CALTECH
Control & Dynamical Systems

Geometric and Computational Dynamics

Jerry Marsden

Control and Dynamical Systems, Caltech

Von Neumann Lecture
SIAM Annual Meeting
Tuesday, July 12, 2005

For papers, movies, etc., visit:
www.cds.caltech.edu/~marsden

Lagrangian Coherent Structures (LCS)

- We begin the lecture by describing a tool from computational dynamical systems applied to fluid mechanics.

Lagrangian Coherent Structures (LCS)

- We begin the lecture by describing a tool from computational dynamical systems applied to fluid mechanics.
- Ideas behind LCS have a complex history, but the main input for my group comes from the work of ***George Haller*** (MIT).[1]

[1] Amongst the many publications, see, for example, G. Haller, [2001], ***Distinguished material surfaces and coherent structures in 3d fluid flows***. *Physica D*, **149**, 248–277, G. Haller [2002], ***Lagrangian coherent structures from approximate velocity data.*** *Phys. Fluids A*, **14**, 1851–1861 and G. Haller [2004], ***Exact theory of unsteady separation for two-dimensional flows.*** *J. Fluid Mech.*, **512**, 257–311.

Lagrangian Coherent Structures (LCS)

- We begin the lecture by describing a tool from computational dynamical systems applied to fluid mechanics.
- Ideas behind LCS have a complex history, but the main input for my group comes from the work of ***George Haller*** (MIT).[1]
- Lets first have a look at an example of what the LCS tool can reveal in a particular fluid system.

[1]Amongst the many publications, see, for example, G. Haller, [2001], ***Distinguished material surfaces and coherent structures in 3d fluid flows***. *Physica D*, **149**, 248–277, G. Haller [2002], ***Lagrangian coherent structures from approximate velocity data.*** *Phys. Fluids A*, **14**, 1851–1861 and G. Haller [2004], ***Exact theory of unsteady separation for two-dimensional flows.*** *J. Fluid Mech.*, **512**, 257–311.

LCS for a GLAS-II Airfoil

How this is computed:

- **1.** Get hold of the velocity field[a]
- **2.** Compute the LCS (below)[b]
- **3.** Place particles on either side of the computed LCS
- **4.** Let things flow
- **Take Home:** ***LCS divides particles with different dynamical behavior***—like a separatrix
- Not so easy to do "by hand" for ***unsteady*** flows

[a]In this case a CFD computation that was provided by Jeff Eldredge, UCLA

[b]The computation was done by Shawn Shadden

FTLE Field

- Fluid particles satisfy (say in $\mathbb{R}^n$, $n = 2, 3$),

$$\dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t)$$

FTLE Field

- Fluid particles satisfy (say in $\mathbb{R}^n$, $n = 2, 3$),

$$\dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t)$$

- **Deformation Gradient**: F = derivative of the solution $\mathbf{x}$ wrt initial conditions (start time t_0, end time $t = t_0 + T$); F is an $n \times n$ matrix.

FTLE Field

- Fluid particles satisfy (say in $\mathbb{R}^n$, $n = 2, 3$),
$$\dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t)$$
- **Deformation Gradient**: F = derivative of the solution $\mathbf{x}$ wrt initial conditions (start time t_0, end time $t = t_0 + T$); F is an $n \times n$ matrix.
- **Cauchy Green tensor**: the matrix $C = F^{\mathrm{T}}F$; measures how the flow deforms the inner product.

FTLE Field

- Fluid particles satisfy (say in $\mathbb{R}^n$, $n = 2, 3$),

$$\dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t)$$

- **Deformation Gradient**: F = derivative of the solution $\mathbf{x}$ wrt initial conditions (start time t_0, end time $t = t_0 + T$); F is an $n \times n$ matrix.
- **Cauchy Green tensor**: the matrix $C = F^{\mathrm{T}} F$; measures how the flow deforms the inner product.
- **FTLE (Finite Time Liapunov Exponent)** is

$$\sigma = \frac{1}{|T|} \log \sqrt{\lambda_{\max}(C)}\ .$$

Definition of LCS

- An ***LCS (Lagrangian Coherent Structure)*** is a ***ridge*** in the FTLE field. LCS are time dependent ***curves in 2d flows*** and ***surfaces in 3d flows***.

Definition of LCS

- An ***LCS (Lagrangian Coherent Structure)*** is a ***ridge*** in the FTLE field. LCS are time dependent ***curves in 2d flows*** and ***surfaces in 3d flows***.
- **Ridge:** roughly (for flows in the plane), a curve such that ***transverse to the curve, the FTLE field is falling off the fastest***— like a mountain ridge.

Definition of LCS

- An ***LCS (Lagrangian Coherent Structure)*** is a ***ridge*** in the FTLE field. LCS are time dependent ***curves in 2d flows*** and ***surfaces in 3d flows***.
- **Ridge:** roughly (for flows in the plane), a curve such that ***transverse to the curve, the FTLE field is falling off the fastest***— like a mountain ridge.
- An LCS depends on the variable starting time; choose the ***run time*** T long enough so the computations resolve.

Definition of LCS

- An ***LCS (Lagrangian Coherent Structure)*** is a ***ridge*** in the FTLE field. LCS are time dependent ***curves in 2d flows*** and ***surfaces in 3d flows***.
- **Ridge:** roughly (for flows in the plane), a curve such that ***transverse to the curve, the FTLE field is falling off the fastest***— like a mountain ridge.
- An LCS depends on the variable starting time; choose the ***run time*** T long enough so the computations resolve.
- For $T > 0$ one gets a ***repelling LCS*** (like a stable manifold), while for $T < 0$, one gets an ***attracting LCS*** (like an unstable manifold).

LCS in Monterey Bay

Hard to tell what the LCS is from the velocity field alone

- **1.** Get hold of the velocity field[a]
- **2.** Compute the LCS
- **3.** Place particles on either side of the computed LCS
- **4.** Let things flow
- ***Take Home:*** LCS divides the particles with different dynamical behaviors, like a separatrix
- LCS still works for rather complex multiscale flows

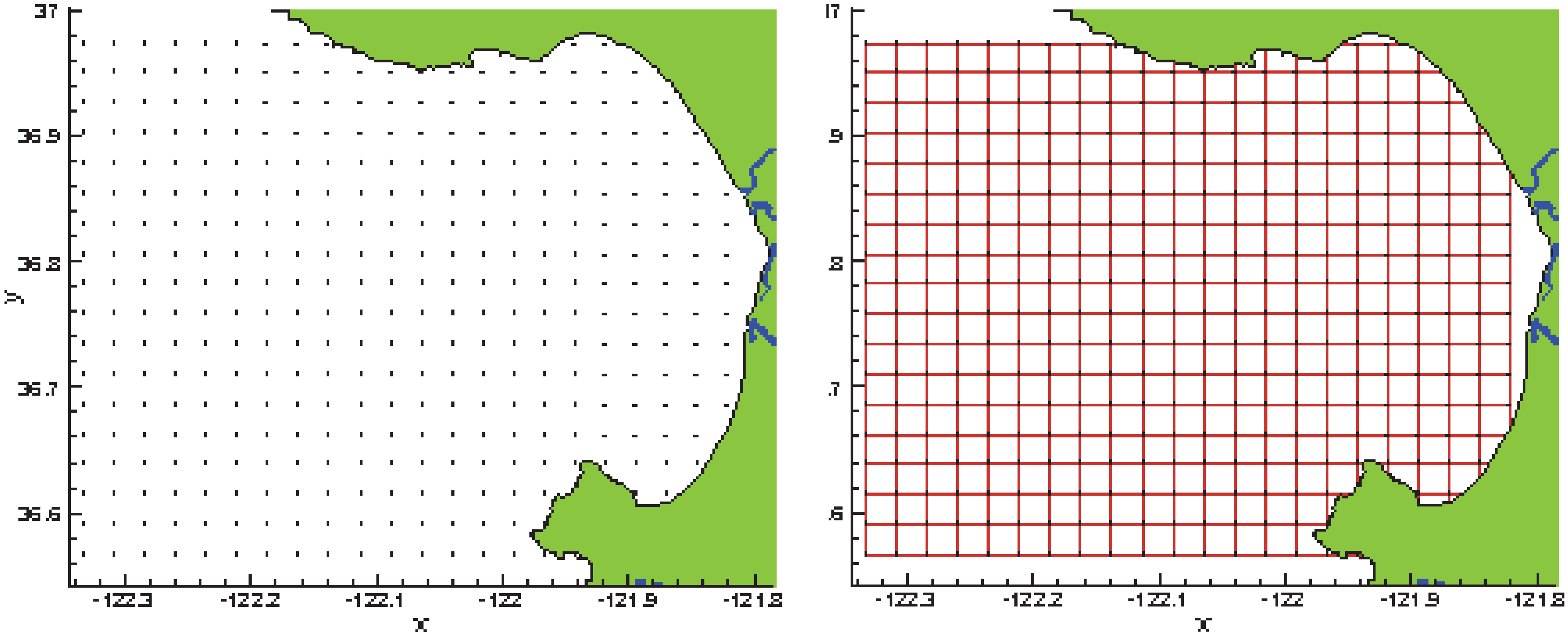
[a]Data obtained from radar (Jeff Paduan) or from HOPS (Harvard Ocean Prediction System)

Compare w/ Experiment

- Can compare the movement of real drifters placed in the Bay with the evolution of the LCS
- LCS also help to navigate efficiently—such as invariant manifolds in the solar systems (later)
- LCS also correlate with interesting ocean features, such as biological fronts

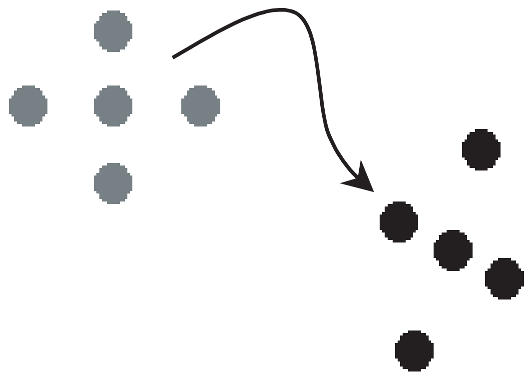
LCS Computation

- **Step 1**: seed the domain with a grid of tracers



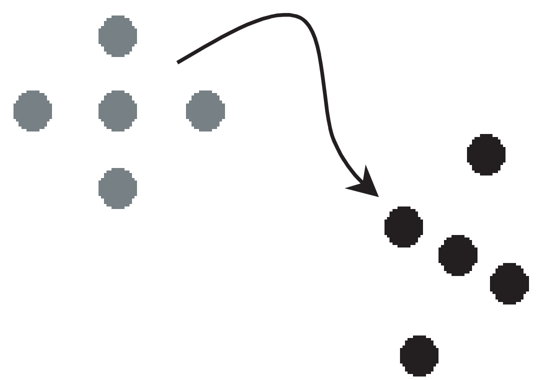
LCS Computation

- **Step 2**: advect groups of particles on the grid



LCS Computation

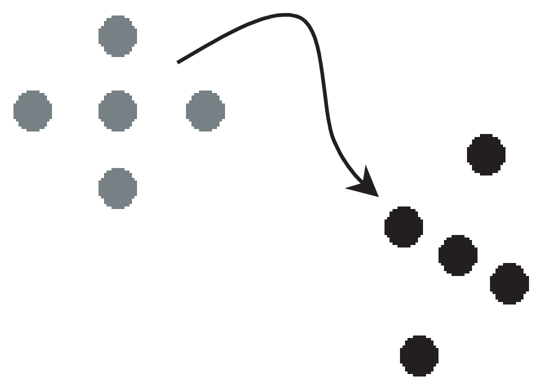
- **Step 2**: advect groups of particles on the grid



- **Step 3**: Compute F approximately using, for example, central differences.

LCS Computation

- **Step 2**: advect groups of particles on the grid



- **Step 3**: Compute F approximately using, for example, central differences.
- **Step 4**: Plug into the definition of the FTLE field, color code the values of the FTLE field, and compute the ridges to get the LCS.

Software

- Software—***MANGEN*** (MANIFOLD GENERATOR) that is able to compute, reasonably automatically, LCS structures as well as other things of interest, such as transport rates.[2]

[2]Mangen was written by Chad Coulliette and Francois Lekien with recent improvements by Shawn Shadden; see Lekien, F. [2003], ***Time-Dependent Dynamical Systems and Geophysical Flows***, *Thesis, California Institute of Technology.*

Software

- Software—***MANGEN*** (MANIFOLD GENERATOR) that is able to compute, reasonably automatically, LCS structures as well as other things of interest, such as transport rates.[2]
- It handles velocity fields that are explicitly given, given from models and assimilated data (such as HOPS) or generated by computational models.

[2]Mangen was written by Chad Coulliette and Francois Lekien with recent improvements by Shawn Shadden; see Lekien, F. [2003], ***Time-Dependent Dynamical Systems and Geophysical Flows***, *Thesis, California Institute of Technology.*

Software

- Software—***MANGEN*** (MANIFOLD GENERATOR) that is able to compute, reasonably automatically, LCS structures as well as other things of interest, such as transport rates.[2]
- It handles velocity fields that are explicitly given, given from models and assimilated data (such as HOPS) or generated by computational models.
- One must be able to compute lots of particle trajectories, and ***systematically*** analyze them (compute FTLE fields and then ridges in these fields).

[2]Mangen was written by Chad Coulliette and Francois Lekien with recent improvements by Shawn Shadden; see Lekien, F. [2003], ***Time-Dependent Dynamical Systems and Geophysical Flows***, *Thesis, California Institute of Technology.*

LCS as Lagrangian Barriers

- The main theoretical result[3] of our work is an estimate on the ***flux across an LCS***, which in turn is a measure of how much of a barrier that LCS is.

[3]Shadden, S. C., F. Lekien, and J. E. Marsden [2005], ***Definition and Properties of Lagrangian Coherent Structures: Mixing and Transport in Two-Dimensional Aperiodic Flows***, *Physica D (submitted)*; see http://www.lekien.com/~francois/papers/qlcs/.

LCS as Lagrangian Barriers

- The main theoretical result[3] of our work is an estimate on the ***flux across an LCS***, which in turn is a measure of how much of a barrier that LCS is.
- The main result says that ***if the FTLE ridge is reasonably sharp and if the run time T is reasonably large, then the LCS is indeed a Lagrangian barrier***.

[3] Shadden, S. C., F. Lekien, and J. E. Marsden [2005], ***Definition and Properties of Lagrangian Coherent Structures: Mixing and Transport in Two-Dimensional Aperiodic Flows***, *Physica D (submitted)*; see http://www.lekien.com/~francois/papers/qlcs/.

LCS as Lagrangian Barriers

- The main theoretical result[3] of our work is an estimate on the ***flux across an LCS***, which in turn is a measure of how much of a barrier that LCS is.
- The main result says that ***if the FTLE ridge is reasonably sharp and if the run time T is reasonably large, then the LCS is indeed a Lagrangian barrier***.
- Numerical tests show that in practice, the flux is indeed small.

[3]Shadden, S. C., F. Lekien, and J. E. Marsden [2005], ***Definition and Properties of Lagrangian Coherent Structures: Mixing and Transport in Two-Dimensional Aperiodic Flows***, *Physica D (submitted)*; see http://www.lekien.com/~francois/papers/qlcs/.

Pollution Studies

LCS can be used for studying pollution release

- **1.** Get hold of the velocity field[a]
- **2.** Compute the LCS
- **3.** Release pollutants on either side of the LCS
- **4.** Let things flow
- ***Take Home:*** LCS can give insight into what happens to the release of pollutants.

[a]Data obtained from radar off the coast of Florida; F. Lekien, C. Coulliette, A.J. Mariano, E.H. Ryan, L.K. Shay, G. Haller, J.E. Marsden ***Pollution release tied to invariant manifolds: A case study for the coast of Florida,*** *Physica D*, (in press), 2005

Vortex Rings

Vortex rings are ubiquitous

- Naturally occur in biology; eg, jellyfish, squid[a]
- Useful for certain highly manouverable underwater vehicles (Helmholtz cavities)
- Create vortex rings in the lab[b]

[a] John Dabiri et al., ***J. of Experimental Biology***, **209** (2005)

[b] John Dabiri and Mory Gharib, ***J. of Fluid Mechanics*** **511** (2004)

LCS and Vortex Boundaries

- LCS can detect the boundaries of vortex rings.[4]

[4]Shadden, S. C., J. O. Dabiri, and J. E. Marsden [2005], ***Lagrangian analysis of entrained and detrained fluid in vortex rings***, *J. Fluid Mech. (submitted)*.

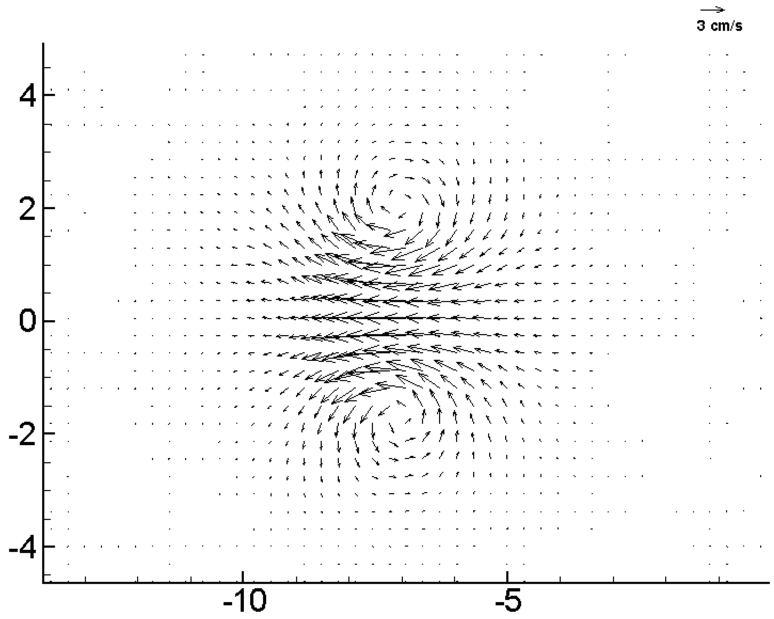
LCS and Vortex Boundaries

- LCS can detect the boundaries of vortex rings.[4]
- Computation uses experimental data (PIV).

[4] Shadden, S. C., J. O. Dabiri, and J. E. Marsden [2005], ***Lagrangian analysis of entrained and detrained fluid in vortex rings***, *J. Fluid Mech. (submitted).*

LCS and Vortex Boundaries

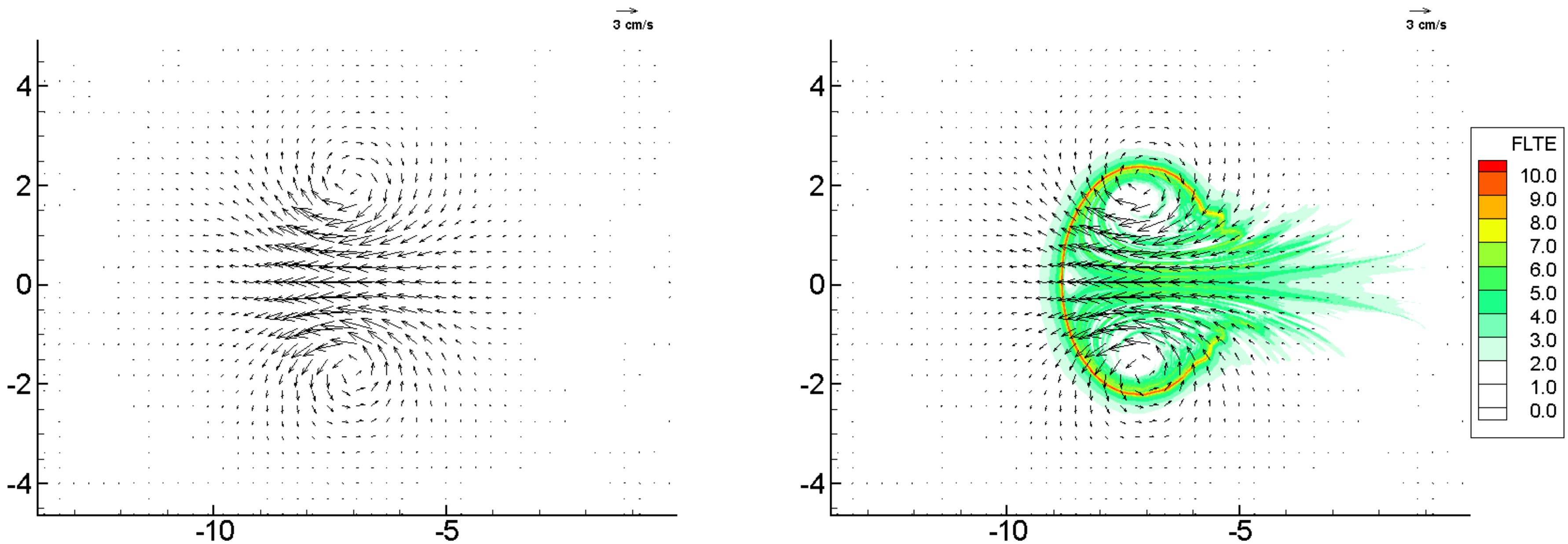
- LCS can detect the boundaries of vortex rings.[4]
- Computation uses experimental data (PIV).
- Can you guess what the boundary is?



[4] Shadden, S. C., J. O. Dabiri, and J. E. Marsden [2005], ***Lagrangian analysis of entrained and detrained fluid in vortex rings***, *J. Fluid Mech. (submitted)*.

LCS and Vortex Boundaries

- How correct were you?



- How do we know that LCS really is the boundary?

Vortex Ring Entraining

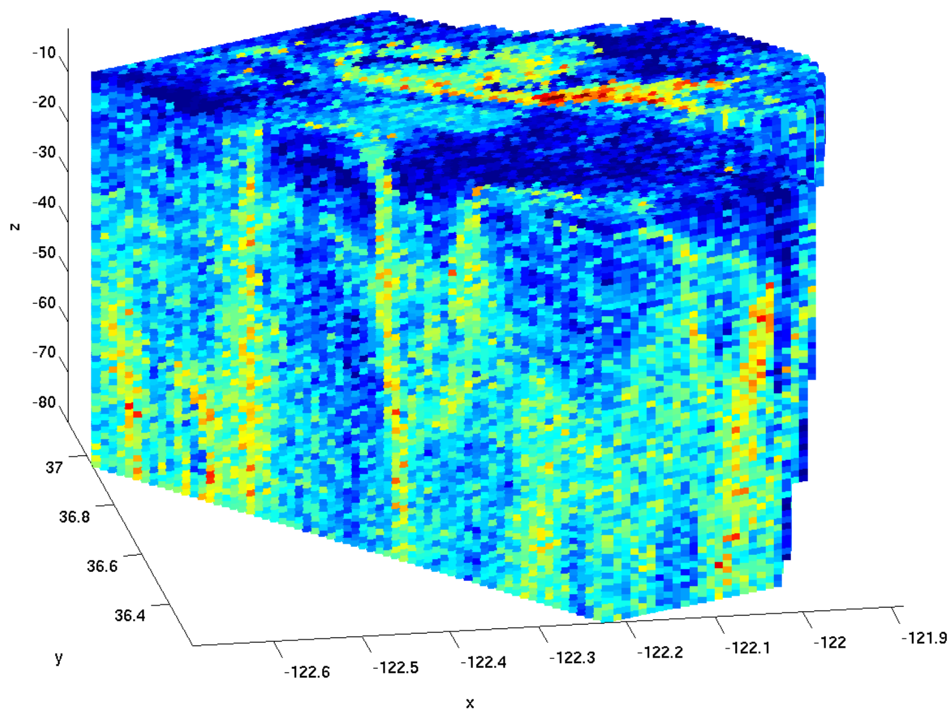
LCS gives detail about how entraining occurs

- First movie shows that LCS really is a good boundary

- Second shows what looks like "heteroclinic lobes" and how they are responsible for entraining and detraining

Extension to 3D

- Progress on extending these types of calculations to 3D using ***GAIO—Global Analysis of (Almost) Invariant Objects***[5]



[5] More on GAIO shortly; this calculation was done by Kathrin Padberg, Paderborn.

Future LCS Computations

- Cardiovascular studies (Charley Taylor, Mory Gharib, John Dabiri)

Future LCS Computations

- Cardiovascular studies (Charley Taylor, Mory Gharib, John Dabiri)
- Atmospheric studies such as the polar vortex and the South Pole ozone hole break up (following Jones and Winkler, Paul Newton, Shane Ross, Tapio Schneider)

Future LCS Computations

- Cardiovascular studies (Charley Taylor, Mory Gharib, John Dabiri)
- Atmospheric studies such as the polar vortex and the South Pole ozone hole break up (following Jones and Winkler, Paul Newton, Shane Ross, Tapio Schneider)
- Microfluidics (Igor Mezic, Sandra Troian)

More CDS Tools—GAIO

- **Context:** Dynamics and transport problems in fluid mechanics, astrodynamics, celestial mechanics, mission design, chemical reaction rates.

More CDS Tools—GAIO

- **Context:** Dynamics and transport problems in fluid mechanics, astrodynamics, celestial mechanics, mission design, chemical reaction rates.
- **Two Key Concepts.** ***Almost invariant sets (& associated transport rates between them) and Conley-McGehee tubes.***

More CDS Tools—GAIO

- **Context:** Dynamics and transport problems in fluid mechanics, astrodynamics, celestial mechanics, mission design, chemical reaction rates.
- **Two Key Concepts.** ***Almost invariant sets (& associated transport rates between them) and Conley-McGehee tubes.***
- **Example** Transport in the solar system and application to Mars crossers[6]

[6] M. Dellnitz, O. Junge, W. S. Koon, F. Lekien, M. W. Lo, J. E. Marsden, K. Padberg, R. Preis, S. Ross, and B. Thiere [2005], ***Transport in dynamical astronomy and multibody problems***, *Intern. J. of Bifurcation and Chaos* **15**, 699–727; Dellnitz, M., O. Junge, M. W. Lo, J. E. Marsden, K. Padberg, R. Preis, S. Ross, and B. Thiere [2005], ***Transport of Mars-crossers from the quasi-Hilda region***, *Physical Review Letters (231102)* **94**, 1–4.

Mars-Crossers

- GAIO uses ***transfer operators*** associated with box subdivision

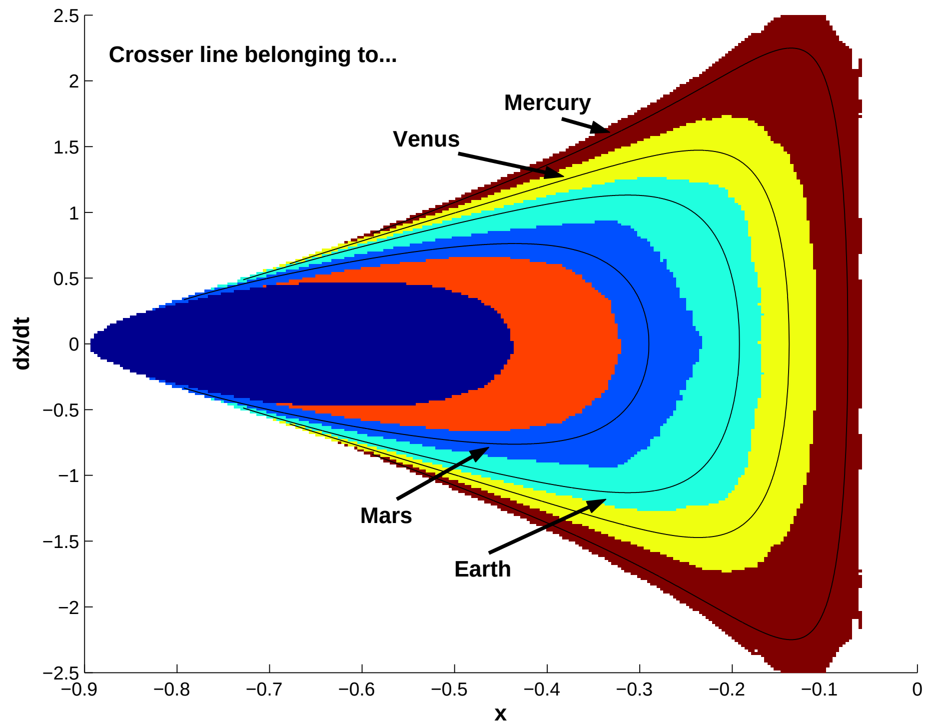
Mars-Crossers

- GAIO uses ***transfer operators*** associated with box subdivision
- Many applications to molecular and other systems[7]

[7]See, eg, Junge, O., J. E. Marsden, and I. Mezic [2004], ***Uncertainty in the dynamics of conservative maps***, *Proc CDC* **43**, 2225–2230.

Mars-Crossers

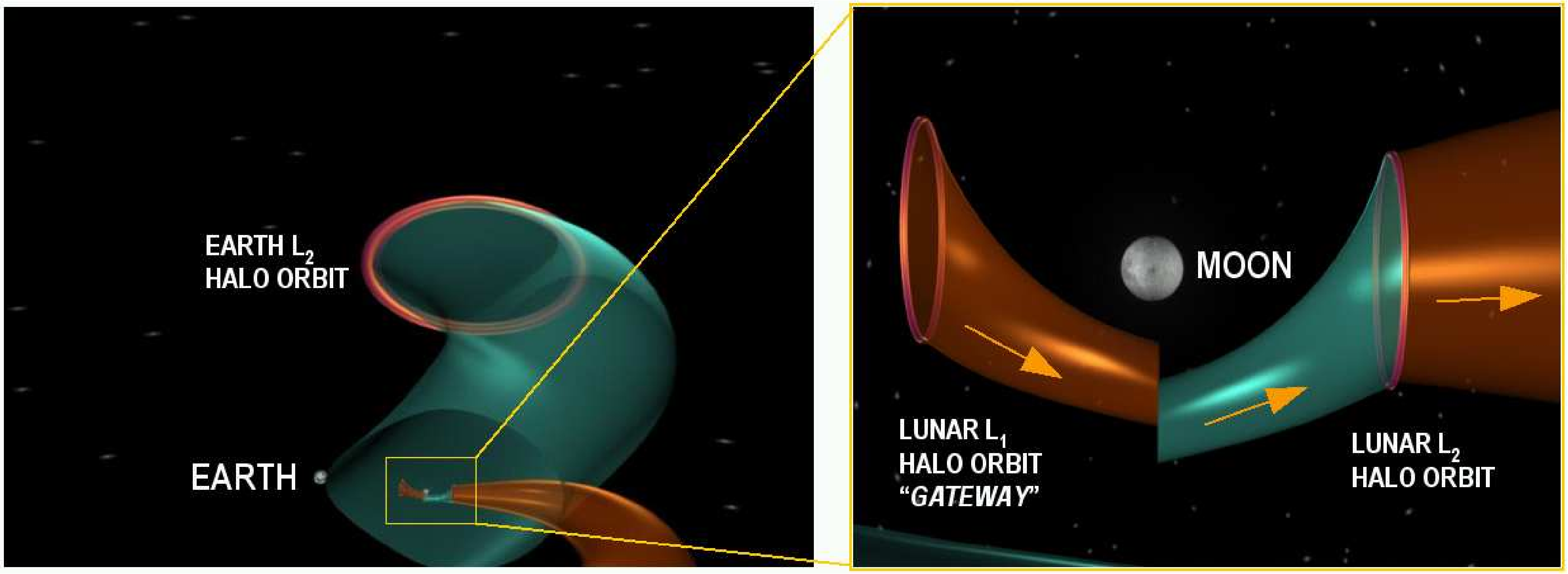
- GAIO uses ***transfer operators*** associated with box subdivision
- Many applications to molecular and other systems[7]
- Almost invariant sets and planetary crosser lines in the three-body system: ***Sun-Jupiter-third body***.



[7]See, eg, Junge, O., J. E. Marsden, and I. Mezic [2004], ***Uncertainty in the dynamics of conservative maps***, *Proc CDC* **43**, 2225–2230.

Tubes on Molecular and Galactic Scales

- Invariant manifold tubes play an important role in mission design, from the Genesis Discovery Mission, to cheap missions to the moon, to the Lunar gateway, to multi-moon orbiters,...[8]



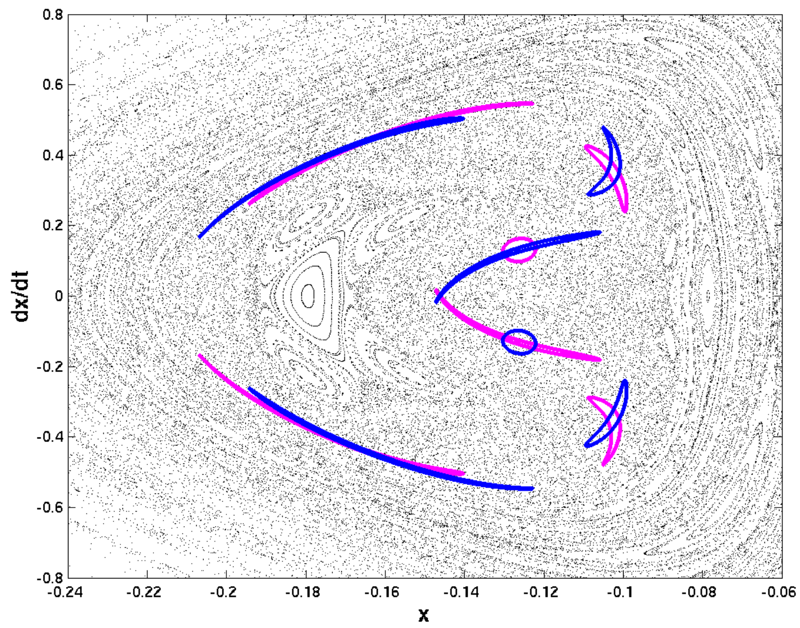
[8] Koon, W. S., M. Lo, J. E. Marsden, and S. Ross [2000], ***Heteroclinic connections between periodic orbits and resonance transitions in celestial mechanics***, *Chaos* **10**, 427–469.

Molecular Tubes

- Tubes mediate transport between realms & are present in molecular systems—connecting, e.g., ***reactants*** and ***products***.

Molecular Tubes

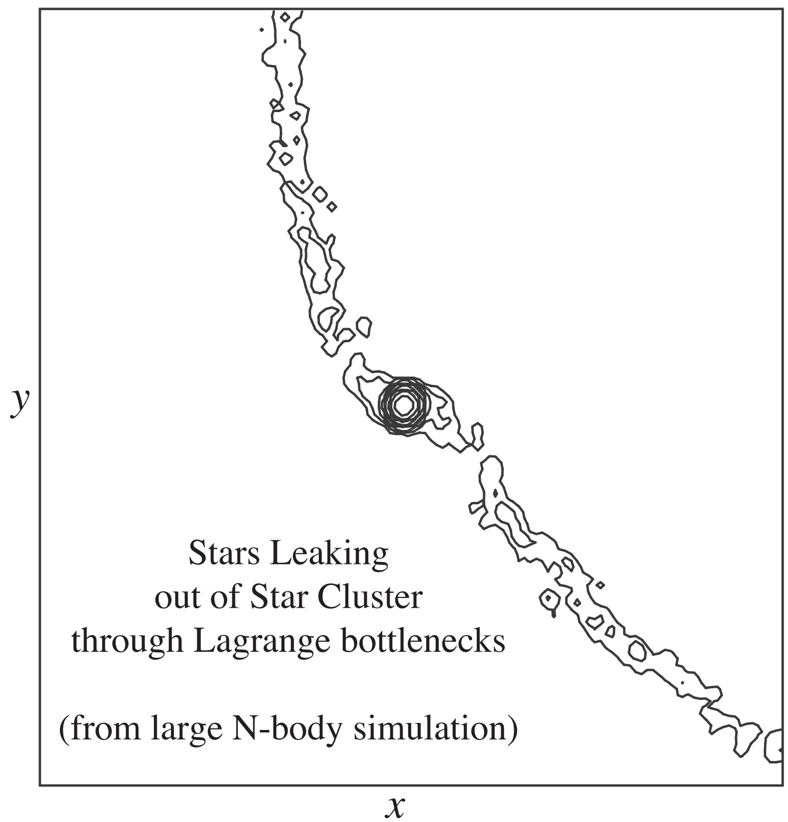
- Tubes mediate transport between realms & are present in molecular systems—connecting, e.g., ***reactants*** and ***products***.
- CDS tools enable reaction rate computation for 3+ dof systems.[9]

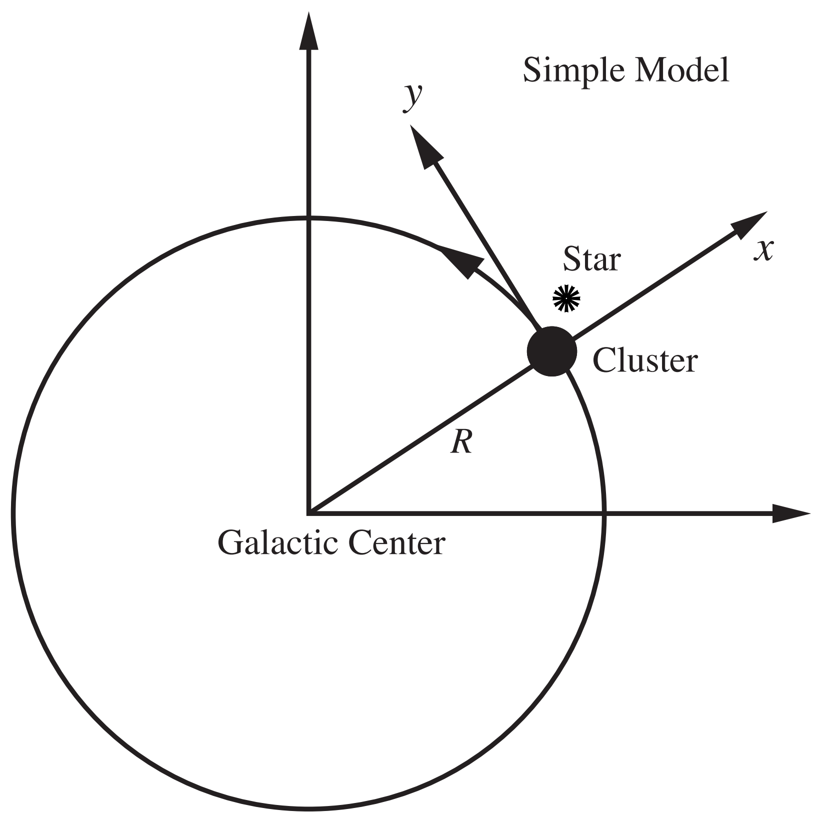


[9] Dellnitz, M., K. Grubits, J. E. Marsden, K. Padberg, and B. Thiere [2005], ***Set-oriented computation of transport rates in 3-degree of freedom systems: the Rydberg atom in crossed fields***, *Regular and Chaotic Dynamics* **10**, 173–192; Gabern, F., W.S. Koon, J.E. Marsden, and S.D. Ross [2005], ***Theory and computation of non-RRKM lifetime distributions and rates in chemical systems with three or more degrees of freedom***, (submitted).

Galactic Sized Tubes

- From ***Shane Ross.*** Tubes are known to govern structure and motion even over galactic scales. The huge tails emanating out of some star clusters in orbit about our galaxy are due to stars slipping into tubes connecting the star cluster with the space outside.





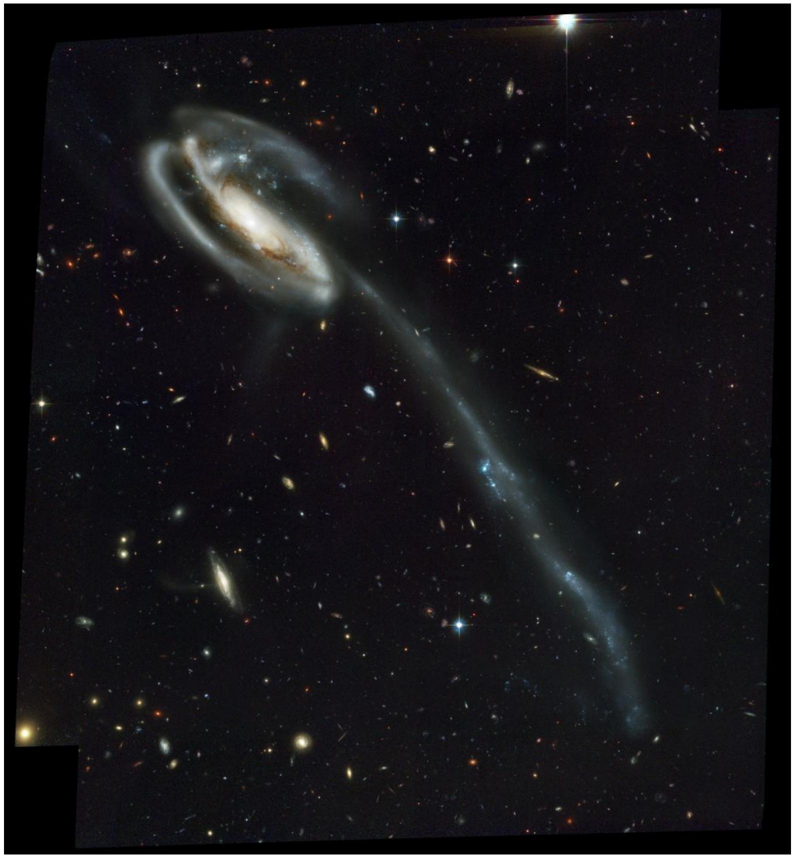
Galactic Sized Tubes

- The figure shows a *million body simulation* by Combes, Leon, and Meylan [1999]. It is believed that this process can eventually lead to the ‘evaporation’ of some star clusters over tens of billions of years. The estimation of this evaporation time scale is possible using a very simple model, in principle similar to the three-body model.

Galactic Sized Tubes

- The figure shows a *million body simulation* by Combes, Leon, and Meylan [1999]. It is believed that this process can eventually lead to the 'evaporation' of some star clusters over tens of billions of years. The estimation of this evaporation time scale is possible using a very simple model, in principle similar to the three-body model.
- Also shown schematically is the star cluster is modelled as a smooth potential (due to the cluster stars) plus the steady tidal field of the galaxy. Stars which are above the energy of the Lagrange points escape via tubes. From the tubes themselves, the half-life, or evaporation time scale can be determined semi-analytically.

Galactic Sized Tubes

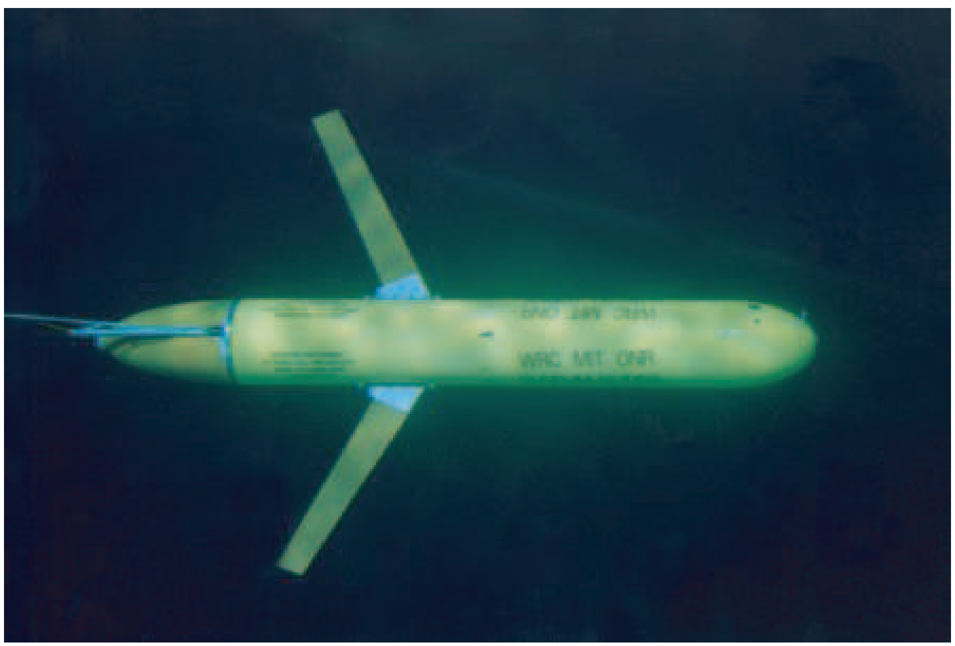


Tadpole Galaxy and its 280 thousand light-year long tail. Presumably a passing smaller galaxy pulled off the stars in the tail.

Photo credit: NASA ACS Science & Engineering Team.

Back to LCS: Optimization

- Monterey Bay studies done in the context of a big AOSN-II experiment done in summer 03 (MBARI) and continuing as ASAP in 06 (Naomi Leonard at Princeton, Steve Ramp at NPS).



Back to LCS: Optimization

- Optimal strategies for gliders—to optimally gather data or to optimize trajectories. Done with NTG.[10]

LCS and Optimization

[10] T. Inanc, S.C. Shadden and J.E. Marsden, [2005], ***Optimal trajectory generation in ocean flows***, *Proceedings of the 2005 American Control Conference*, 674–679.

More Optimization: ESA Darwin study

- Group of spacecraft in a halo orbit about L_2 for the Earth-Sun-spacecraft 3-body system.

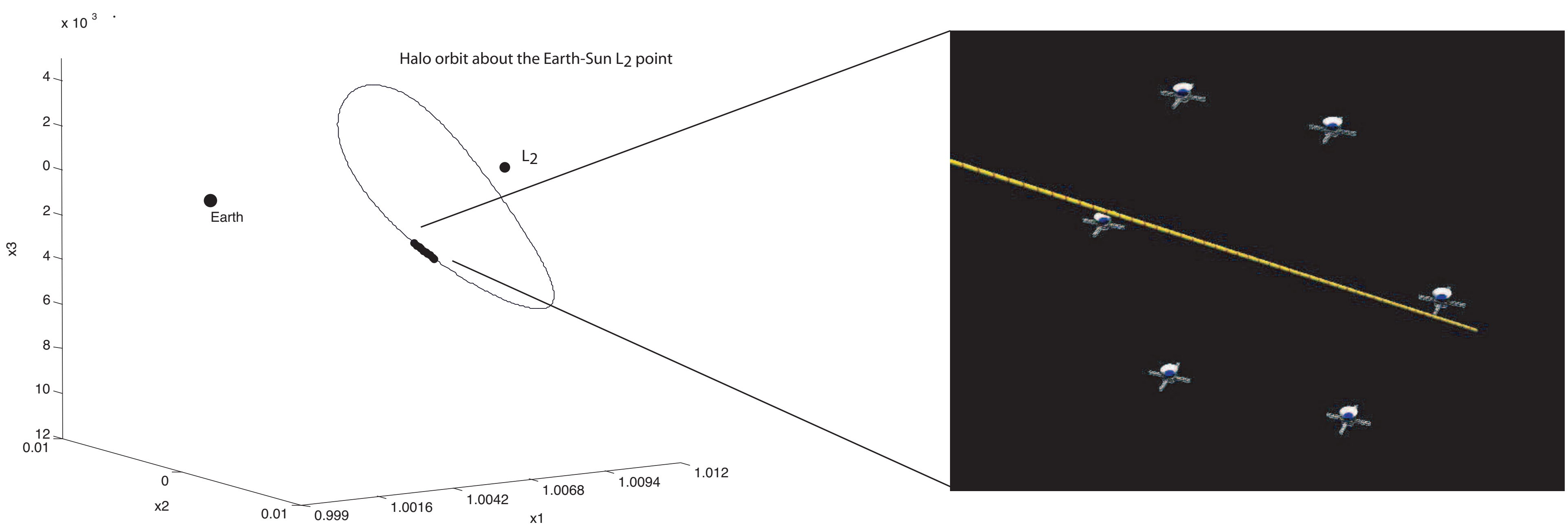
More Optimization: ESA Darwin study

- Group of spacecraft in a halo orbit about L_2 for the Earth-Sun-spacecraft 3-body system.
- **Goal:** Optimally reconfigure the group to achieve a hexagonal configuration, including collision avoidance, e.g., pointing to an interesting distant solar system.[11]

[11] Provided courtesy of Sina Ober-Blöbaum and Oliver Junge.

More Optimization: ESA Darwin study

- Group of spacecraft in a halo orbit about L_2 for the Earth-Sun-spacecraft 3-body system.
- **Goal:** Optimally reconfigure the group to achieve a hexagonal configuration, including collision avoidance, e.g., pointing to an interesting distant solar system.[11]
- DMOC: $N = 10$ time intervals, SQP-method: `E04UEF` (NAG).



[11]Provided courtesy of Sina Ober-Blöbaum and Oliver Junge.

More Optimization: ESA Darwin study

Darwin Movie

DMOC: Discrete Mechanics and Optimal Control

- Use discrete mechanics to discretize the equations.

DMOC: Discrete Mechanics and Optimal Control

- Use discrete mechanics to discretize the equations.
- Optimize a given cost function (such as the control effort) using this with standard SQP methods using the discrete equations of mechanics as constraints.

DMOC: Discrete Mechanics and Optimal Control

- Use discrete mechanics to discretize the equations.
- Optimize a given cost function (such as the control effort) using this with standard SQP methods using the discrete equations of mechanics as constraints.
- Similar examples using, e.g., orbit transfer of Earth-bound satellites using low thrust.

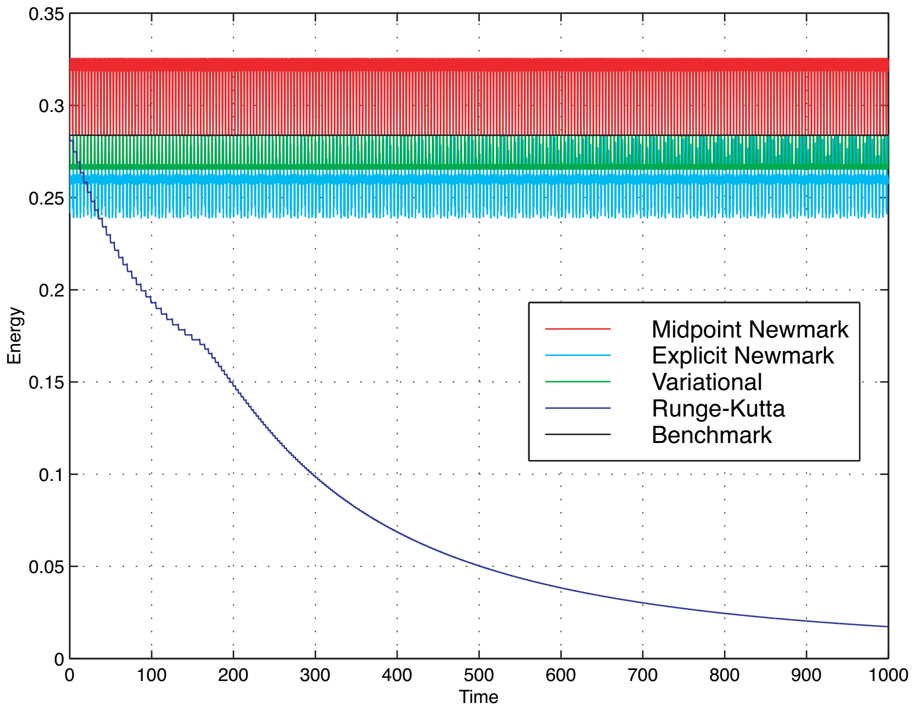
Discrete Mechanics: Example

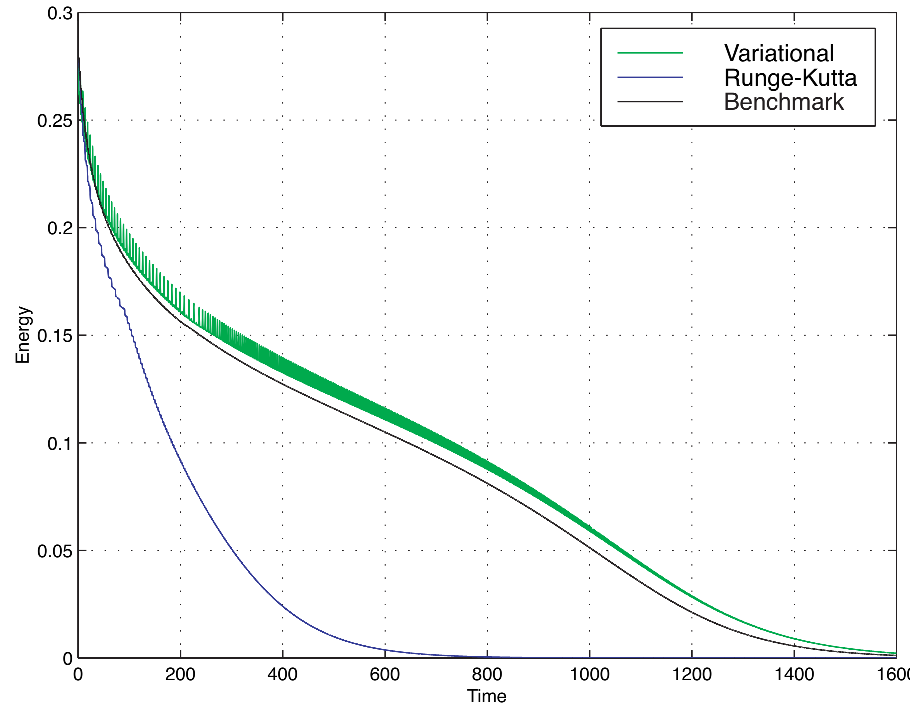
- Particle in $\mathbb{R}^2$ moving in the field of a radially symmetric polynomial potential (left); with small dissipation (right).[12]

[12] Kane, C., J.E. Marsden, M. Ortiz, M. West [2000], ***Variational integrators & the Newmark alg. for conserv. and dissip. mech. systems***, *Int. J. Num. Meth. Eng.* **49**, 1295–1325.

Discrete Mechanics: Example

◦ Particle in $\mathbb{R}^2$ moving in the field of a radially symmetric polynomial potential (left); with small dissipation (right).[12]

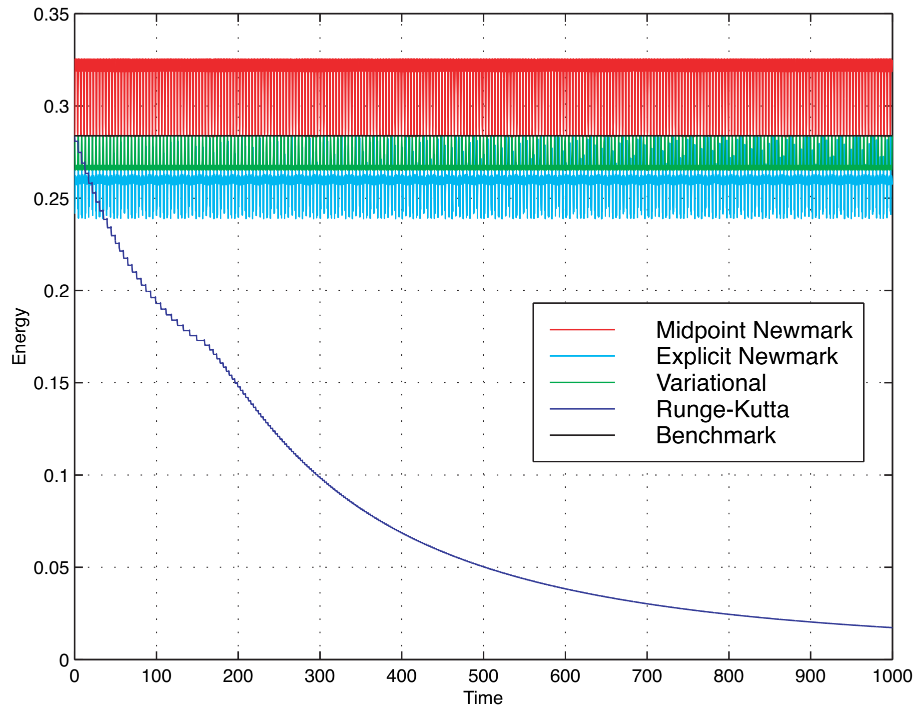


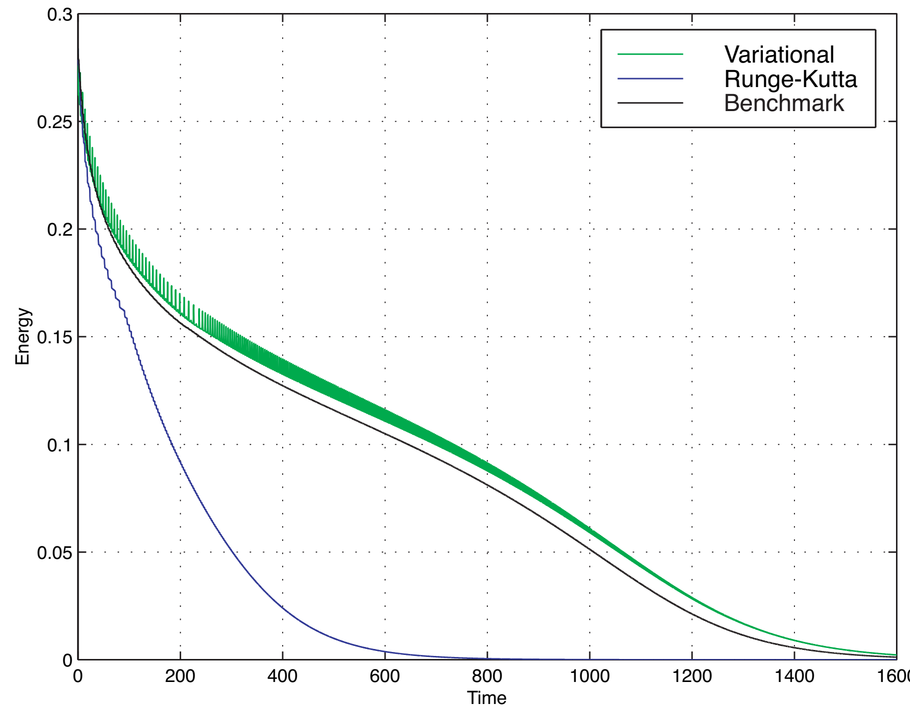


[12] Kane, C., J.E. Marsden, M. Ortiz, M. West [2000], ***Variational integrators & the Newmark alg. for conserv. and dissip. mech. systems***, *Int. J. Num. Meth. Eng.* **49**, 1295–1325.

Discrete Mechanics: Example

- Particle in $\mathbb{R}^2$ moving in the field of a radially symmetric polynomial potential (left); with small dissipation (right).[12]





- ***Good energy behavior in both the conservative and dissipative/controlled cases***; the integrator, in the absence of dissipation is symplectic and angular momentum preserving.

[12] Kane, C., J.E. Marsden, M. Ortiz, M. West [2000], ***Variational integrators & the Newmark alg. for conserv. and dissip. mech. systems***, *Int. J. Num. Meth. Eng.* **49**, 1295–1325.

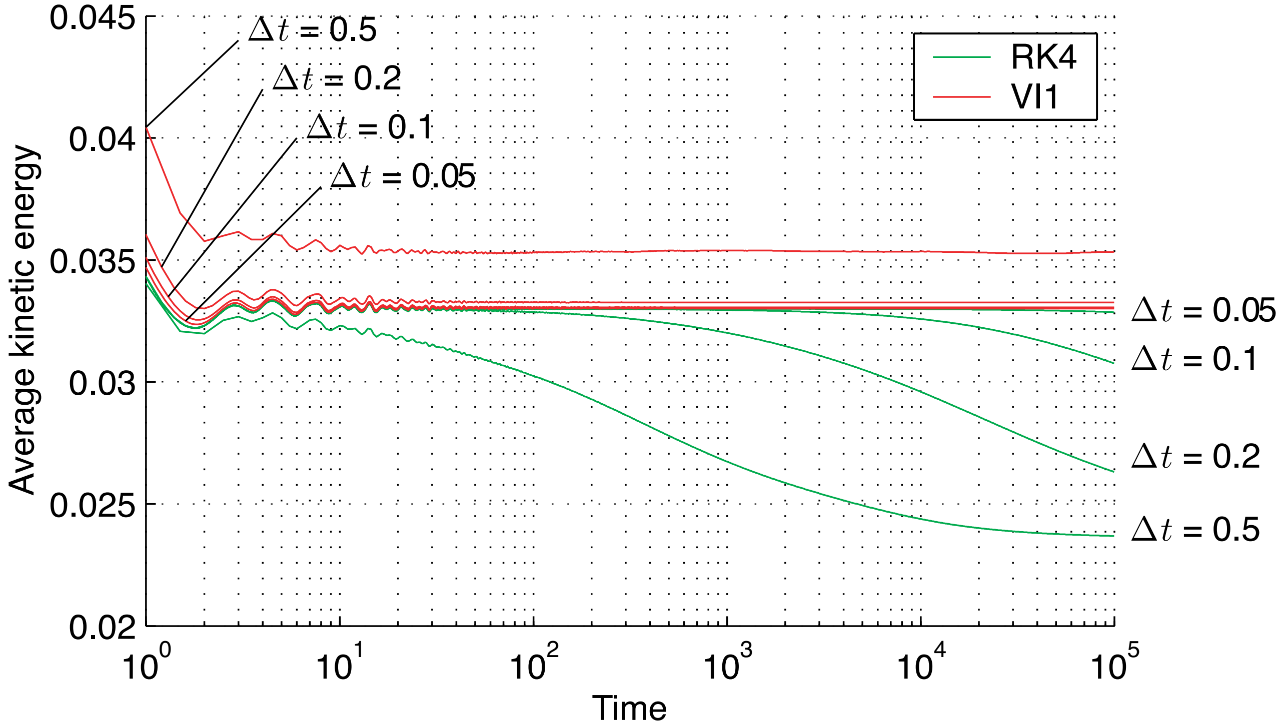
Discrete Mechanics: Example

- Gets key coarse variables right: statistical computations.[13]

[13] Lew, A., J. E. Marsden, M. Ortiz, and M. West [2004], ***Variational time integration for mechanical systems***, *Intern. J. Num. Meth. in Engin.* **60**, 153–212.

Discrete Mechanics: Example

- Gets key coarse variables right: statistical computations.[13]



Excellent performance in the computation of the *"temperature"* (time average of the kinetic energy) of a system of interacting particles.

[13] Lew, A., J. E. Marsden, M. Ortiz, and M. West [2004], ***Variational time integration for mechanical systems***, *Intern. J. Num. Meth. in Engin.* **60**, 153–212.

Why Variational?

- The flexibility of the variational view allows for a natural extension to PDEs, asynchronous computations, etc.
- The framework is not symplectic maps or geometry, but ***multi-symplectic geometry***[a]

[a]Gotay, M., J. Isenberg, J. E. Marsden and R. Montgomery [1997], ***Momentum Maps and the Hamiltonian Structure of Classical Relativistic Field Theories***, http://www.cds.caltech.edu/~marsden/; Marsden, J. E., G. W. Patrick, and S. Shkoller [1998], ***Multisymplectic geometry, variational integrators and nonlinear PDEs***, *Comm. Math.Phys.* **199**, 351–395; Lew, A., J. E. Marsden, M. Ortiz, and M. West [2003], ***Asynchronous variational integrators***, *Archive for Rat. Mech. An.* **167**, 85–146.

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* **(roots: discrete optimal control from the 1960's).**

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* **(roots: discrete optimal control from the 1960's).**
- Performance: **works very well for *both conservative and dissipative or forced* mechanical systems**

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* (roots: discrete optimal control from the 1960's).
- Performance: works very well for *both conservative and dissipative or forced* mechanical systems
- Design: of *higher-order* integrators; easier than finding approximate solutions to the HJ equation (Channe Scovel). E.g. naturally gives SPARK methods.

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* (roots: discrete optimal control from the 1960's).
- Performance: works very well for *both conservative and dissipative or forced* mechanical systems
- Design: of *higher-order* integrators; easier than finding approximate solutions to the HJ equation (Channe Scovel). E.g. naturally gives SPARK methods.
- Flexible and Broad: includes *explicit or implicit* algorithms, extends naturally to include multisymplectic schemes for PDE.

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* (roots: discrete optimal control from the 1960's).
- Performance: works very well for *both conservative and dissipative or forced* mechanical systems
- Design: of *higher-order* integrators; easier than finding approximate solutions to the HJ equation (Channe Scovel). E.g. naturally gives SPARK methods.
- Flexible and Broad: includes *explicit or implicit* algorithms, extends naturally to include multisymplectic schemes for PDE.
- Structure Preserving: *symplectic and momentum* conserving (for the non-forced case)

Theory Highlights

- Basis: *discrete mechanics & Hamilton's Principle* (roots: discrete optimal control from the 1960's).
- Performance: works very well for *both conservative and dissipative or forced* mechanical systems
- Design: of *higher-order* integrators; easier than finding approximate solutions to the HJ equation (Channe Scovel). E.g. naturally gives SPARK methods.
- Flexible and Broad: includes *explicit or implicit* algorithms, extends naturally to include multisymplectic schemes for PDE.
- Structure Preserving: *symplectic and momentum* conserving (for the non-forced case)
- Discrete Reduction: Discrete analogs of symplectic and Poisson reduction theory.

Discrete Mechanics

- **Key Idea:**[14] Approximate the action integral with a quadrature rule—gives a ***discrete Lagrangian***:

$$L_d(q_0, q_1, h) \approx \int_0^h L\big(q(t), \dot{q}(t)\big)\, dt$$

where $q(t)$ is the ***exact solution*** of the Euler–Lagrange equations for L joining q_0 to q_1 over the ***time step interval*** $0 \leq t \leq h$.

[14] See Marsden, J.E. and M. West [2001], ***Discrete variational mechanics and variational integrators***, *Acta Numerica* **10**, 357–514 for a survey of the theory.

Discrete Mechanics

- **Key Idea:**[14] Approximate the action integral with a quadrature rule—gives a ***discrete Lagrangian***:

$$L_d(q_0, q_1, h) \approx \int_0^h L\big(q(t), \dot{q}(t)\big)\, dt$$

where $q(t)$ is the ***exact solution*** of the Euler–Lagrange equations for L joining q_0 to q_1 over the ***time step interval*** $0 \leq t \leq h$.

- Using the exact value and not an approximation would lead to a solution to the ***Hamilton-Jacobi equation (Jacobi, 1840)***

[14] See Marsden, J.E. and M. West [2001], ***Discrete variational mechanics and variational integrators***, *Acta Numerica* **10**, 357–514 for a survey of the theory.

Discrete Hamilton's Principle

- Given a discrete Lagrangian, form the ***action sum:***

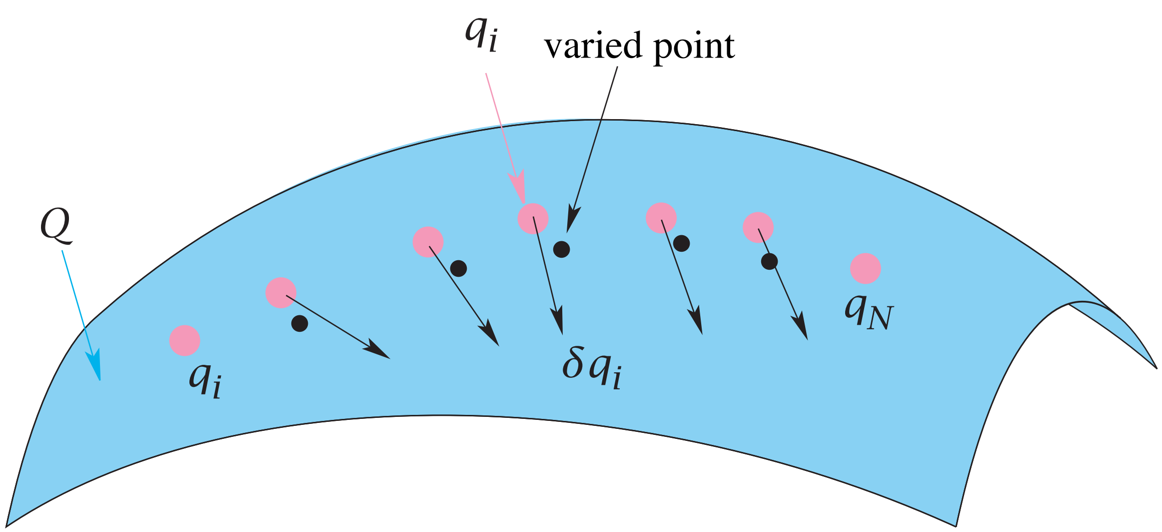
$$S_d = \sum_{k=0}^{N-1} L_d\left(q_k, q_{k+1}, h_k\right)$$

Discrete Hamilton's Principle

- Given a discrete Lagrangian, form the ***action sum:***

$$S_d = \sum_{k=0}^{N-1} L_d\left(q_k, q_{k+1}, h_k\right)$$

- **Discrete variational (Hamilton) principle**: Extremize S_d with fixed end points, q_0 and q_N



Discrete variational principle

Discrete Hamilton's Principle

- vary the point q_i; the only terms in the sum that are affected are $L_d\left(q_{i-1}, q_i, h_{i-1}\right) + L_d\left(q_i, q_{i+1}, h_i\right)$; this gives the ***DEL***, that is, the ***Discrete Euler–Lagrange*** equations:

$$D_2 L_d\left(q_{i-1}, q_i, h_{i-1}\right) + D_1 L_d\left(q_i, q_{i+1}, h_i\right) = 0$$

Discrete Hamilton's Principle

- vary the point q_i; the only terms in the sum that are affected are $L_d(q_{i-1}, q_i, h_{i-1}) + L_d(q_i, q_{i+1}, h_i)$; this gives the ***DEL***, that is, the ***Discrete Euler–Lagrange*** equations:

$$D_2 L_d(q_{i-1}, q_i, h_{i-1}) + D_1 L_d(q_i, q_{i+1}, h_i) = 0$$

- This defines the DEL ***algorithm***:

$$(q_{i-1}, q_i) \mapsto (q_i, q_{i+1})$$

Simple Example

- Let M be a positive definite symmetric $n \times n$ matrix and $V : \mathbb{R}^n \to \mathbb{R}$ be a given potential. Lagrangian: $L(q, \dot{q}) = \frac{1}{2}\dot{q}^T M\dot{q} - V(q)$.

Simple Example

- Let M be a positive definite symmetric $n \times n$ matrix and $V : \mathbb{R}^n \to \mathbb{R}$ be a given potential. Lagrangian: $L(q, \dot{q}) = \frac{1}{2}\dot{q}^T M \dot{q} - V(q)$.
- Choose the discrete Lagrangian to be

$$L_d(q_0, q_1, h) = h\left[\frac{1}{2}\left(\frac{q_1 - q_0}{h}\right)^T M \left(\frac{q_1 - q_0}{h}\right) - V(q_0)\right]$$

Simple Example

- Let M be a positive definite symmetric $n \times n$ matrix and $V : \mathbb{R}^n \to \mathbb{R}$ be a given potential. Lagrangian: $L(q, \dot{q}) = \frac{1}{2}\dot{q}^T M \dot{q} - V(q)$.
- Choose the discrete Lagrangian to be

$$L_d(q_0, q_1, h) = h\left[\frac{1}{2}\left(\frac{q_1 - q_0}{h}\right)^T M \left(\frac{q_1 - q_0}{h}\right) - V(q_0)\right]$$

- Use the "rectangle rule" on the action integral and the approximation $\dot{q} \approx (q_1 - q_0)/h$.

Simple Example

- Let M be a positive definite symmetric $n \times n$ matrix and $V : \mathbb{R}^n \to \mathbb{R}$ be a given potential. Lagrangian: $L(q, \dot{q}) = \frac{1}{2}\dot{q}^T M \dot{q} - V(q)$.
- Choose the discrete Lagrangian to be

$$L_d(q_0, q_1, h) = h\left[\frac{1}{2}\left(\frac{q_1 - q_0}{h}\right)^T M \left(\frac{q_1 - q_0}{h}\right) - V(q_0)\right]$$

- Use the "rectangle rule" on the action integral and the approximation $\dot{q} \approx (q_1 - q_0)/h$.
- DEL equations are a discretization of Newton's equations:

$$M\left(\frac{q_{k+1} - 2q_k + q_{k-1}}{h^2}\right) = -\nabla V(q_k)$$

Simple Example

- Let M be a positive definite symmetric $n \times n$ matrix and $V : \mathbb{R}^n \to \mathbb{R}$ be a given potential. Lagrangian: $L(q, \dot{q}) = \frac{1}{2}\dot{q}^T M \dot{q} - V(q)$.
- Choose the discrete Lagrangian to be

$$L_d(q_0, q_1, h) = h\left[\frac{1}{2}\left(\frac{q_1 - q_0}{h}\right)^T M \left(\frac{q_1 - q_0}{h}\right) - V(q_0)\right]$$

- Use the "rectangle rule" on the action integral and the approximation $\dot{q} \approx (q_1 - q_0)/h$.
- DEL equations are a discretization of Newton's equations:

$$M\left(\frac{q_{k+1} - 2q_k + q_{k-1}}{h^2}\right) = -\nabla V(q_k)$$

- **Many Other Examples:** Midpoint rule, Newmark algorithms, symplectic partitioned Runge-Kutta algorithms, Verlet, etc etc.

Forced or Controlled Case

- Dealing with external forces is important for us as we will be doing optimal control.

Forced or Controlled Case

- Dealing with external forces is important for us as we will be doing optimal control.
- For this (and the dissipative case) one discretizes the Lagrange-d'Alembert principle:

$$\delta \int L(q, \dot{q})\, dt + \int f \cdot \delta q\, dt = 0$$

Forced or Controlled Case

- Dealing with external forces is important for us as we will be doing optimal control.
- For this (and the dissipative case) one discretizes the Lagrange-d'Alembert principle:

$$\delta \int L(q, \dot{q})\, dt + \int f \cdot \delta q \, dt = 0$$

- In addition to approximating the action with the discrete Lagrangian, we approximate the virtual work:

$$f_k^- \cdot \delta q_k + f_k^+ \cdot \delta q_{k+1} \approx \int_{kh}^{(k+1)h} f(t) \cdot \delta q(t)\, dt,$$

where $f_k^-, f_k^+ \in T^*Q$ are the *left* and *right discrete forces*.

Optimization

- **Idea:** merge discrete mechanics and optimal control

Optimization

- **Idea:** merge discrete mechanics and optimal control
- Room for lots of new ideas too; eg, use discrete multi-symplectic mechanics, pattern search ideas, SMF (surrogate management framework; John Dennis), etc.

Optimization

- **Idea:** merge discrete mechanics and optimal control
- Room for lots of new ideas too; eg, use discrete multi-symplectic mechanics, pattern search ideas, SMF (surrogate management framework; John Dennis), etc.
- Many ***existing methodologies***: NTG (Milam and Murray)—the one we used for optimization in a dynamic ocean environment and COOP (Petzold) that was applied to optimal orbit insertion for Genesis[15]

[15] Serban, R., W. S. Koon, M. Lo, J. E. Marsden, L. R. Petzold, S. D. Ross and R. S. Wilson [2002], ***Halo orbit mission correction maneuvers using optimal control,*** *Automatica* **38**, 571–583.

Optimization

- **Idea:** merge discrete mechanics and optimal control
- Room for lots of new ideas too; eg, use discrete multi-symplectic mechanics, pattern search ideas, SMF (surrogate management framework; John Dennis), etc.
- Many ***existing methodologies***: NTG (Milam and Murray)—the one we used for optimization in a dynamic ocean environment and COOP (Petzold) that was applied to optimal orbit insertion for Genesis[15]
- DMOC ***respects the energy budget*** well and ***needs remarkably few division points***. That is, can take large time steps, Δt.

[15] Serban, R., W. S. Koon, M. Lo, J. E. Marsden, L. R. Petzold, S. D. Ross and R. S. Wilson [2002], *Halo orbit mission correction maneuvers using optimal control, Automatica* **38**, 571–583.

The Lagrange Problem

- Minimize a cost function

$$J(x,u) = \int_{t_0}^{t_f} C(x(t), u(t))\, dt$$

subject to dynamical (plus other) constraints:

$$\dot{x}(t) = f(x(t), u(t))$$

subject to initial contitions $x(t_0) = x^0$ and final conditions $x(t_f) = x^f$.

The Lagrange Problem

- Minimize a cost function

$$J(x,u) = \int_{t_0}^{t_f} C(x(t),u(t))\,dt$$

subject to dynamical (plus other) constraints:

$$\dot{x}(t) = f(x(t),u(t))$$

subject to initial contitions $x(t_0) = x^0$ and final conditions $x(t_f) = x^f$.

- **Theory.** Centered on Pontryagin maximum principle, receding horizon control,....

The Lagrange Problem

- Minimize a cost function

$$J(x,u) = \int_{t_0}^{t_f} C(x(t), u(t))\, dt$$

  subject to dynamical (plus other) constraints:

$$\dot{x}(t) = f(x(t), u(t))$$

  subject to initial contitions $x(t_0) = x^0$ and final conditions $x(t_f) = x^f$.

- **Theory.** Centered on Pontryagin maximum principle, receding horizon control,....
- **Practice.** Brute force optimization software.

MOC: Mechanical Optimal Control

- **Mechanical systems**: x is a point in the velocity phase space of the system; $x = (q, \dot{q}) \in TQ$, where Q is the configuration manifold.

MOC: Mechanical Optimal Control

- **Mechanical systems**: x is a point in the velocity phase space of the system; $x = (q, \dot{q}) \in TQ$, where Q is the configuration manifold.
- Most current software packages use ***SQP (sequential quadratic programing)*** to do the basic optimization.

MOC: Mechanical Optimal Control

- **Mechanical systems**: x is a point in the velocity phase space of the system; $x = (q, \dot{q}) \in TQ$, where Q is the configuration manifold.
- Most current software packages use ***SQP (sequential quadratic programing)*** to do the basic optimization.
- For the Lagrange problem, one has to also deal with the constraints of the equations of motion. How these are handled is one of the key differences between software packages.

MOC: Mechanical Optimal Control

- ***Mechanical Case:*** Equations are of Euler–Lagrange type with control forces, which are determined from the "variational" principle of ***Lagrange-d'Alembert type.***

$$\delta \int_{t_0}^{t_f} L(q(t), \dot{q}(t))\, dt + \int_{t_0}^{t_f} u(t)\delta q(t)\, dt = 0$$

for a given Lagrangian $L : TQ \to \mathbb{R}$.

MOC: Mechanical Optimal Control

- ***Mechanical Case:*** Equations are of Euler–Lagrange type with control forces, which are determined from the "variational" principle of ***Lagrange-d'Alembert type.***

$$\delta \int_{t_0}^{t_f} L(q(t), \dot{q}(t))\, dt + \int_{t_0}^{t_f} u(t)\delta q(t)\, dt = 0$$

  for a given Lagrangian $L : TQ \to \mathbb{R}$.

- Our strategy[16]: make use of direct SQP methods for dealing with optimization—but ***replace the equations of motion by their discrete variational counterpart.***

[16] Junge, O., J. E. Marsden, and S. Ober-Blöbaum [2005], ***Discrete mechanics and optimal control***, *2005 IFAC Proceedings.*

DMOC Summary

□ Procedure.

- ◦ Choose a discrete L_d using a chosen quadrature method

DMOC Summary

Procedure.

- Choose a discrete L_d using a chosen quadrature method
- Discretize the virtual work term in the Lagrange-d'Alembert principle, giving left and right discrete forces u^- and u^+

DMOC Summary

Procedure.

- Choose a discrete L_d using a chosen quadrature method
- Discretize the virtual work term in the Lagrange-d'Alembert principle, giving left and right discrete forces u^- and u^+
- Result: ***purely algebraic constraints***

$$D_1 L_d(q_k, q_{k+1}) + D_2(q_{k-1}, q_k) + u^-(u_k, u_{k+1}) + u^+(u_{k-1}, u_k) = 0$$

DMOC Summary

□ Procedure.

- Choose a discrete L_d using a chosen quadrature method
- Discretize the virtual work term in the Lagrange-d'Alembert principle, giving left and right discrete forces u^- and u^+
- Result: ***purely algebraic constraints***

$$D_1 L_d(q_k, q_{k+1}) + D_2(q_{k-1}, q_k) + u^-(u_k, u_{k+1}) + u^+(u_{k-1}, u_k) = 0$$

- Discretize the cost functional

$$J_d(q_d, u_d) = \sum_{k=0}^{N-1} C_d(q_k, q_{k+1}, u_k, u_{k+1})$$

DMOC Summary

Procedure.

- Choose a discrete L_d using a chosen quadrature method
- Discretize the virtual work term in the Lagrange-d'Alembert principle, giving left and right discrete forces u^- and u^+
- Result: ***purely algebraic constraints***

$$D_1 L_d(q_k, q_{k+1}) + D_2(q_{k-1}, q_k) + u^-(u_k, u_{k+1}) + u^+(u_{k-1}, u_k) = 0$$

- Discretize the cost functional

$$J_d(q_d, u_d) = \sum_{k=0}^{N-1} C_d(q_k, q_{k+1}, u_k, u_{k+1})$$

- Hand this off to standard and powerful SQP packages, enforcing the initial and final conditions and any other constraints.

Hovercraft Example

- A group of hovercraft are asked to move from a given starting position to a hexagonal final formation shown in an optimal way. (The hovercraft themselves need to decide who goes where).

Hovercraft Example

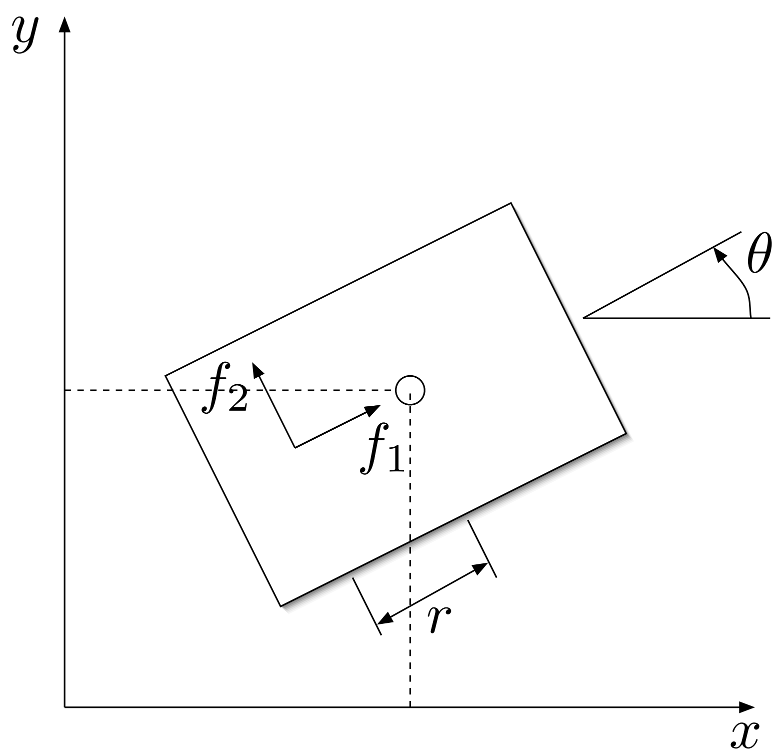
- A group of hovercraft are asked to move from a given starting position to a hexagonal final formation shown in an optimal way. (The hovercraft themselves need to decide who goes where).
- **Model:** mechanical systems with 3 degrees of freedom (position (x, y), heading angle θ), so $Q = \mathbb{R}^2 \times S^1$

Hovercraft Example

- A group of hovercraft are asked to move from a given starting position to a hexagonal final formation shown in an optimal way. (The hovercraft themselves need to decide who goes where).
- **Model:** mechanical systems with 3 degrees of freedom (position (x, y), heading angle θ), so $Q = \mathbb{R}^2 \times S^1$
- **Two actuation forces**—one along the axis of the hovercraft (forward acceleration), and a perpendicular force towards the rear of the hovercraft. not through the center of mass of the hovercraft (sideways slip and steering).

Hovercraft Example

- The system is ***underactuated***. The Lagrangian for the system is the standard kinetic energy of the hovercraft and the equations of motion are the standard Euler–Lagrange equations with forcing.



Hovercraft Example

- System is ***underactuated***, but is ***configuration controllable***—each point in Q can be reached by applying suitably chosen forces $f_1(t)$ and $f_2(t)$.

Hovercraft Example

- System is ***underactuated***, but is ***configuration controllable***—each point in Q can be reached by applying suitably chosen forces $f_1(t)$ and $f_2(t)$.
- The Lagrangian = kinetic energy:

$$L(q, \dot{q}) = \frac{1}{2}(m\dot{x}^2 + m\dot{y}^2 + J\dot{\theta}^2),$$

where $q = (x, y, \theta)$, m is the mass of the hovercraft and J its moment of inertia. The forces acting in x-, y- and θ- direction resulting from f_1 and f_2 are

$$f(t) = \begin{pmatrix} \cos\theta(t) f_1(t) - \sin\theta(t) f_2(t) \\ \sin\theta(t) f_1(t) + \cos\theta(t) f_2(t) \\ -r f_2(t) \end{pmatrix}.$$

Hovercraft Example

- Forced discrete Euler-Lagrange equations

$$\frac{1}{h}M\left(-q_{k-1}+2q_k-q_{k+1}\right)+\frac{h}{2}\left(\frac{f_{k-1}+f_k}{2}+\frac{f_k+f_{k+1}}{2}\right)=0,$$

$$k=1,\ldots,N-1,\text{ where } M=\begin{pmatrix} m & 0 & 0\\ 0 & m & 0\\ 0 & 0 & J\end{pmatrix}.$$

- Let $q_i=(x_i,y_i,\theta_i)$ the configuration of the i-th hovercraft and by $f_i=(f_{i1},f_{i2})$ the corresponding forces.

Hovercraft Example

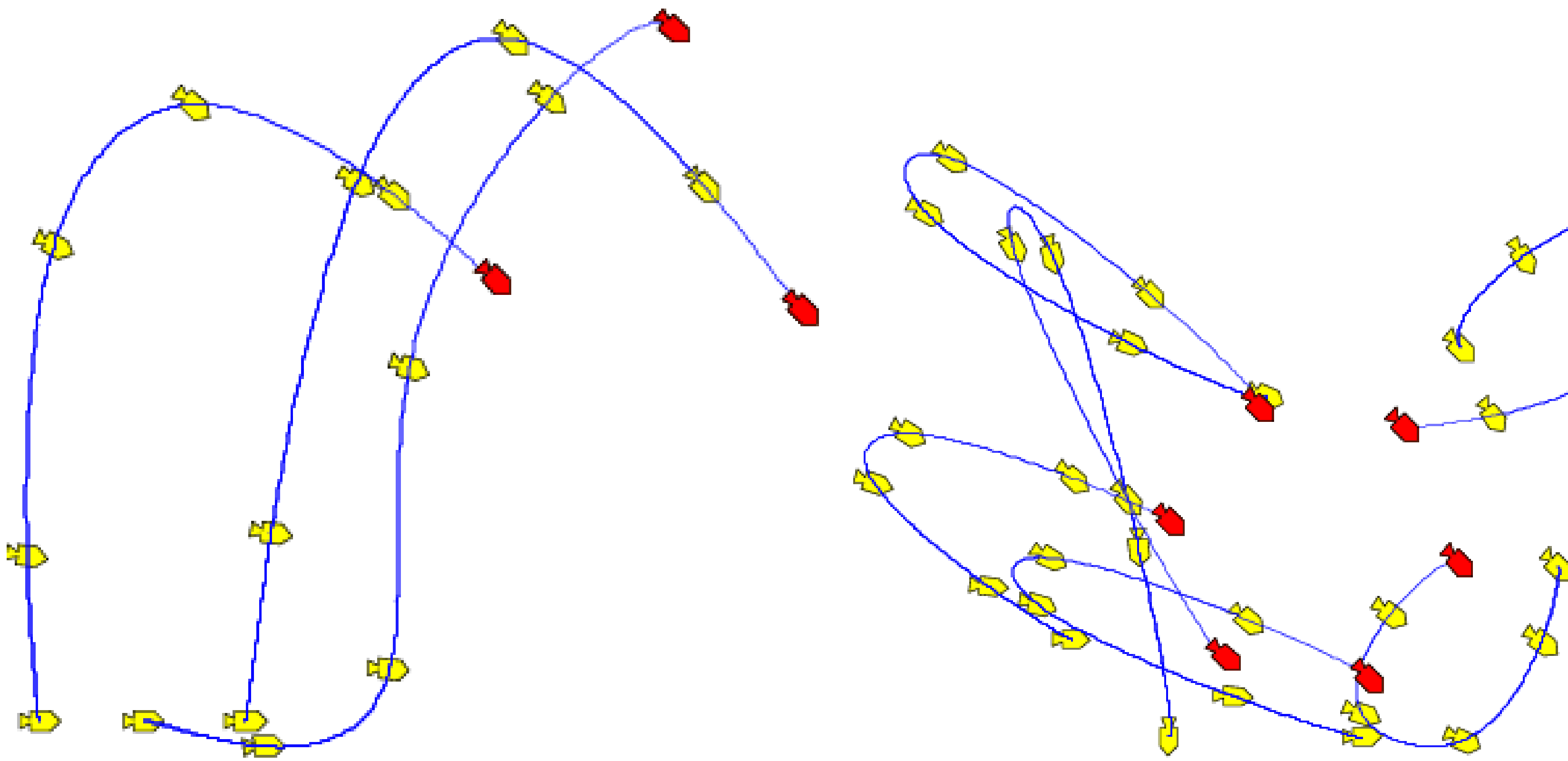
- **Goal**: minimize the control effort needed to attain the final formation.

Hovercraft Example

- **Goal**: minimize the control effort needed to attain the final formation.
- **Sample Cost Function:** Add the costs for each hovercraft

$$J(q_i, f_i) = \int_0^1 f_{i1}^2(t) + f_{i2}^2(t)\, dt,$$

Hovercraft Example



- **Left:** Optimal rearrangement of a group of three hovercraft from an initial configuration along a line into a triangle
  Right: Optimal rearrangement of a group of six hovercraft from a random initial configuration into a hexagon.

Hovercraft Example

Hovercraft

Locomotion Optimization

- Articulated bodies in fluids[17]

[17] Kanso, E., J. E. Marsden, C. W. Rowley, and J. Melli-Huber [2004], ***Locomotion of articulated bodies in a perfect fluid***, *J. Nonlinear Science (to appear)*. We owe a lot to Scott Kelly and Jim Radford !

Locomotion Optimization

- Articulated bodies in fluids[17]
- Coupled rigid bodies (simulating and elastic swimming fish) interacting dynamically with potential flow.

[17] Kanso, E., J. E. Marsden, C. W. Rowley, and J. Melli-Huber [2004], ***Locomotion of articulated bodies in a perfect fluid***, *J. Nonlinear Science (to appear)*. We owe a lot to Scott Kelly and Jim Radford !

Locomotion Optimization

- Articulated bodies in fluids[17]
- Coupled rigid bodies (simulating and elastic swimming fish) interacting dynamically with potential flow.
- Symplectic reduction theory from geometric mechanics (cotangent bundle reduction theorem) proves useful; one uses this to get rid of the fluid particle relabeling symmetry (which gives Kelvin's theorem).

[17] Kanso, E., J. E. Marsden, C. W. Rowley, and J. Melli-Huber [2004], ***Locomotion of articulated bodies in a perfect fluid***, *J. Nonlinear Science (to appear)*. We owe a lot to Scott Kelly and Jim Radford !

Locomotion Optimization

- Attractive feature of DMOC: not hard to implement
- This problem: ***find optimal controls that achieve a given forward movement with the least amount of expended energy***. (The underlying theory is related to the falling cat theorem).[a]

[a]Kanso, E. and J. E. Marsden [2005], ***Optimal motion of an articulated body in a perfect fluid***, *Proc. CDC (submitted)*. The optimal flapper is due to Shane Ross.

More Future Directions

- Use GAIO, Perron Frobenius eigenfunctions, and coarse-fine techniques to separate local from global minima.

More Future Directions

- Use GAIO, Perron Frobenius eigenfunctions, and coarse-fine techniques to separate local from global minima.
- Extend DMOC to the AVI context and use hierarchical and network optimization (primal-dual ideas).

More Future Directions

- Use GAIO, Perron Frobenius eigenfunctions, and coarse-fine techniques to separate local from global minima.
- Extend DMOC to the AVI context and use hierarchical and network optimization (primal-dual ideas).
- Further work on optimization and collision avoidance using collision potentials, gyroscopic controls, and self organized patterns.[18]

[18] Chang, D., S. Shadden, J. E. Marsden, and R. Olfati-Saber [2003], ***Collision avoidance for multiple agent systems***, *Proc. CDC* **42**, 539–543.

More Future Directions

- Use GAIO, Perron Frobenius eigenfunctions, and coarse-fine techniques to separate local from global minima.
- Extend DMOC to the AVI context and use hierarchical and network optimization (primal-dual ideas).
- Further work on optimization and collision avoidance using collision potentials, gyroscopic controls, and self organized patterns.[18]
- Discrete geometry (Whitney forms) for fluids and solid mechanics

[18] Chang, D., S. Shadden, J. E. Marsden, and R. Olfati-Saber [2003], ***Collision avoidance for multiple agent systems***, *Proc. CDC* **42**, 539–543.

The End

Typesetting Software: TEX, *Textures*, LATEX, `hyperref`, `texpower`, Adobe Acrobat 4.05
Graphics Software: Adobe Illustrator 11.0
LATEX Slide Macro Packages: Wendy McKay, Ross Moore