
7k vertices, 32k tets; 0.45s per frame on PIV (3GHz)

Merging Vortices





Movie



Take-Home Message

Don't Discretize your PDE

- discretize its geometric structures
  - PDEs often hide these structures
- uncover the nature of the variables involved
  - usually, natural locations on mesh
- turn the crank....

Next

- circles may be the right discrete geometry!
  - conformal geometry