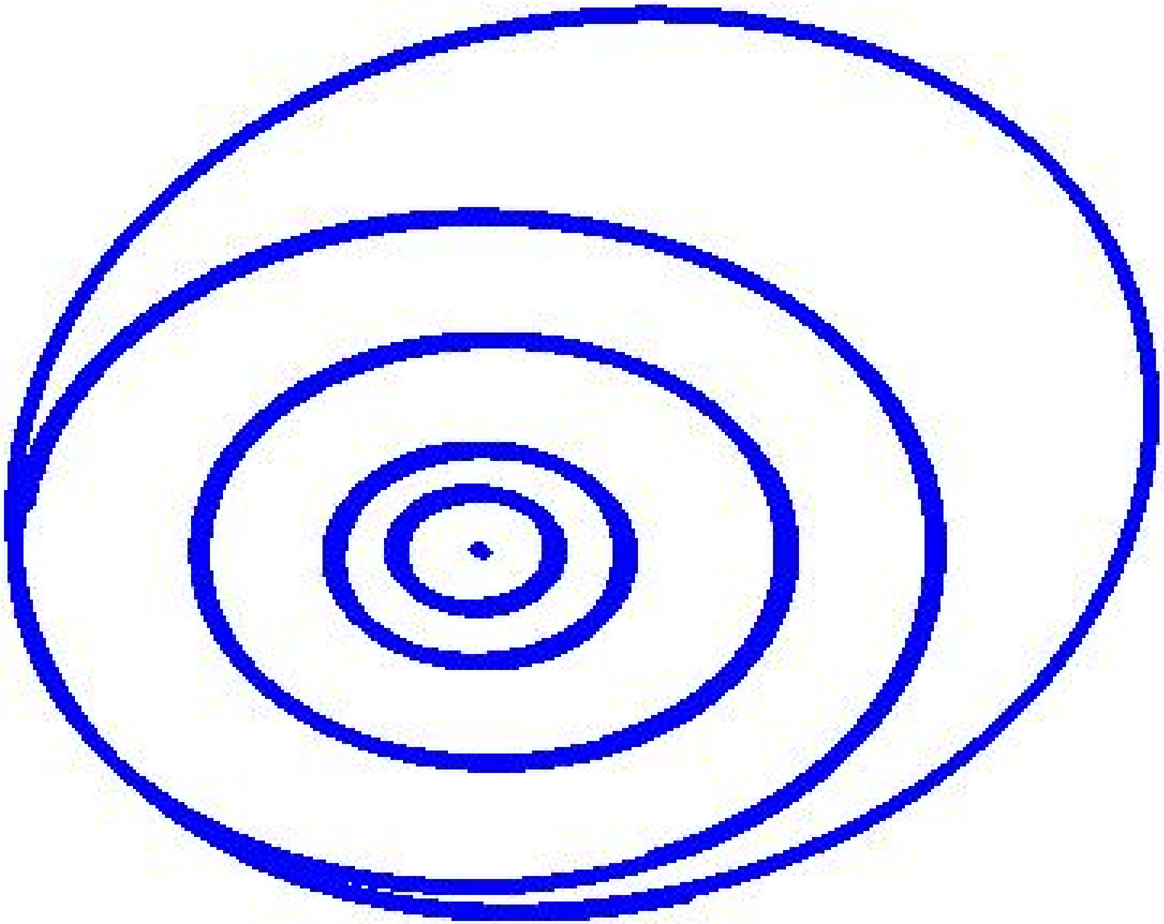
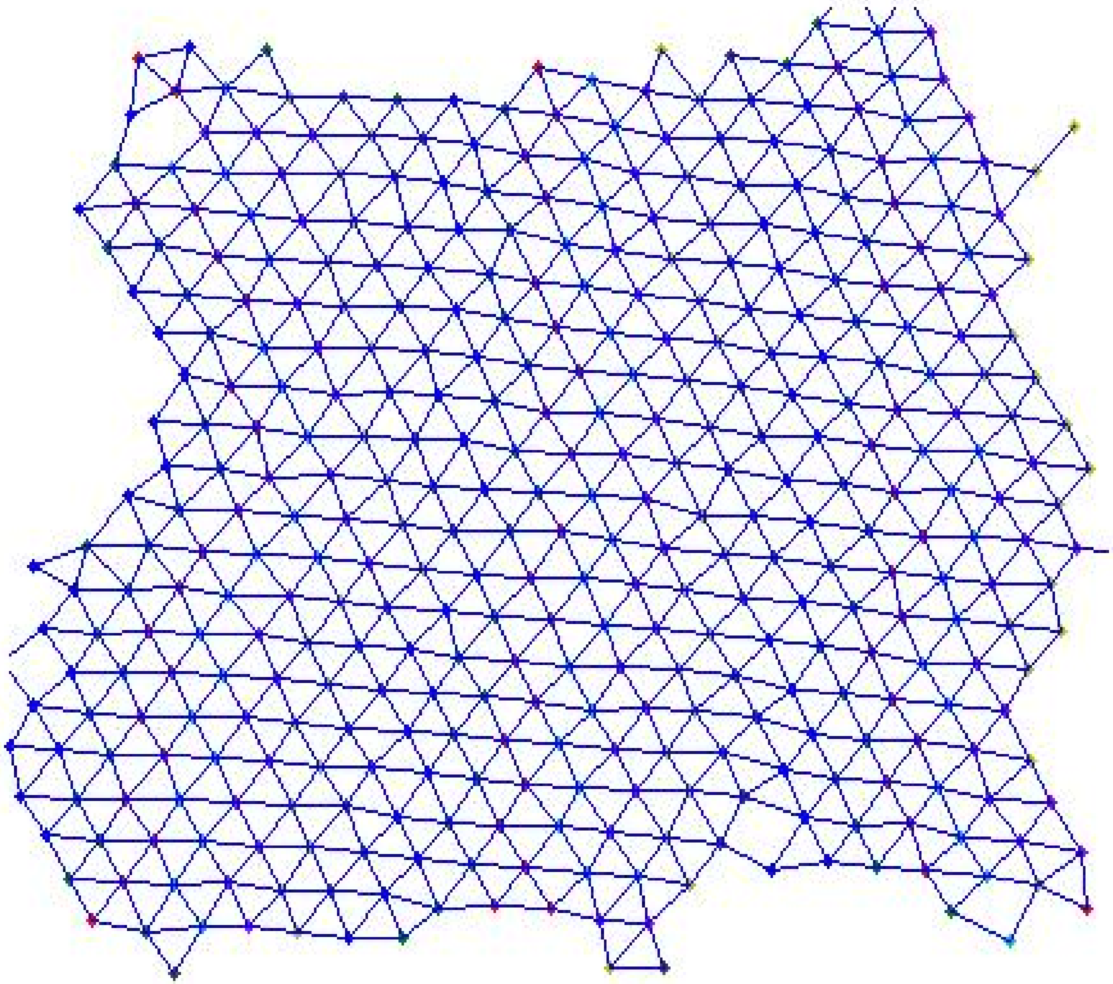
Numerical Methods for Shape Analysis

Stephen Marsland
(with input from Robert McLachlan)
Massey University, New Zealand
http://www-ist.massey.ac.nz/smarsland/


Initial
Final
Change



t= 16000

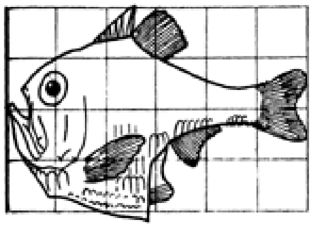


Aim

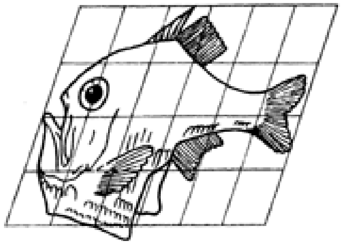
Compute (possibly long-term) motion of a set of points (or curves/surfaces) under some dynamical equations

Numerical Methods

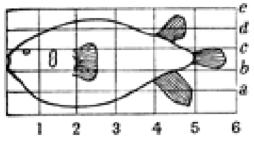
- Numerical Integration
- Optimisation and numerical linear algebra
- Finite Differences and Finite Elements
- Number Representation

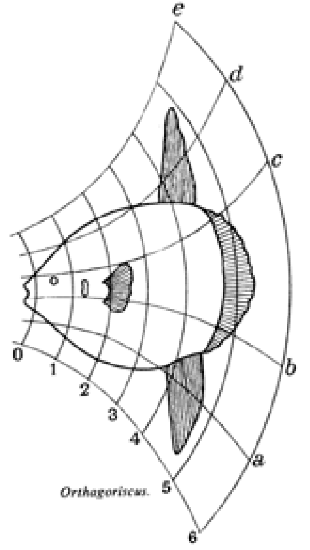


Argyropelecus olfersi.

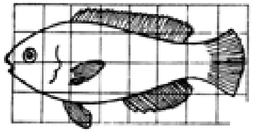


Sternoptyx diaphana.

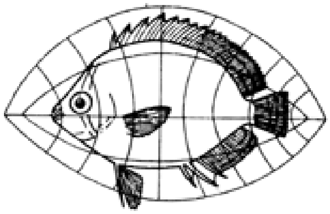




Orthagoriscus.



Scarus sp.



Pomacanthus.

Relevant Books

- Iserles, A First Course in the Numerical Analysis of Dynamical Equations, Cambridge
- Leimkuhler & Reich, Simulating Hamiltonian Dynamics, Cambridge
- Hairer, Lubich & Wanner, Geometric Numerical Integration, Springer
- Lots more
- Numerical Optimisation
  - Nocedal & Wright, Numerical Optimisation, Springer

Getting Started

$$\dot{z} = f(z), z(t_0) = z^0 \in \mathbb{R}^k$$

- Defines a 1-parameter family $\{\phi_t\}_{t \geq 0}$ with flow map

$$\phi_t(z^0) = z(t; z^0) : \mathbb{R}^k \to \mathbb{R}^k$$

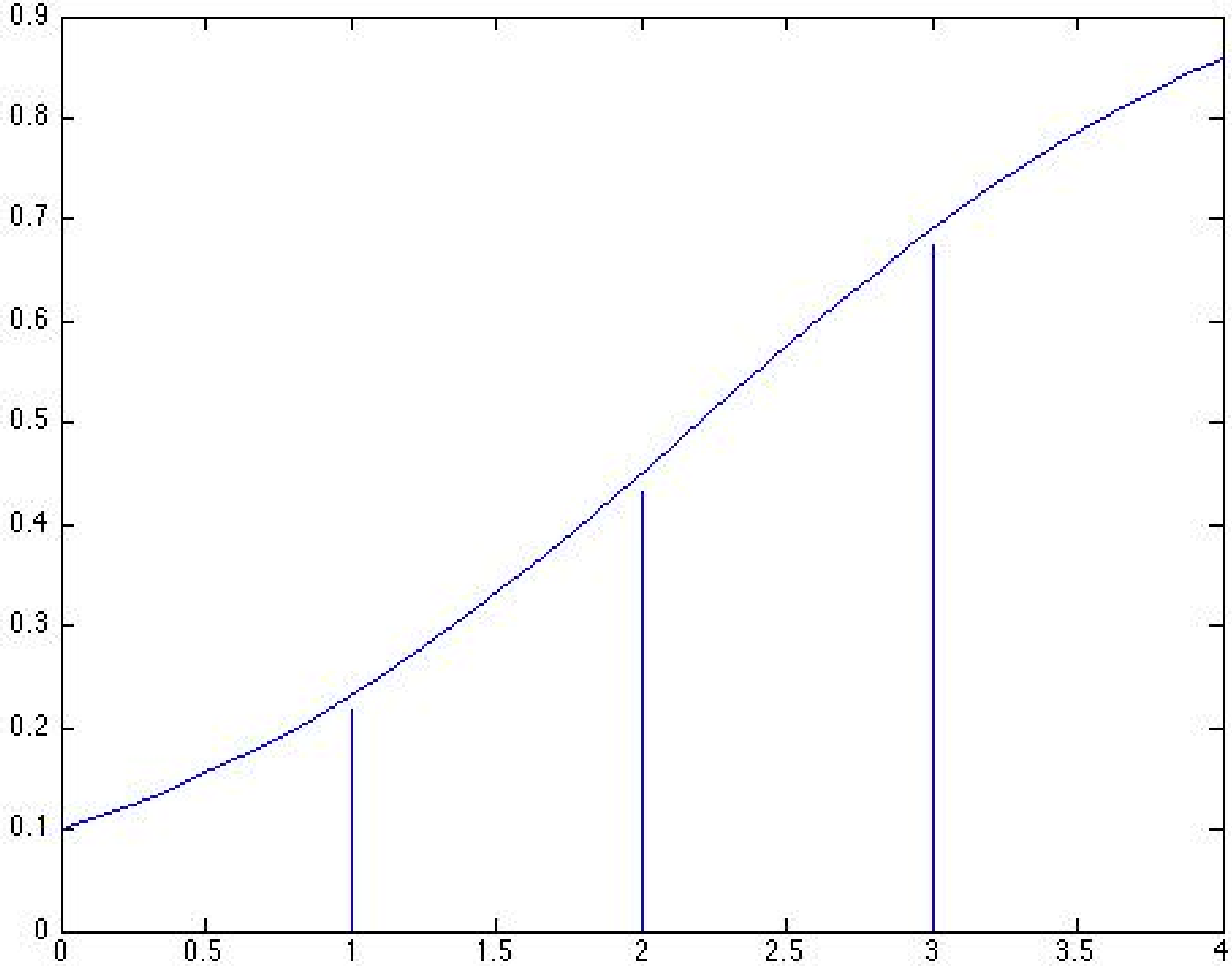
- Make approximations $\{\phi^n_{\Delta t}(z^0)\}_{n=0}^{\infty}$ to true solution $z^n = \psi^n_{\Delta t}(z^0)$

Approximation

- Integrate $\dot{z} = f(z)$ over (small) $[t, t + \delta t]$

$$z(t + \delta t) - z(t) = \int_0^{\delta t} f(z(t + \tau))d\tau$$

$$\approx \sum_{i-1}^{s} b_i f(z(t + \tau_i))$$



$$\dot{z} = z(1 - z), z(0) = \frac{1}{10}$$

Piecewise Linear Approximations

- Euler $$z^{n+1} = z^n + \Delta t f(z^n)$$
- Implicit Euler $$z^{n+1} = z^n + \Delta t f(z^{n+1})$$
- Trapezoidal
$$z^{n+1} = z^n + \frac{1}{2}\Delta t(f(z^n) + f(z^{n+1}))$$
- Implicit Midpoint
$$z^{n+1} = z^n + \Delta t f\left(\frac{1}{2}(z^n + z^{n+1})\right)$$

Second Order System

$$\frac{d^2q}{dt^2} = g(q)$$

- Introduce $\dot{q} = v$, $\dot{v} = g(q)$
- Then solve $$\dot{z} = \begin{pmatrix} v \\ g(q) \end{pmatrix}$$

where $$z = \begin{pmatrix} q \\ v \end{pmatrix}$$

Jacobian

$$f'(z) = \begin{pmatrix} 0 & 1 \\ g'(q) & 0 \end{pmatrix}$$

Second-Order System

- So Euler's method becomes

$$q^{n+1} = q^n + \Delta t\, v^n$$

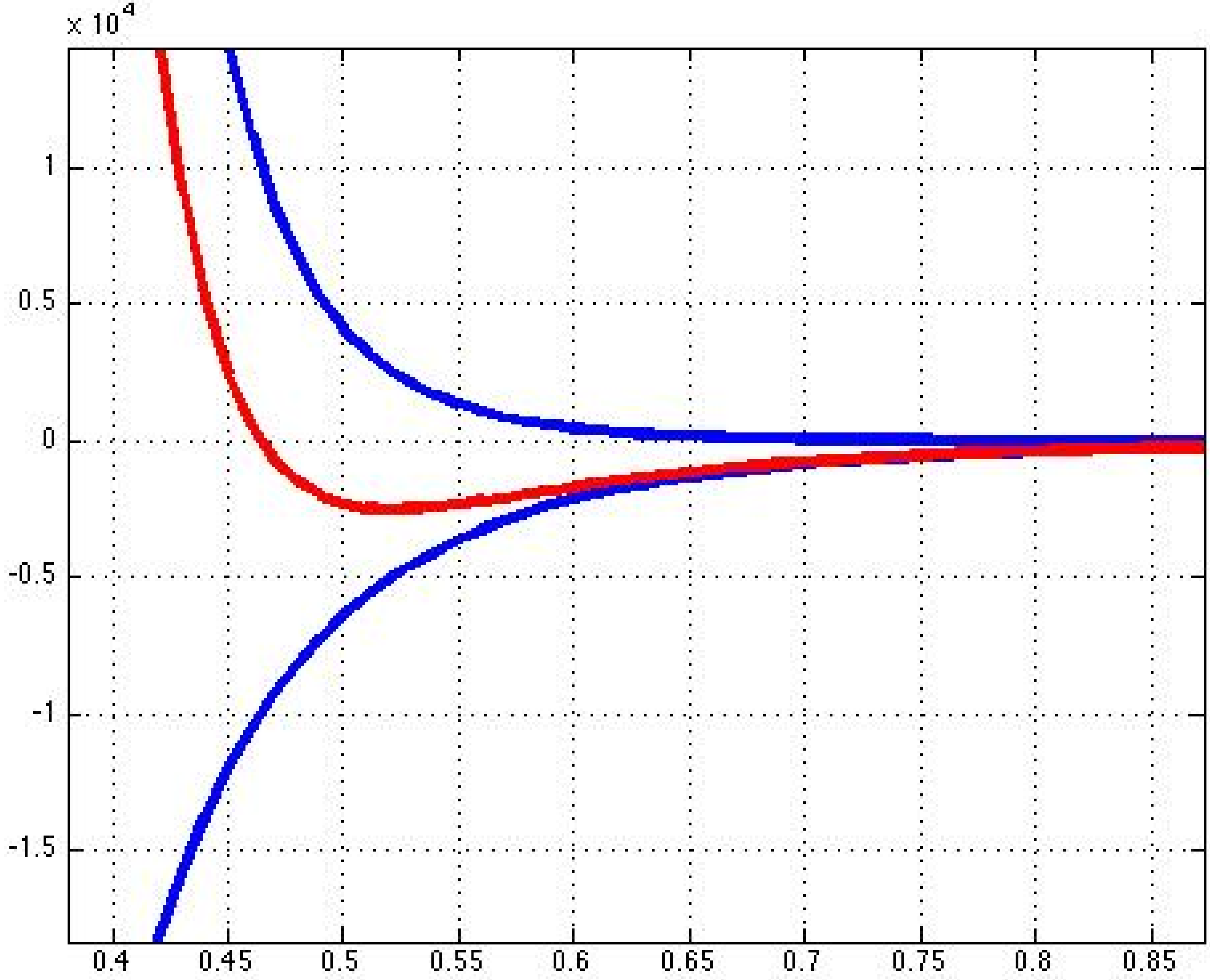
$$v^{n+1} = v^n + \Delta t\, g(q^n)$$

- Example: Lennard-Jones oscillator

$$\dot{q} = v$$

$$\dot{v} = -\phi'(q)$$

$$\phi(q) = \frac{1}{6} q^{-12} - q^{-6}$$



x 10^4
1
0.5
0
-0.5
-1
-1.5
0.4
0.45
0.5
0.55
0.6
0.65
0.7
0.75
0.8
0.85

Higher-Order Integrators

$$z^{n+1} = z^n + \Delta t \sum_{i=1}^{s} b_i f(Z_i)$$

$$Z_i = z^n + \Delta t \sum_{j=1}^{s} a_{ij} f(Z_j)$$

Runge-Kutta 4

$$Z_1 = z^n$$

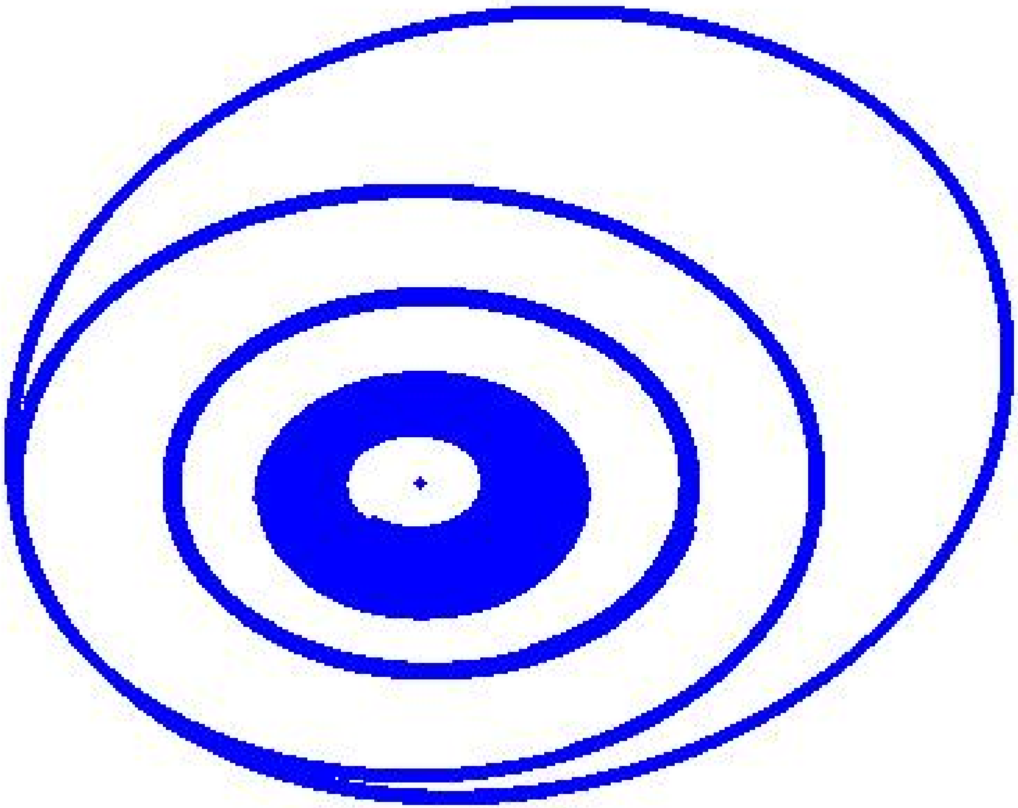
$$Z_2 = z^n + \frac{1}{2}\Delta t\, f(Z_1)$$

$$Z_3 = z^n + \frac{1}{2}\Delta t\, f(Z_2)$$

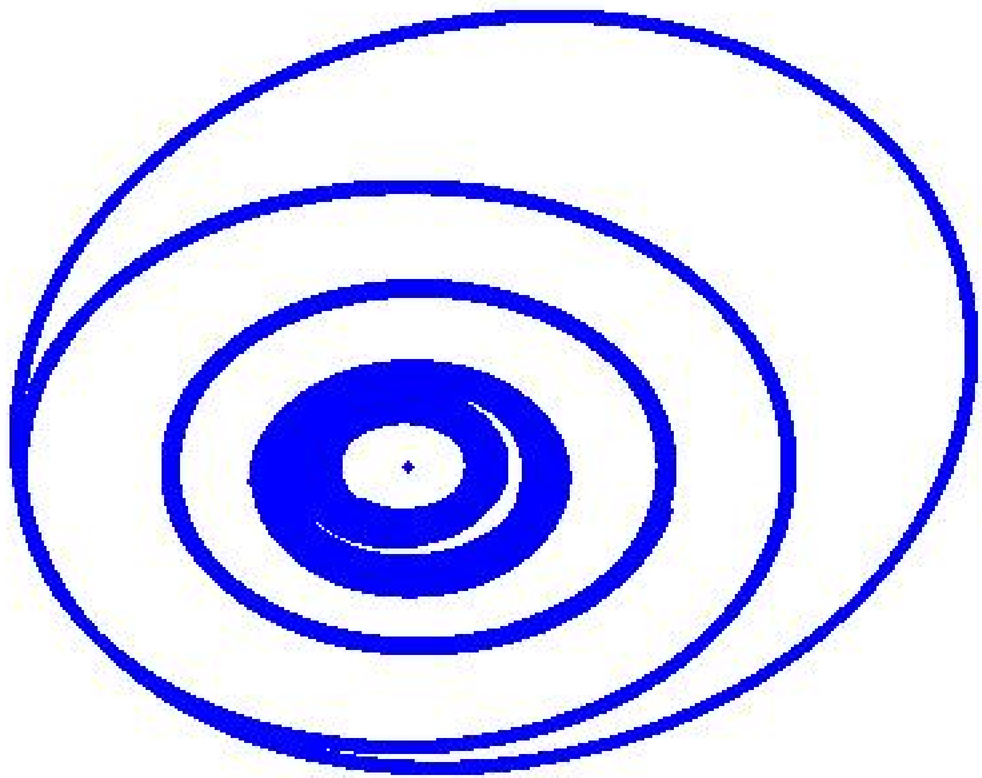
$$Z_4 = z^n + \frac{1}{2}\Delta t\, f(Z_3)$$

$$z^{n+1} = z^n + \frac{1}{6}\Delta t\,(f(Z_1) + 2f(Z_2) + 2f(Z_3) + f(Z_4))$$

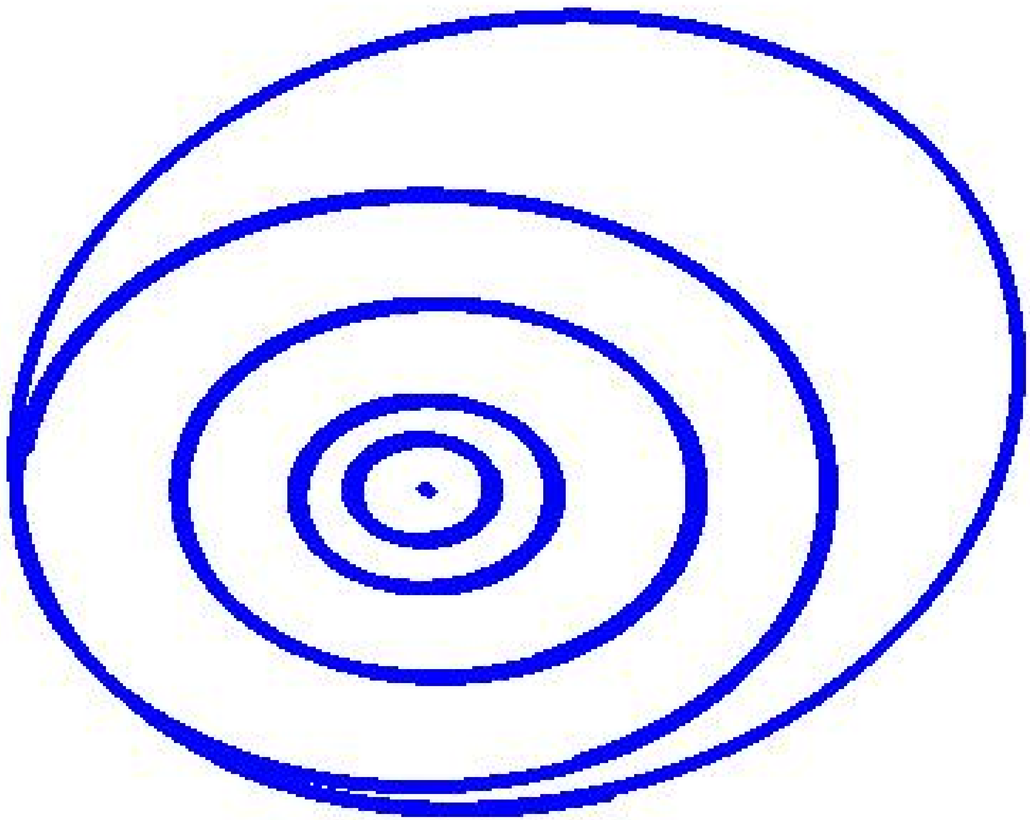
Euler



Midpoint



Runge-Kutta 4



‘Generic’ Procedure

- Choose numerical integration method
- Choose stepsize
- Run
- Monitor error (or perform error analysis)
- Decide whether or not to trust

Hamiltonian Systems

$$H : \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R}$$

$$\dot{q} = \nabla_p H(q,p)$$

$$\dot{p} = -\nabla_q H(q,p)$$

$$z = \begin{pmatrix} q \\ p \end{pmatrix}$$

$$\dot{z} = J \nabla_z H(z)$$

$$J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$$

Structure matrix

Partitioning

- If can split into $\dot{q} = g(q,p)$, $\dot{p} = h(q,p)$
- Can solve independently with different quadrature points

$$q^{n+1} = q^n + \Delta t\, p^{n+1}$$ Euler-A

$$p^{n+1} = p^n - \Delta t\, \nabla_q \phi(q^n)$$

Euler-B

$$q^{n+1} = q^n + \Delta t\, p^n$$

$$p^{n+1} = p^n - \Delta t\, \nabla_q \phi(q^{n+1})$$

Combine with a Halfstep

$$p^{n+\frac{1}{2}} = p^n - \frac{1}{2}\Delta t\, \nabla_q V(q^n)$$
$$q^{n+\frac{1}{2}} = q^n + \Delta t\, p^{n+\frac{1}{2}}$$
$$p^{n+1} = p^{n+\frac{1}{2}} - \frac{1}{2}\Delta t\, \nabla_q V(q^n)$$

- Stormer-Verlet method
- Leapfrog

Properties of Hamiltonian Systems

- H is a first integral

$$H'(y)f(y) = 0 \ \forall y$$

$$H(y(t)) = H(y_0)$$

  - There may be others
- Hamiltonian systems are symplectic
- Hamiltonian systems with smooth bounded H give diffeomorphic flow maps

Symplecticness

- Basic object: 2D parallelogram in $\mathbb{R}^{2d}$
- Spanned by vectors

$$\xi = \begin{pmatrix} \xi^q \\ \xi^p \end{pmatrix} \quad \eta = \begin{pmatrix} \eta^q \\ \eta^p \end{pmatrix}$$

- (d=1) Oriented area $= \det \begin{pmatrix} \xi^q & \eta^q \\ \xi^p & \eta^p \end{pmatrix}$

$$= \xi^q \eta^p - \xi^p \eta^q$$

- Area preservation

Symplecticness

- (d>1) Sum of oriented areas of projections of parallelograms onto (q_i, p_i)

$$\Omega(\xi,\eta) = \sum_{i=1}^{d} \det \begin{pmatrix} \xi_i^q & \eta_i^q \\ \xi_i^p & \eta_i^p \end{pmatrix}$$

$$= \xi^T J \eta$$

Skew-symmetric bilinear function:
Two-form

$$J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$$

Structure matrix

Definition of Symplectic Map

- A map ψ in phase space $\mathbb{R}^{2d}$ is symplectic with respect to structure matrix J if its Jacobian $\psi_z(z)$ satisfies

$$\psi_z(z)^T J^{-1} \psi_z(z) = J^{-1}$$

- (Linear) $A : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ is symplectic if

$$A^T J^{-1} A = J^{-1}$$
$$\Rightarrow \Omega(A\xi, A\eta) = \Omega(\xi, \eta) \;\; \forall \xi, \eta \in \mathbb{R}^{2d}$$

Hamiltonian Flows are Symplectic

- (Poincare, 1899) For Hamiltonian systems with H(q,p) twice continuously differentiable on $U \subset \mathbb{R}^{2d}$, for fixed t, flow ϕ_t is symplectic

The Wedge Product

$$\Omega(\xi,\eta) = \xi^T J^{-1} \eta$$

$$= \sum_{i=1}^{d} \Omega_0(\xi^i, \eta^i)$$

$$= \sum_{i=1}^{d} [dq_i(\xi)dp_i(\eta) - dp_i(\xi)dq_i(\eta)]$$

$$= \sum_{i=1}^{d} dq_i \wedge dp_i \quad = dq \wedge dp$$

The Wedge Product

- Useful way to check symplecticness

$$d\hat{q} = \psi_q^1(q,p)dq + \psi_p^1(q,p)dp$$
$$d\hat{p} = \psi_q^2(q,p)dq + \psi_p^2(q,p)dp$$

- Symplectic means

$$d\hat{q} \wedge d\hat{p} = dq \wedge dp$$

Symplectic Integrators

- Can numerical integrators preserve any of these properties?

$$dq^{n+1} \wedge dp^{n+1} = dq^n \wedge dp^n$$

Euler-B is Symplectic

$$q^{n+1} = q^n + \Delta t \, p^n$$

$$p^{n+1} = p^n - \Delta t \, \nabla_q \phi(q^{n+1})$$

$$\begin{aligned}
dq^{n+1} \wedge dp^{n+1} &= d(q^n + \Delta t \, p^{n+1}) \wedge dp^{n+1} \\
&= dq^n \wedge dp^{n+1} + \Delta t \, dp^{n+1} \wedge dp^{n+1} \\
&= dq^n \wedge d(p^n - \Delta t \nabla_q H(q^n, p^{n+1})) \\
&= dq^n \wedge dp^n
\end{aligned}$$

Midpoint Rule is Symplectic

$$z^{n+\frac{1}{2}} = z^n + \frac{1}{2}\Delta t\, J\nabla H(z^{n+\frac{1}{2}})$$

$$z^{n+1} = z^{n+\frac{1}{2}} + \frac{1}{2}\Delta t\, J\nabla H(z^{n+\frac{1}{2}})$$

- Rewrite:

$$z^{n+1} = z^n + \Delta t\, J\nabla H(z^{n+\frac{1}{2}})$$

$$z^{n+\frac{1}{2}} = \frac{1}{2}\left(z^{n+1} + z^n\right)$$

Midpoint Rule is Symplectic

$$dz^{n+1} = dz^n + \Delta t\, J H_{zz} \frac{1}{2} \left(dz^{n+1} + dz^n \right)$$

- Compute wedge products with $J^{-1} dz^n$ and $J^{-1} dz^{n+1}$

$$\Rightarrow J^{-1} dz^{n+1} \wedge dz^{n+1} = J^{-1} dz^n \wedge dz^n$$

Symplectic Discretisation

- Spatial truncation
  - Reduce PDE to system of Hamiltonian ODES
    - Grid
    - Particles
- Timestep finite dim ODES by symplectic method

Euler Equations

Constructing a diffeomorphic warp requires solving the Euler equations on $\mathrm{Diff}(\mathbb{R}^2)$

$$\dot{m} = \pm \mathrm{ad}^*_{\mathcal{A}^{-1}m} m$$

$$u \to m := \mathcal{A}u$$

Velocity

Momentum

Inertia operator

$$u = \mathbf{G} * m$$

Green's Function

Scalar function for A rotationally invariant and diagonal

Euler Equations on $\mathrm{Diff}(\mathbb{R}^n)$

$$\dot{m} + u \cdot \nabla m + \nabla u^T \cdot m + m(\mathrm{div}\, u) = 0$$

- Also known as *EPDiff*
- Find geodesic by (non-linear) optimisation

- There are exact solutions with momentum concentrated at a finite set of points

For Fluids: Point Vortices

QuickTime™ and a
decompressor
are needed to see this picture.

For Images: Point Particles

- Start from vector field

$$v(t,x) = \sum_{i=1}^{n} \alpha_i G(q_i(t), x)$$

- Compute Lagrangian and discretise via particle ansatz

$$m(x,t) = \sum_{i=1}^{n} p_i(t)\delta(x - q_i(t))$$

Discretisation

- Write as (discrete) Hamiltonian via Legendre transform

$$H = \frac{1}{2}\sum_{i,j} p_i \cdot p_j G(q_i - q_j)$$

- Choose *G* (corresponds to metric)

$$\mathcal{A}_k := (1 - \alpha^2\nabla^2)^k$$

length scale

$$A_k \to A_\infty := \exp(-\varepsilon^2\nabla^2)$$

Getting to Hamiltonian Form

- Legendre transform

$$H(q,p) = p^T\dot{q} - L(q,\dot{q}) \qquad p = \frac{\partial L}{\partial \dot{q}}$$

- Mapping from fibre T_qQ to $T_q{}^*Q$ (fibre derivative)

$$FL(u,e) = (u, D_2L(u,e))$$

Point Particles

$$\dot{q}_i = \sum_j G(\|q_i - q_j\|)p_j$$

$$\dot{p}_i = -\sum_j (p_i \cdot p_j)\nabla G(\|q_i - q_j\|)\frac{q_i - q_j}{\|q_i - q_j\|}$$

- 4 conserved quantities
  - *H*, linear momentum $\sum_i p_i$ and angular momentum $\sum_i q_i \times p_i$

Integration

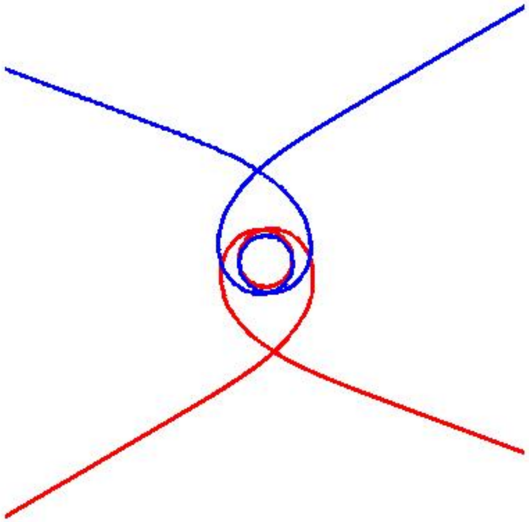
- Now just integrate particles forward in time
  - Euler
  - Runge-Kutta
  - Symplectic Integrator
  - Marker-and-Cell methods
- Use test particles (zero momentum) for rest of image

Entrainment

QuickTime™ and a
H.264 decompressor
are needed to see this picture.

Dynamics of the System

- Depends on choice of metric (via *G*)
- Pairs of particles interact in 3 main ways:
  - – Scattering
  - – Capture ('dipole')
  - – Ejection



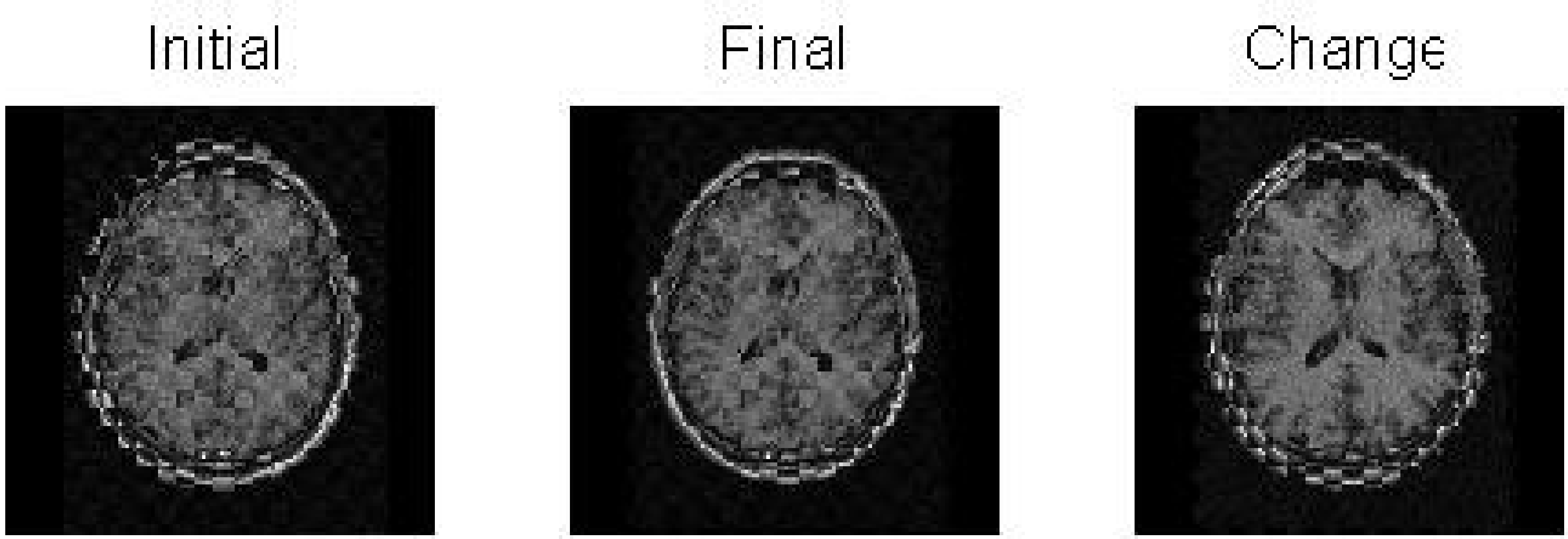
Dipoles

QuickTime™ and a
decompressor
are needed to see this picture.

Finally: Some Image Stuff

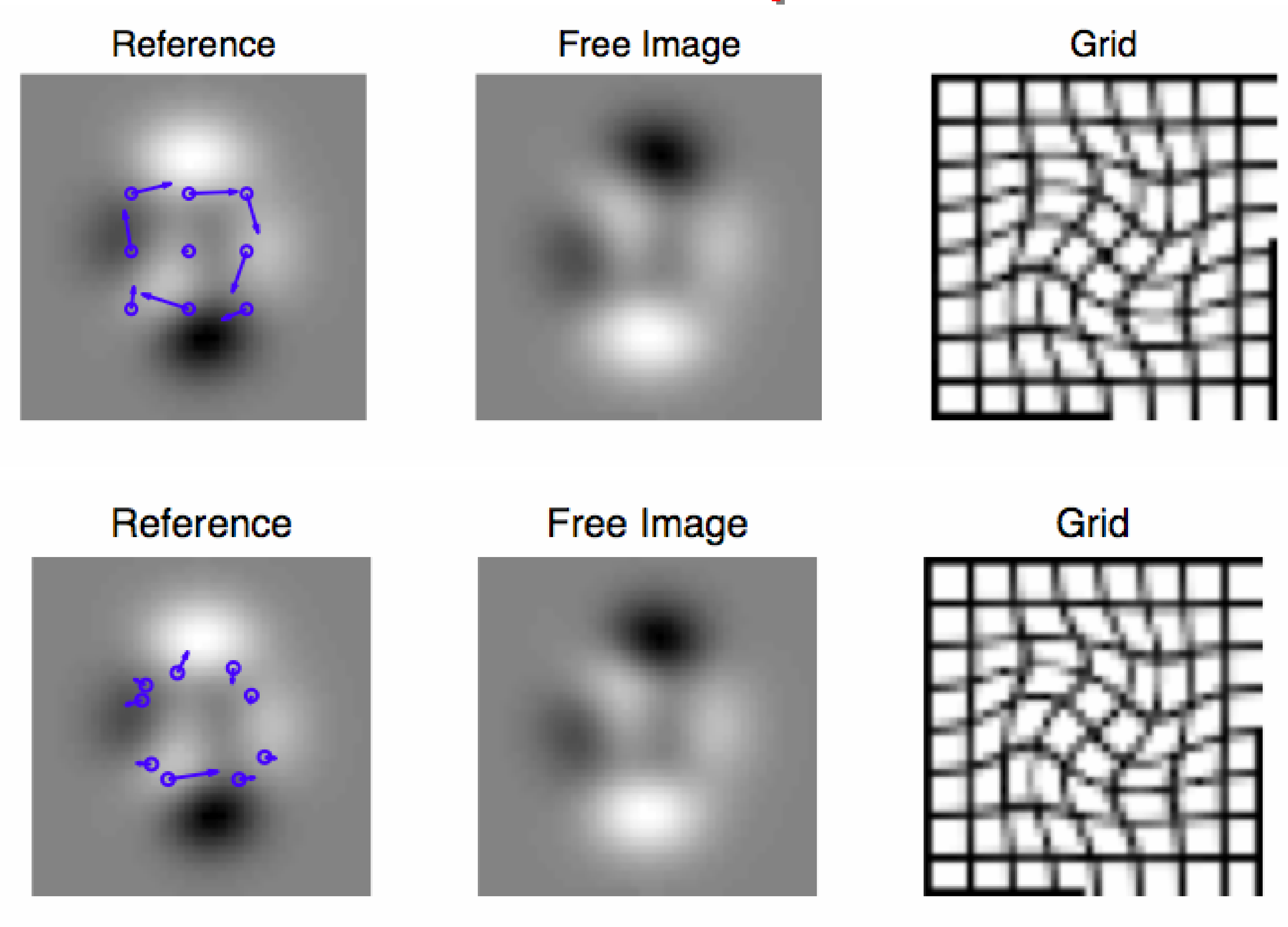
- 9 point particles, 47.7 seconds
- 7 points added, 180 seconds optimisation
- Matlab code

Registration



- 10 point particles on skull (3 minutes)
- 11 more added (3 minutes)

Simultaneous Optimisation



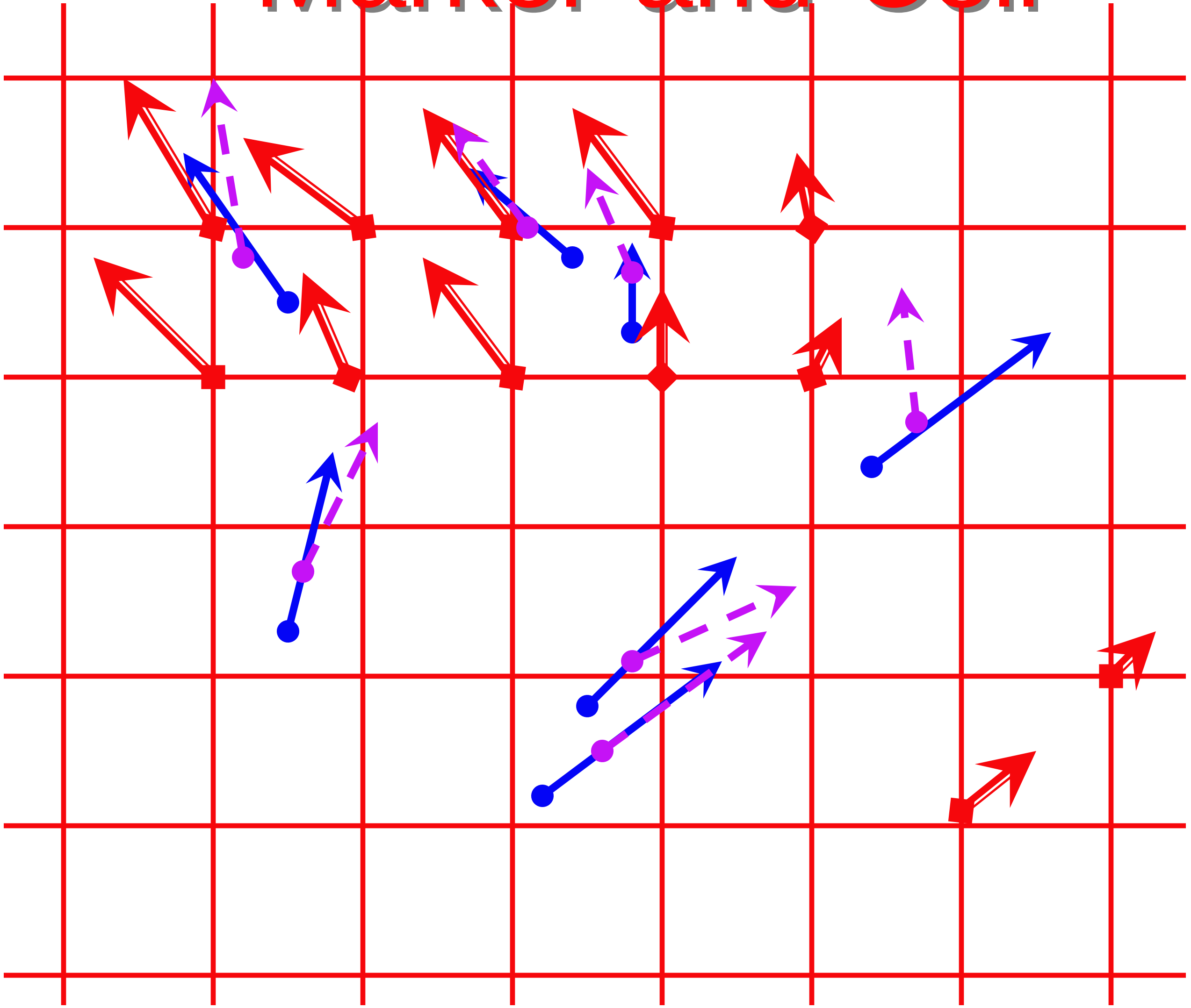
Symplectic Integration

- Equations of motion are Hamiltonian
- Flow therefore symplectic
- Can use a symplectic integrator
- Unfortunately, only have implicit methods
  - And need to solve down to round-off error
  - But we do have good initial guesses
  - And the Jacobian comes for 'free'

Marker-and-Cell

- Interpolate particle momenta onto grid
- Calculate velocity field on grid
  - Fourier transform
  - Multigrid
- Interpolate back to particles
- Used for atmospheric dynamics
  - Over 1 million points
  - Linear in number of particles

Marker-and-Cell



Momentum Sheets

- Consider a line defined by a set of particles
  - Same dynamics
  - Particle relabelling symmetry
- Investigate stability in various metrics for different initial perturbations

QuickTime™ and a
Cinepak decompressor
are needed to see this picture.