Exploiting the "unreasonable effectiveness" Of Geometry in Computing

Mathieu Desbrun

GeomeriX

Dutch-Flemish Scientific Computing Society – 49th Woudschoten Conference - September 2025

1

The Role of Geometry

Geometry is key in m fields
- at the cros
  - obser world around us
- "moth es
  - fro al structures and syn
- studie
  - Cart m, Noether…
- mostly d gh
  - based culus

Large body o geometry
… alas, discrete in substance

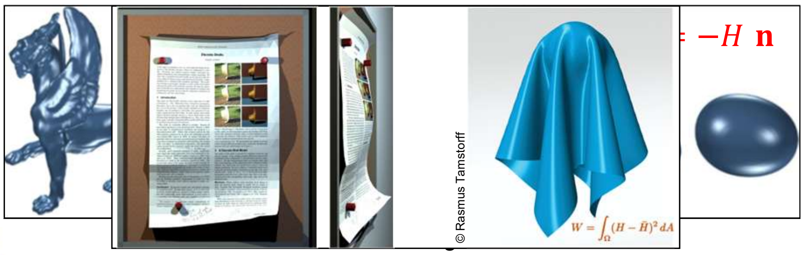
2

2

Continuum vs. Finitude

"Discrete" differential geometry

- *finite-dimensional counterpart* to continuous theory
  - where we *leverage* differential understanding
- geometry as a guiding principle to **discretization**
  - discretize the geometric principles
    - predictive power guaranteed, , even with low-order basis fcts
  - NOT THE PDES DIRECTLY !!



The heresy of piecewise linear functions for shells

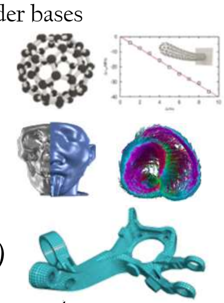
Continuum vs. Finitude

"Discrete" differential geometry

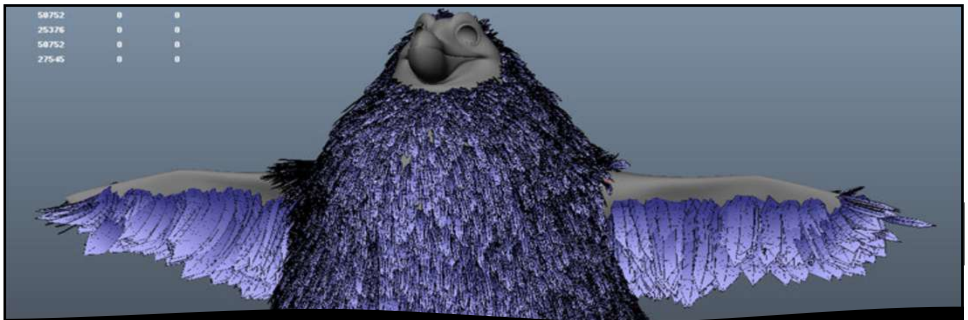
- *finite-dimensional counterpart* to continuous theory
  - where we *leverage* differential understanding
- geometry as a guiding principle to **discretization**
  - discretize the geometric principles
    - predictive power guaranteed, , even with low-order bases
  - NOT THE PDES DIRECTLY !!
    - PDEs often hide structures completely

Of both academic and practical interests

- education *(simple discrete analogs)*
- Hollywood *(cool graphics, fast animation)*
- computational science *(new numerical methods)*



Next, four vignettes to illustrate a few aspects...



Vignette 1:
Grooming

5

5

How Do You Grow Hair?

© Pixar/Disney

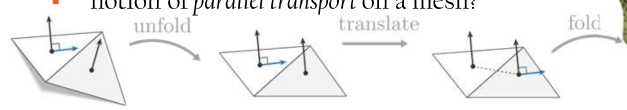
6

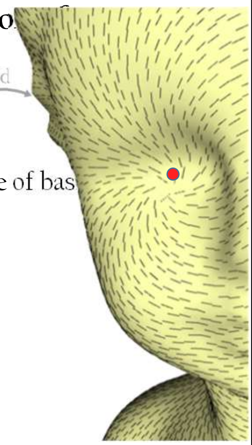
6

Vector Field Processing

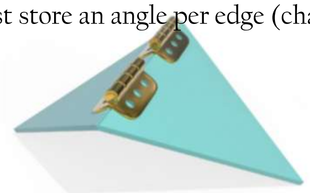
How to design tangent {vector|direction|frame} fields?

- need to control smoothness, and singularities…
- geometry to the rescue: use of **connection o**
  - notion of *parallel transport* on a mesh?





  - code for it? just store an angle per edge (change of bas



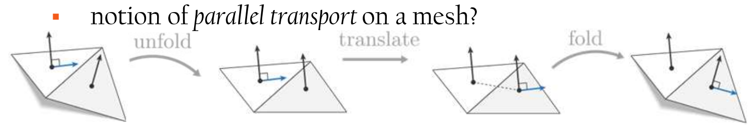
7

7

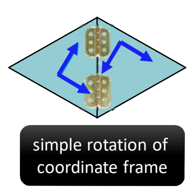
Vector Field Processing

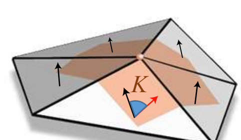
How to design tangent {vector|direction|frame} fields?

- need to control smoothness, and singularities…
- geometry to the rescue: use of **connection one-forms**
  - notion of *parallel transport* on a mesh?



  - code for it? just store an angle per edge (change of basis)
  - discrete Levi-Civita (metric) connection and its discrete holonomy





$$K = 2\pi - \sum_a \gamma_a$$

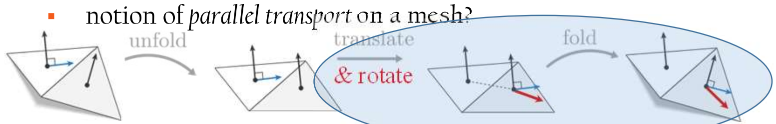
8

8

Vector Field Processing

How to design tangent {vector|direction|frame} fields?

- need to control smoothness, and singularities…
- geometry to the rescue: use of **connection one-forms**
  - notion of *parallel transport* on a mesh?



  - code for it? just store an angle per edge (change of basis)
  - discrete Levi-Civita (metric) connection and its discrete holonomy
- extension to an arbitrary principal connection?
  - add **adjustment rotation** during the translation…
  - *integrated* connection 1-form
    - see discrete 1-forms in Discrete Exterior Calculus

9

9

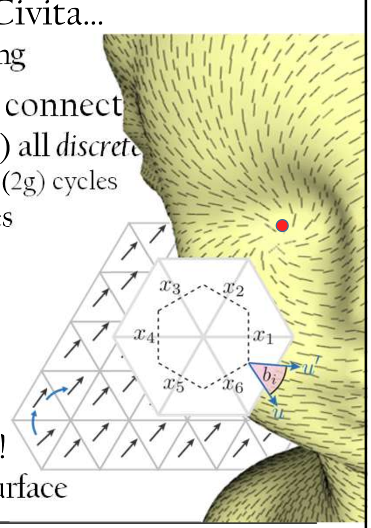
Discrete Trivial Connection

We can find an *adjustment* to Levi-Civita…

- one rotation angle per edge crossing

to cancel holonomy of Levi-Civita connect

- forcing zero holonomy on (almost) all *discret*
  - contractible (V) & noncontractible (2g) cycles
- *except* for a few chosen singularities
  - Poincaré-Hopf theorem
- and get smallest adjustments!
  - L^2-minimum of adjustment vector for "straightest" solution
    - link to torsion [Braune et al. 2025]



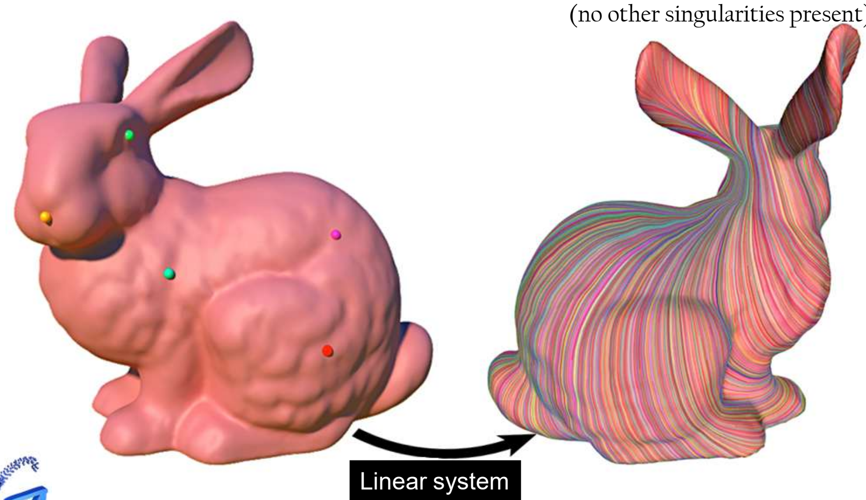
Now, path-independent transport!

- creating discrete vector field on surface

10

Growing Hair on a Bunny...

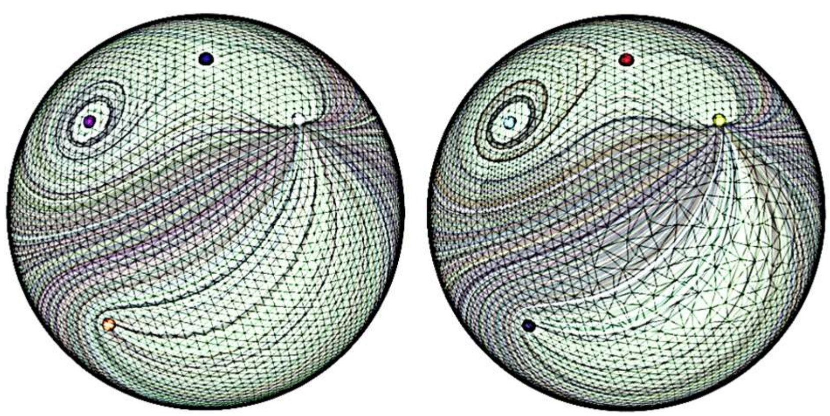
Resulting trivial connection
(no other singularities present)



11

11

Robustness to Meshing Too!

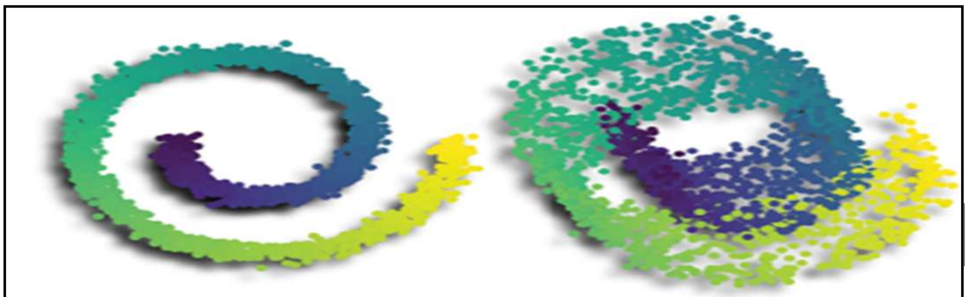


12

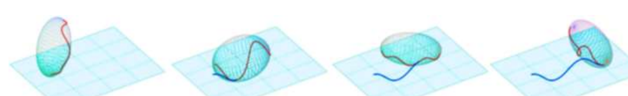
Recent Use

13

13



Vignette 2:
Cartan for Data Science



© Raffaelli et al. 2016

14

14

Context

Era of big data

- lots of high dimensional datasets available
  - think of them as pointsets in high dimension ($\mathbb{R}^D$)
- need for data analysis tools... can geometry help!?
- data often sample a d-manifold with $d \ll D$
  - example: just a bunch of 128x128 images? (here, D=128^2)
    - only camera angle varies (two dimensional, in disguise!)

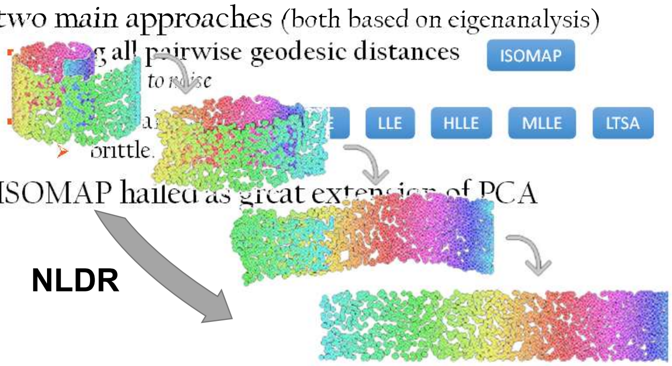
15

Geometry for Data Science?

Dimensionality Reduction:
mapping data from $\mathbb{R}^D$ to $\mathbb{R}^d$, with $d \ll D$

e.g., sum of squared differences of pixel intensitiess

- i.e., finding a *Euclidean embedding* in low dimension
  - in a "most isometric" way (e.g., *try to preserve pairwise distances*)
- two main approaches (both based on eigenanalysis)
  - ...g all pairwise geodesic distances ISOMAP
    - *to n...se*
  - ...a... LLE HLLE MLLE LTSA
    - ...brittle.
- ISOMAP hailed as great extension of PCA

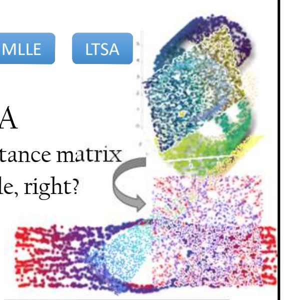


16

Geometry for Data Science?

Dimensionality Reduction:
mapping data from $\mathbb{R}^D$ to $\mathbb{R}^d$, with $d \ll D$

- i.e., finding a *Euclidean embedding* in low dimension
  - in a "most isometric" way (e.g., *try to preserve pairwise distances*)
- two main approaches (both based on eigenanalysis)
  - **using all pairwise geodesic distances** ISOMAP
    - *robust to noise*
  - using local positioning LE LLE HLLE MLLE LTSA
    - brittle...
- ISOMAP hailed as great extension of PCA
  - Dijskstra for geo distances, then MDS of distance matrix
  - but planar pointsets *should* be trivial to handle, right?
  - pointsets on a developable surfaces too
  - should be robust to holes too, right?

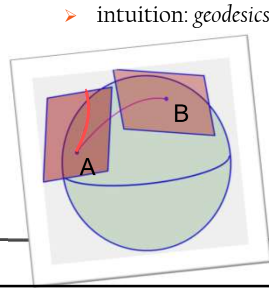


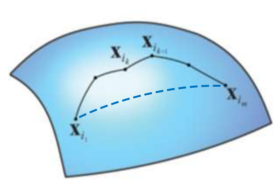
17

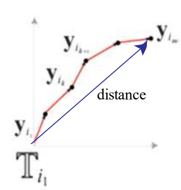
Connection-based ISOMAP

Key idea: Parallel transport to find geodesic distances

- use *intrinsic neighborhoods* (k-NN) to estimate tangent spaces
- define *metric connection* between tangent spaces $\mathbb{T}_i$
  - rotation of a tangent space frame to get to a parallel one nearby
    - $\mathbf{R}_{ij} = \underset{\mathbf{R}\in O(d)}{\operatorname{argmin}} \|\mathbb{T}_j - \mathbb{T}_i\mathbf{R}\|_F^2$, Procrustes problem solved via SVD
- geodesic distances easy to evaluate (instead of Dijkstra's)
  - through Cartan's development (unfold path in tangent space)
    - intuition: *geodesics are straight under development*







18

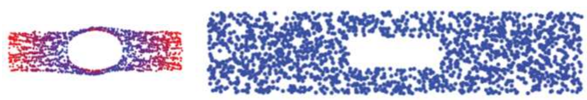
Connection-based ISOMAP

Key idea: Parallel transport to find geodesic distances

- use *intrinsic neighborhoods* (k-NN) to estimate tangent spaces
- define *metric connection* between tangent spaces $\mathbb{T}_i$
  - rotation of a tangent space frame to get to a parallel one nearby
    - $\mathbf{R}_{ij} = \underset{\mathbf{R} \in \mathcal{O}(d)}{\operatorname{argmin}} \left\| \mathbb{T}_j - \mathbb{T}_i \mathbf{R} \right\|_F^2$
- geodesic distances easy to evaluate (instead of Dijkstra's)
  - through Cartan's development (unfold path in tangent space)
    - intuition: *geodesics are straight under development*

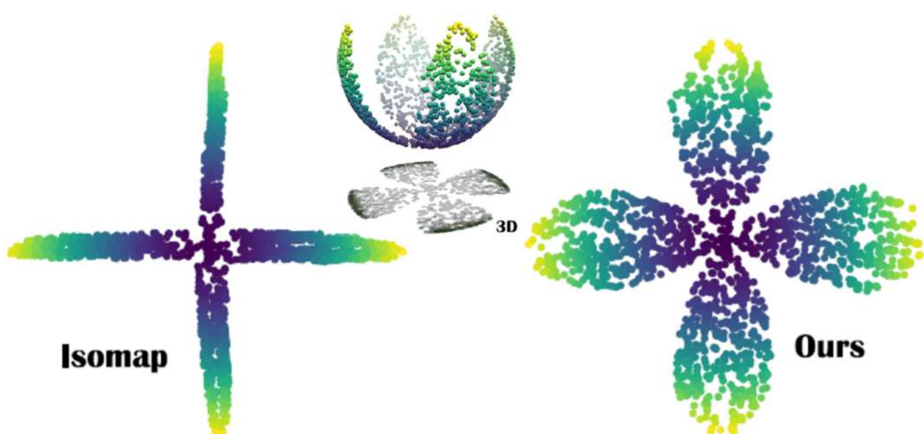
6 lines of code to change in ISOMAP....

- https://tinyurl.com/PTUcode



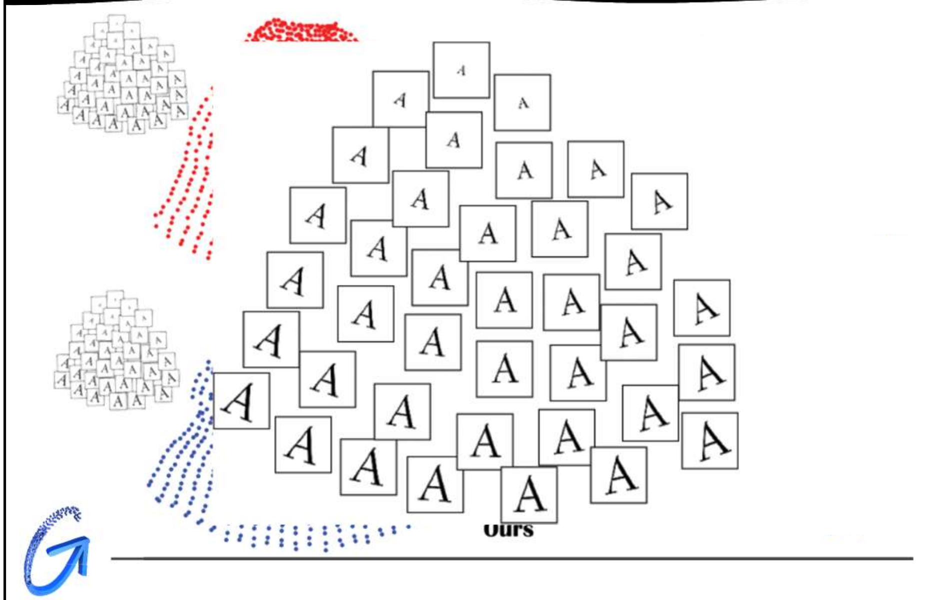
19

Result on 3D Embedded in 100D



20

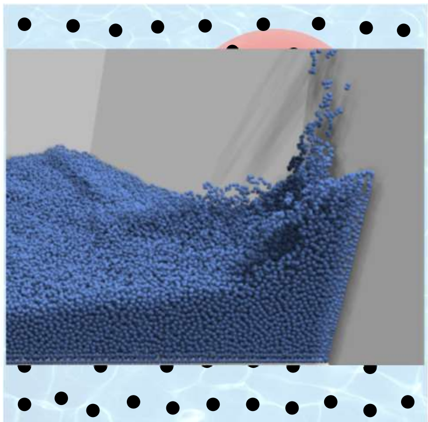
On «Real» Data (letter A rotated/scaled)



21

22

Revisiting Fluid Incompressibilty



FIELD APPROXIMATION

kernel methods mostly (SPH)

evaluate density, forces

brittle

INCOMPRESSIBILITY

divergence-free velocity field

inaccurate

LIMITATIONS

clustering of particles

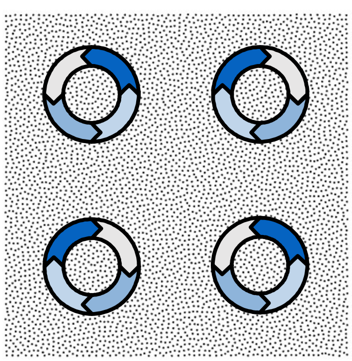
requires tiny time-steps

artificial damping

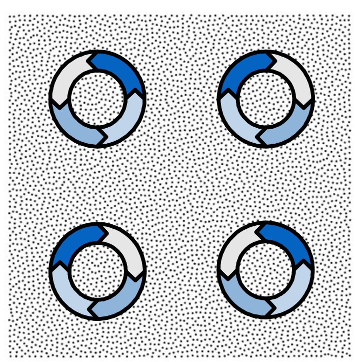
Examples of Previous Work

Typical results, even for small timesteps

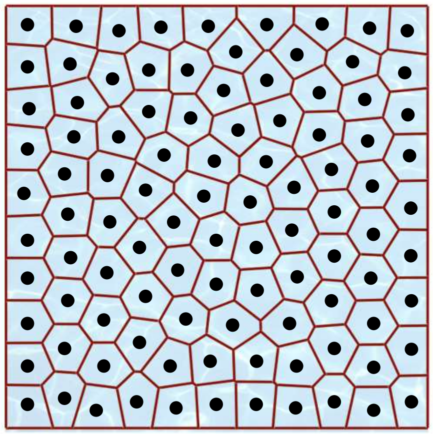
Divergence-free velocity (PIC-FLIP)



Volume preservation (PBF,PCSPH)



Geometric Approach



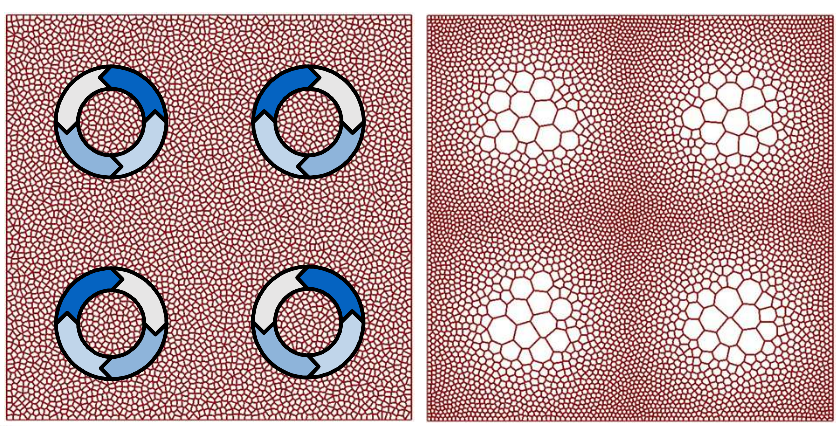
INCOMPRESSIBILITY
divergence-free velocity field
= preservation of local volumes

PARTICLES + POWER CELLS
= POWER PARTICLES
preservation of finite volumes
well-centered **power diagram**
no kernel evaluation needed

25

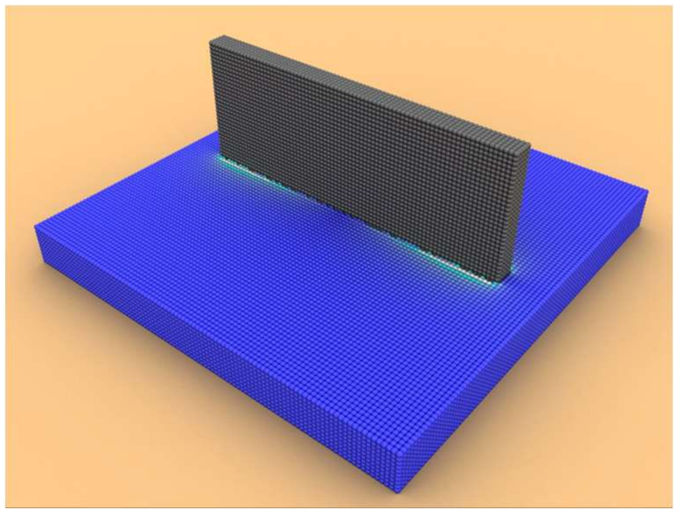
Geometric Fluid

Can handle any initial set of particles



26

Rotating Blade



27

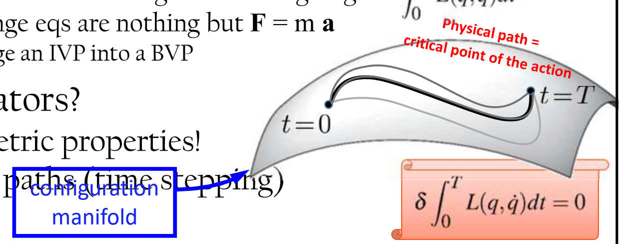
Variational Nature of Mechanics

The basic structure of mechanics is geometric

- Hamilton's stationary action principle and variants
  - motion extremizes the integral of the Lagrangian $\int_0^T L(q,\dot{q})dt$
  - Euler-Lagrange eqs are nothing but **F** = m **a**
    - but change an IVP into a BVP

Numerical integrators?

- leverage geometric properties!
- just discretize paths (time stepping)



$q(0)$ $q(T)$

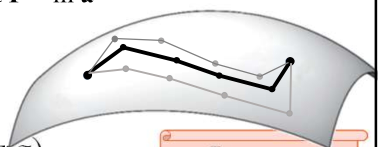
28

28

Variational Nature of Mechanics

The basic structure of mechanics is geometric

- Hamilton's stationary action principle and variants
  - motion extremizes the integral of the Lagrangian $\int_0^T L(q,\dot{q})dt$
  - Euler-Lagrange eqs are nothing but **F** = m **a**
    - but change an IVP into a BVP



$$\delta \int_0^T L(q,\dot{q})dt = 0$$

Numerical integrators?

- leverage geometric properties!
- just discretize paths (time stepping)
  - and use quadrature to evaluate discrete action

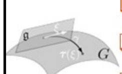
solve DEL (time integration) $L_d(q_k, q_{k+1}) = \int_{t_k}^{t_{k+1}} L(q,\dot{q})dt$ $(q_{k-1}, q_k) \rightarrow q_{k+1}$

- make for better numerics
  - preserves symplecticity
  - conserves energy remarkably well [Hairer]
  - preserves symmetries through (discrete) Noether's theorem

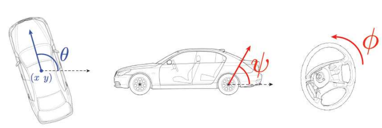
29

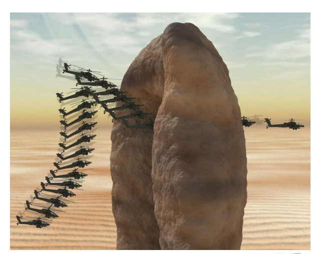
29

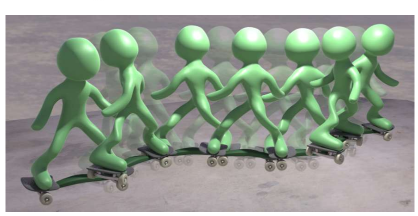
Very Successful Developments

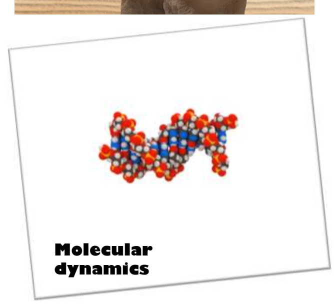


- Links to known integrators
- Lie group integrators
- (non)holonomic constraints,
- time adaption
- etc...









30

30

Fluids in Computer Graphics

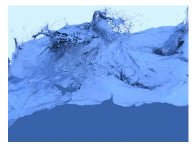
Often using incompressible Navier-Stokes equations

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\frac{1}{\rho}\nabla p + \nu\,\nabla\cdot\nabla\boldsymbol{u} + \boldsymbol{F} \quad \text{with} \quad \nabla\cdot\boldsymbol{u} = 0$$

- can be expressed in various ways — in particular, with vorticity

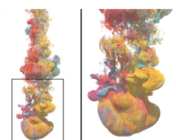
Solvers:

- Eulerian methods
  - grid- or mesh-based
- Lagrangian methods
  - particles, typically
- hybrid methods
  - e.g., MPM, PIC,...
  - geometric integrators too!
    - Arnold's geodesics of the *volume-preserving diffeomorphisms*

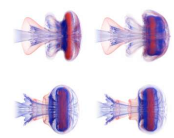


Tetrahedral Meshes [Ando et al. SIG '13]

Vorticity-based [Zhang et al. SIG '15]



PolyPIC [Fu et al. SIG '17]



Reflection-advection solver [Zehnder et al. SIG '18]

31

31

Practical Issues

Particles cool, but messy

- follow the motion, so seemingly efficient
  - but local density keeps on changing, so noisy numerics
- air is everywhere, so millions of particles needed!
  - adaptive sampling tricky and not memory friendly

Meshes/grids great, but limited

- fixed resolution, whether there's action or not
  - adaptive grid size costly in practice
- often require smaller time steps
  - for similar visual quality

Issues with even hybrid methods:

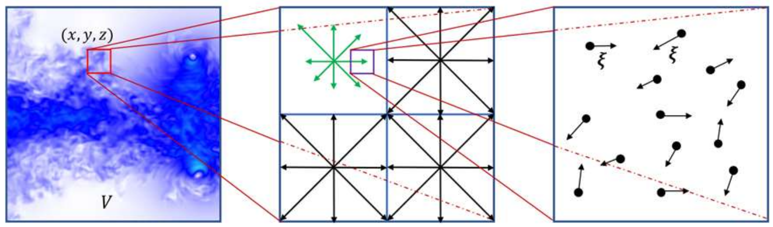
- always need to somehow maintain divergence-freeness
- lack of accuracy in nonlinear advection (dissipation, dispersion...)

32

32

Boltzmann Discretization

Introducing a mesoscopic description of fluid



Macroscopic view	**Mesoscopic view**	**Microscopic view**
Navier-Stokes equations	Boltzmann equation	Molecular dynamics

Boltzmann Discretization

Introducing a mesoscopic description of fluid

- based on a statistical-mechanics (a.k.a. kinetic) model
  - use a **particle distribution function** $f(\boldsymbol{x}, \boldsymbol{v}, t)$
    - probability for a particle to be at $\boldsymbol{x}$ at time t with a velocity $\boldsymbol{v}$
- Boltzmann transport equation:

$$\frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \nabla f = \Omega(f) + \boldsymbol{F} \cdot \nabla_{\boldsymbol{v}} f$$

streaming (particles going straight) — collision (particles collide) — forcing (particles pushed around)

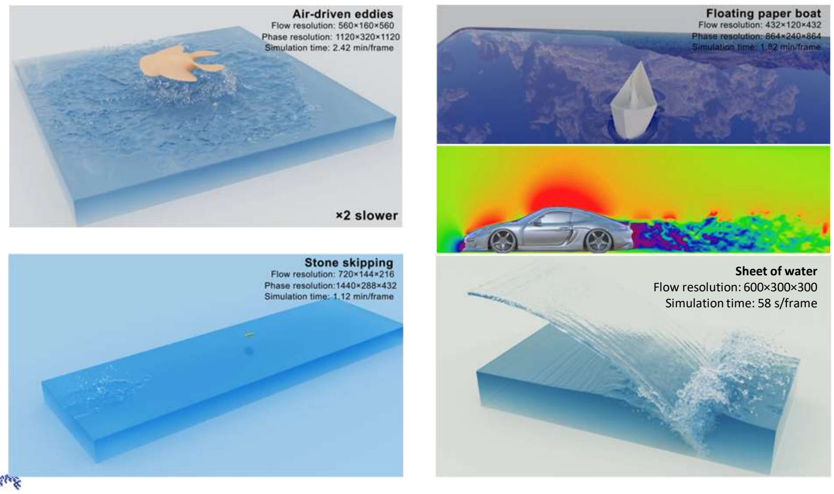
  - amounts to (near) incompressible Navier-Stokes
  - once disc[...]ttice [...]m[...]xes issues, and importantly, can be done in a massively-parallel way!

Macroscopic quantities simple to recover!

$$\rho(\boldsymbol{x}, t) = \int f d\boldsymbol{v} \qquad \rho\boldsymbol{u}(\boldsymbol{x}, t) = \int \boldsymbol{v} f d\boldsymbol{v}$$

note: "$\boldsymbol{u}$=0" ≠ no motion!

Great for Graphics...



35

And for Usual Obstacle Course

But still no "variational" approach for it...

36

36

Vignette 4:
Coarse-graining & Fast Solvers

37

37

What's Up with Coarse Meshes?

Take a simple hanging cube…

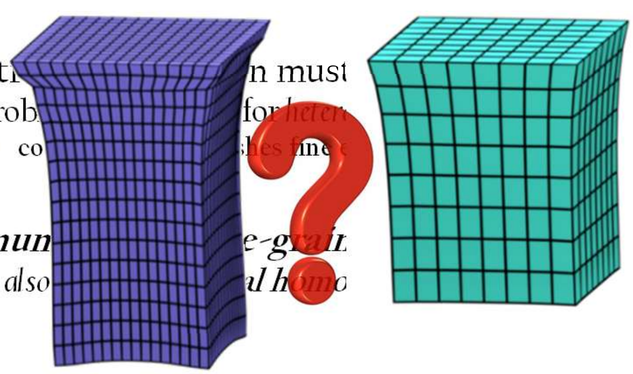
- even with the exact same simulation technique…
  - say, non-linear deformable body simulated with trilinear FEM
- simply choosing a twice-coarser grid changes *everything*

Geometr… n must … hysics!

- prob… for *heter*…
  - co… hes fine … neities

Enter ***num*** … ***e-grai***…

- *also* … *al homo*…



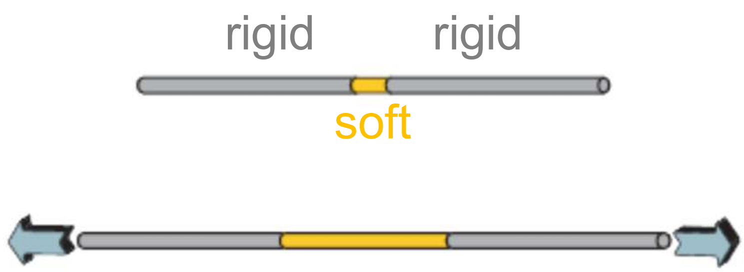
38

38

Aggravating Circumstances

Averaging elasticity/stiffness is just not that easy…

- hard + soft in 1D?
- does not behave like a half-hard single bar



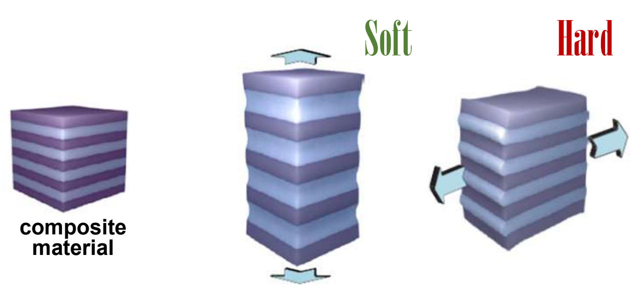
39

39

Aggravating Circumstances

Complexity emerges from simplicity

- in elasticity, *isotropic materials* are defined by 2 coefficients…
  - Poisson's ratio and Young's modulus
- *anisotropic linear elasticity* requires 21 parameters
  - the whole elasticity tensor, with symmetries removed
- but two isotropic materials create … an **anisotropic** material



40

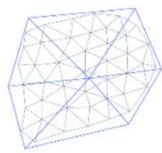
Numerical Homogenization

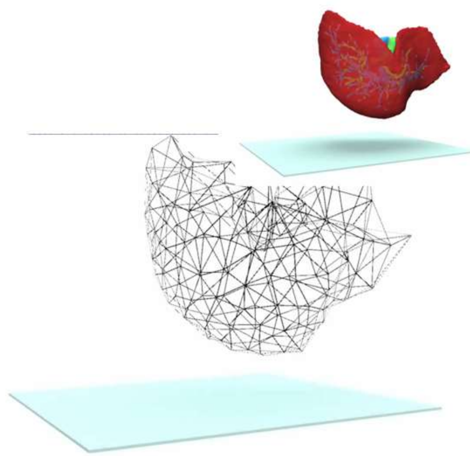
Mostly two approaches to improve results

Richer coarse models

- **Idea**: compute homogenized material per coarse element
  - making sure elastic potentials match

[Kharevych 2009, Panetta 2015,...]

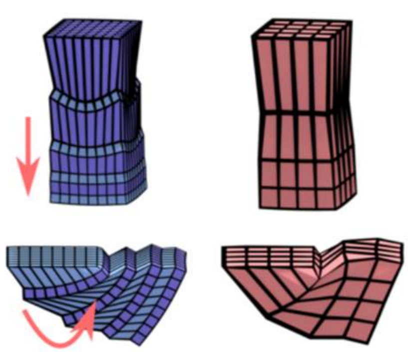




- Limitations: only two levels; imperfect

Numerical Homogenization

Mostly two approaches to improve results



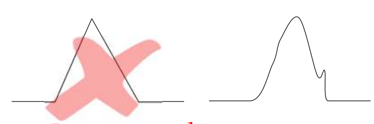
[Chen 2018]

imperfect

Adapted basis functions

- **Idea**: change shape functions to offer a richer solution space
  - precomputed locally or globally

[Nesme 2009, Chen 2018,...]



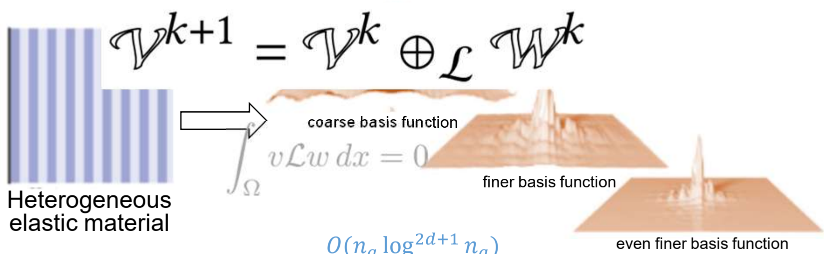
- Limitation: slow preprocess, imperfect

Better Still: Operator Adaptation

Finding refinable operator-adapted basis functions

- adaptivity through operator- and material-adapted wavelets
  - **block-diagonalizes** the stiffness matrix
  - basis fcts localized in **both space and eigenspace**
  - **tight bounds** on accuracy!

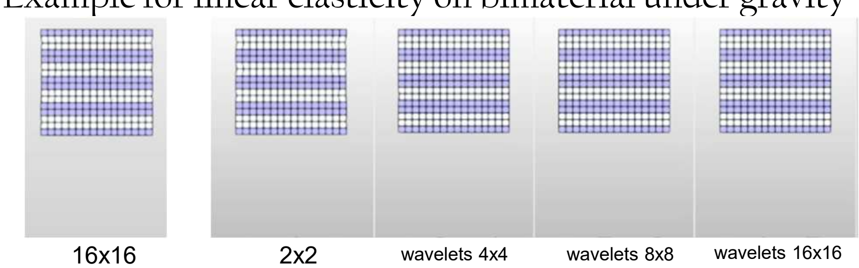
  [Owhadi 2017; Budninskiy et al. 2019, Chen et al. 2019]



- Limitation: still some preprocessing to do....

Best Homogenization Thus Far

Example for linear elasticity on bimaterial under gravity



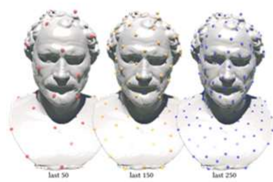
Even for Whitney differential forms

- ex: div-free bases adapted to 1-form Laplacian
- and restricted to embedded domains
  - i.e., edges on a regular grid
  - but bases adapted to a given domain

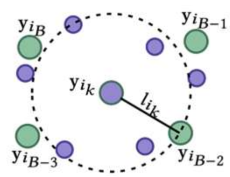
From Homogenization to Solvers?

This line of work surprisingly led to fast linear solvers…

- [Schäfer 2021] proved that homogenization can be rewritten as a simple Cholesky factorization
- homogenization leads to linear complexity solvers (!)
  - one needs new **hierarchical ordering** and **sparsity pattern**
  - then perform **incomplete (reverse) Cholesky factorization**
  - resulting matrix used a preconditioner in conjugate gradient
    - beats typical Cholesky, multigrid preconditioners, SVD, etc…
    - often by **orders of magnitude**

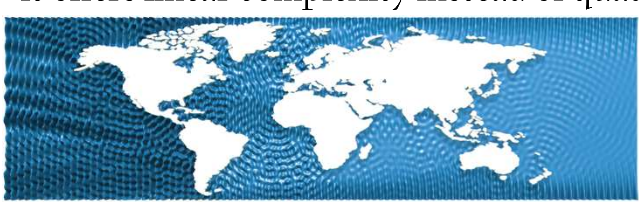


Fine-to-coarse reordering

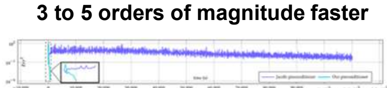


Lightning-Fast BIE Solver

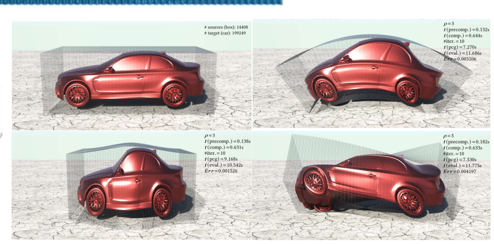
Whether on Helmholtz equation (low wave number) or elasticity, it offers linear complexity instead of quadratic for SVD



Allows >1M dofs on laptops, 3 to 5 orders of magnitude faster

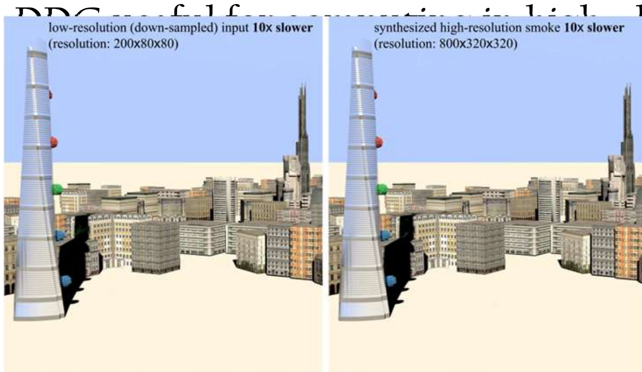


Needs only 9 iterations!



Wrapping Up

ow dims



ential treatments

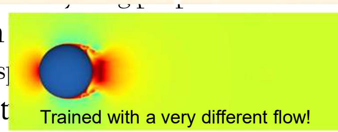
ly

dge!

bra, etc…

you can

- even
  - sp … , for instance
- and t



47

Acknowledgements

Work mostly done by other people

- Fernando de Goes
- Max Budninskiy
- Beibei Liu
- Keenan Crane
- Patrick Mullen
- Evan Gawlik
- Wei Li
- Chaoyang Lv
- Florian Schaefer
- François Gay-Balmaz
- Yiying Tong
- Houman Owhadi
- Jerry Marsden†
- Xiaopei Liu

48

QUESTIONS?

49

49

50