# Efficient Guidance of Underpowered Vehicles in Time-Varying Flow Fields

Blane Rhoads

Ph.D. Defense

Wednesday, October 30th, 2013

# Motivation: underwater gliders

Pro: range, duration, stealth
Con: 0.9-1.4 kph

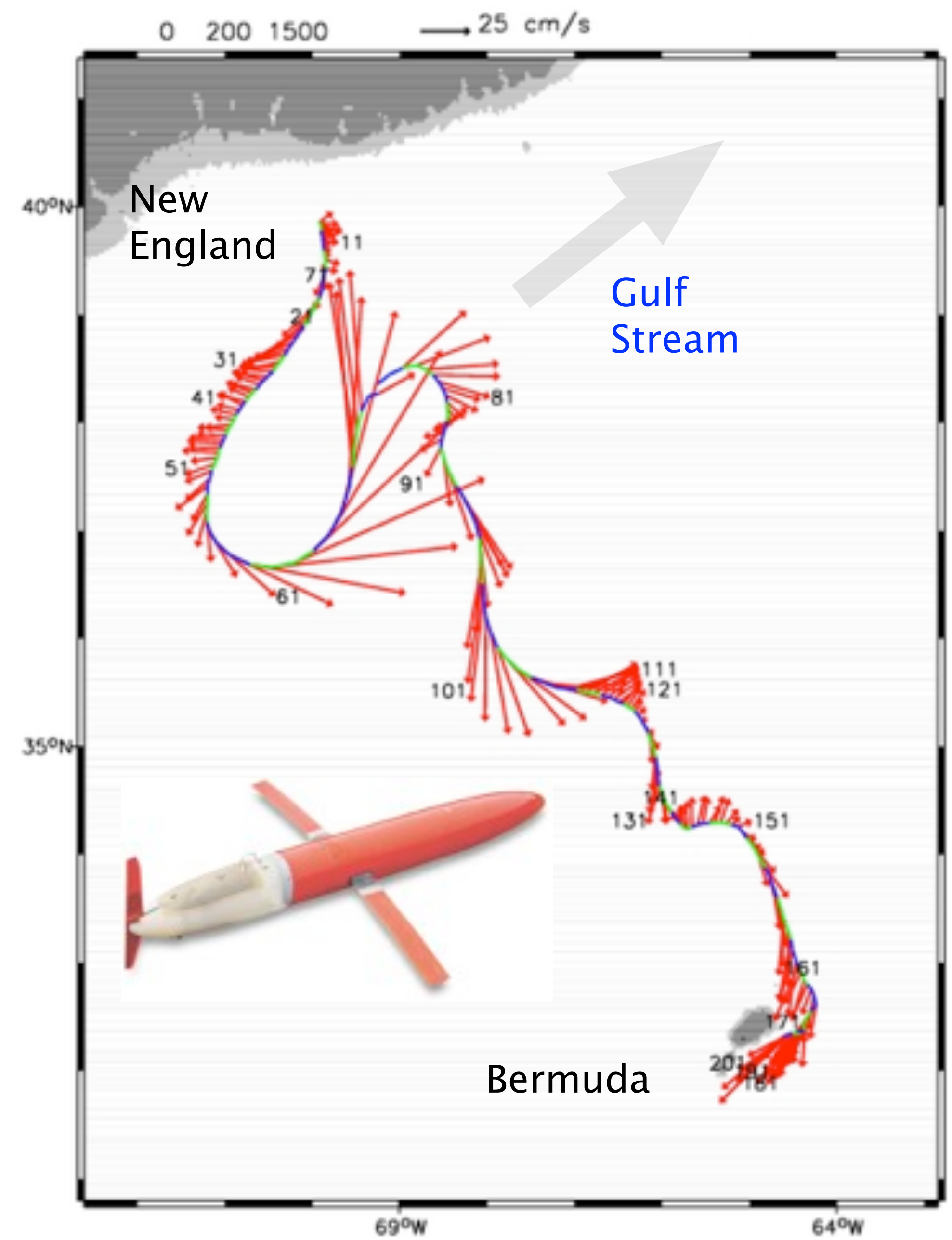
$$\dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t) + \mathbf{u}$$

$\mathbf{v}$: 2D, time-varying flow field

$\mathbf{u}$: Velocity relative to flow

"… a methodology to exploit available estimates of the flow field is of significant interest."

[Leonard, et al. 2007]




Actual 50-day trajectory (blue) and currents (red) [spray.ucsd.edu]

# Overview

- Minimum time
  - Forward Lagrangian
  - Backward Lagrangian (briefly)
- Minimum energy
  - Backward semi-Lagrangian
- Flow map
  - Alternate perspective on minimum energy control
  - Simple greedy control

# Overview

- Minimum time
  - Forward Lagrangian
  - Backward Lagrangian (briefly)
- Minimum energy
  - Backward semi-Lagrangian
- Flow map
  - Alternate perspective on minimum energy control
  - Simple greedy control

# Minimum time control

$$\min_{t_f,\mathbf{u}} \int_{t_0}^{t_f} (1)dt$$ subject to:

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

2D, time-varying flow field (↑ $\mathbf{v}$)

Velocity relative to flow (↑ $\mathbf{u}$)

$$\mathbf{x}(t_0) = \mathbf{x}_0$$
$$\mathbf{x}(t_f) = \mathbf{x}_f$$
$$t_f \in [t_0, t_0 + T_{\max}]$$
$$|\mathbf{u}| \leq s$$

*Time-to-go T* solves Hamilton Jacobi Bellman (HJB) PDE:

$$\frac{\partial T}{\partial t} + \min_{\mathbf{u}} \{(\mathbf{v}+\mathbf{u}) \cdot \nabla T + 1\} = 0 \qquad T(\mathbf{x}_f, t) = 0$$

Feedback control law:

$$\mathbf{u} = -s\frac{\nabla T}{|\nabla T|}$$

# Why is this problem difficult?

- Time-dependence in **v**
  ... a third dimension
- Spatial complexity in **v**
  ... many locally optimal trajectories
  ..."***shocks***"
  = discontinuities in $\nabla T$ and thus in **u**
  (or "*cusps*" in $T$ ).
- |**v**| > *s* (i.e. flow is "strong" or vehicle is "underpowered")
  ... lack of local *controllability/reachability*
  ..."***discontinuities***"
  = discontinuities in $T$ itself.

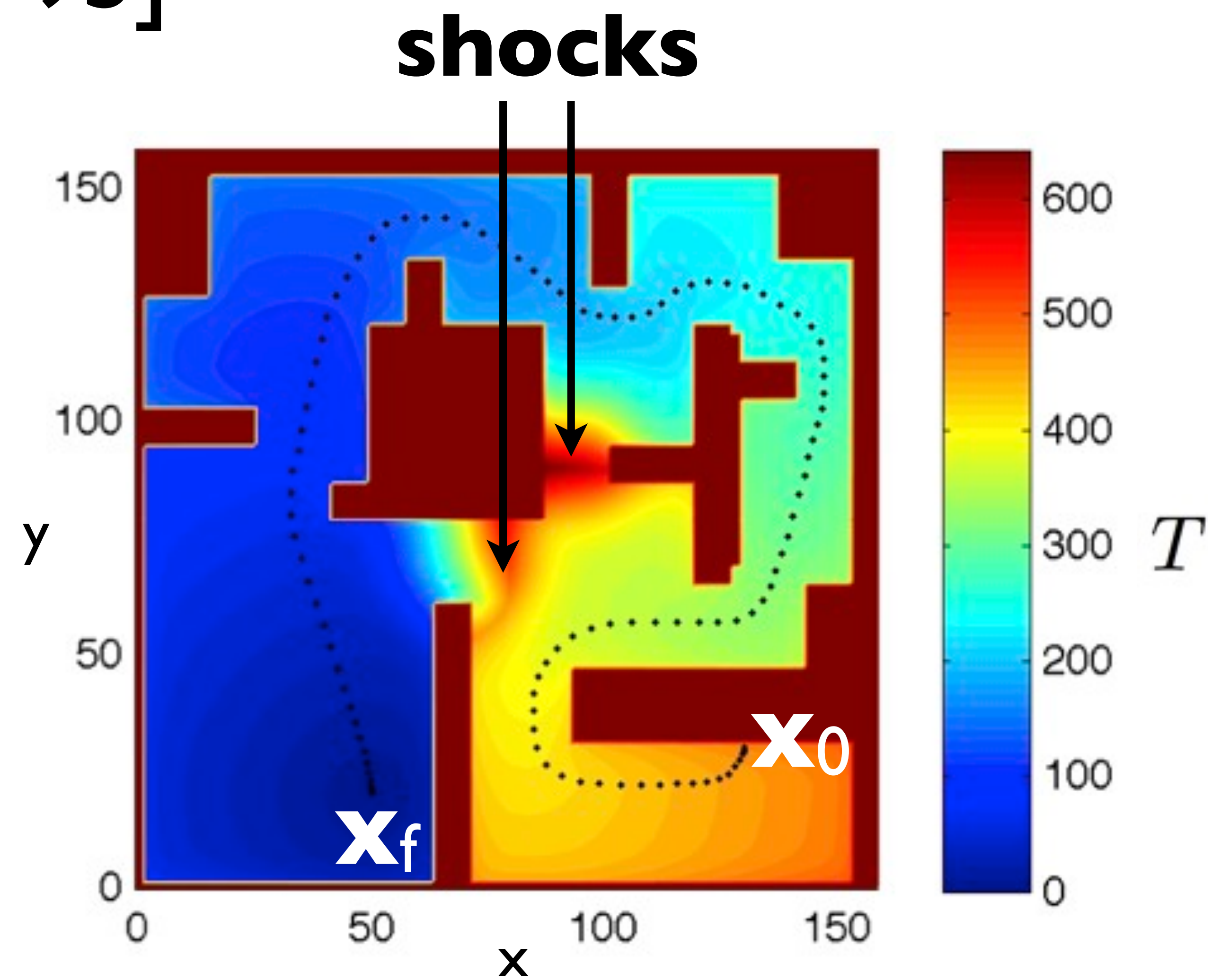
# Benchmark for **v**(**x**,t) = 0: Fast Marching (FM) method [Sethian '96, Tsitsiklis '95]

HJB = Eikonal:

$$s\,|\nabla T| = 1$$

Boundary condition:

$$T(\mathbf{x}_f) = 0$$




*Front tracking* method. Computes $T$ one grid point at a time, in order.

# Benchmark for **v**(**x**,t) = **v**(**x**): Ordered Upwind Methods (OUMs) [Sethian & Vladmirsky ’01]

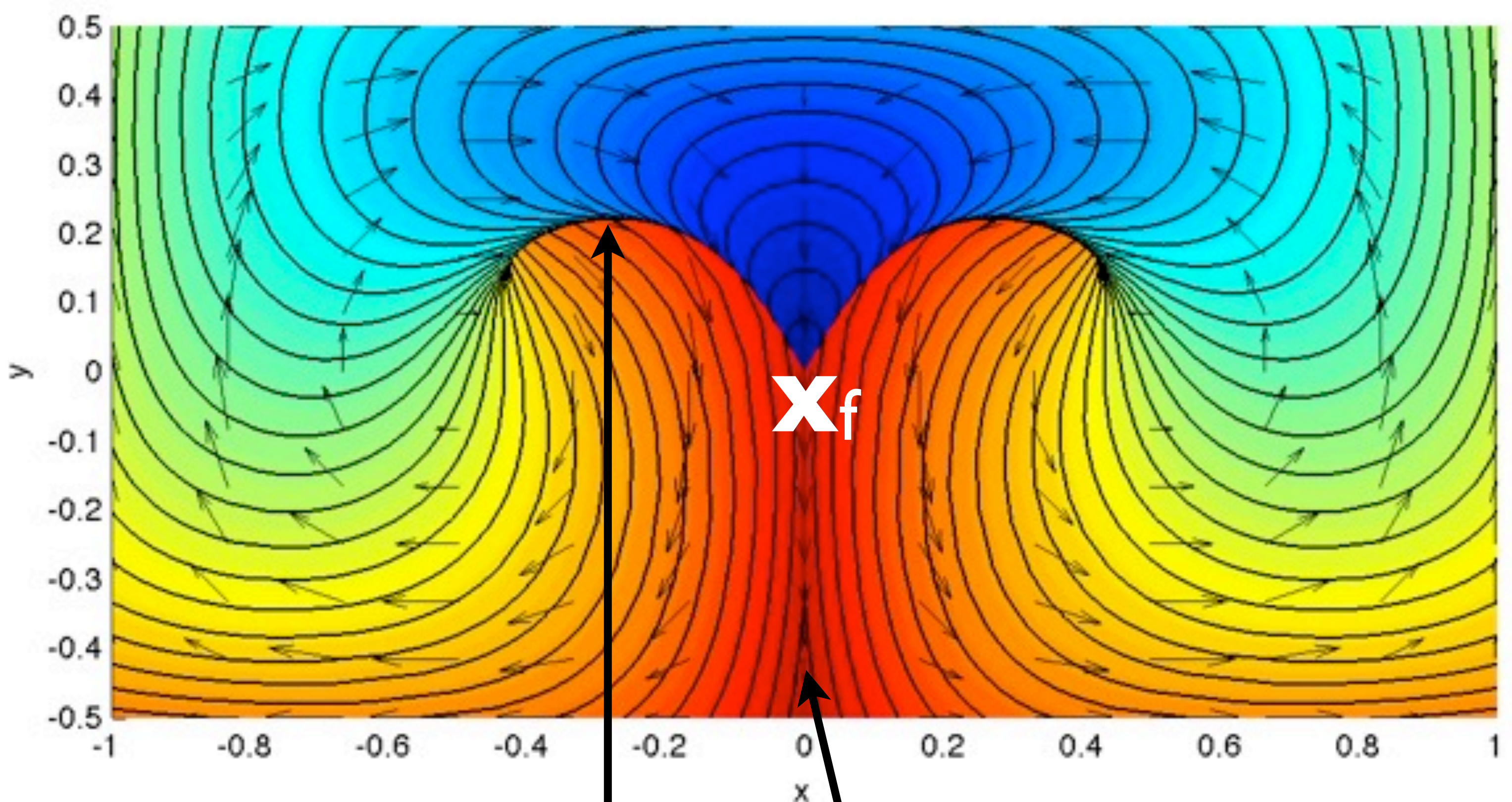
Generalization of FM method to “anisotropic” (direction-dependent) front speeds.

HJB:

$$\underbrace{\left( s - \mathbf{v}(\mathbf{x}) \cdot \frac{\nabla T}{|\nabla T|} \right)}_{\text{speed of front}} |\nabla T| = 1$$

Limitation: Can’t handle |**v**| > s.

# Alternative (for $|\mathbf{v}| > s$): Lagrangian ("marker particle") front tracking method



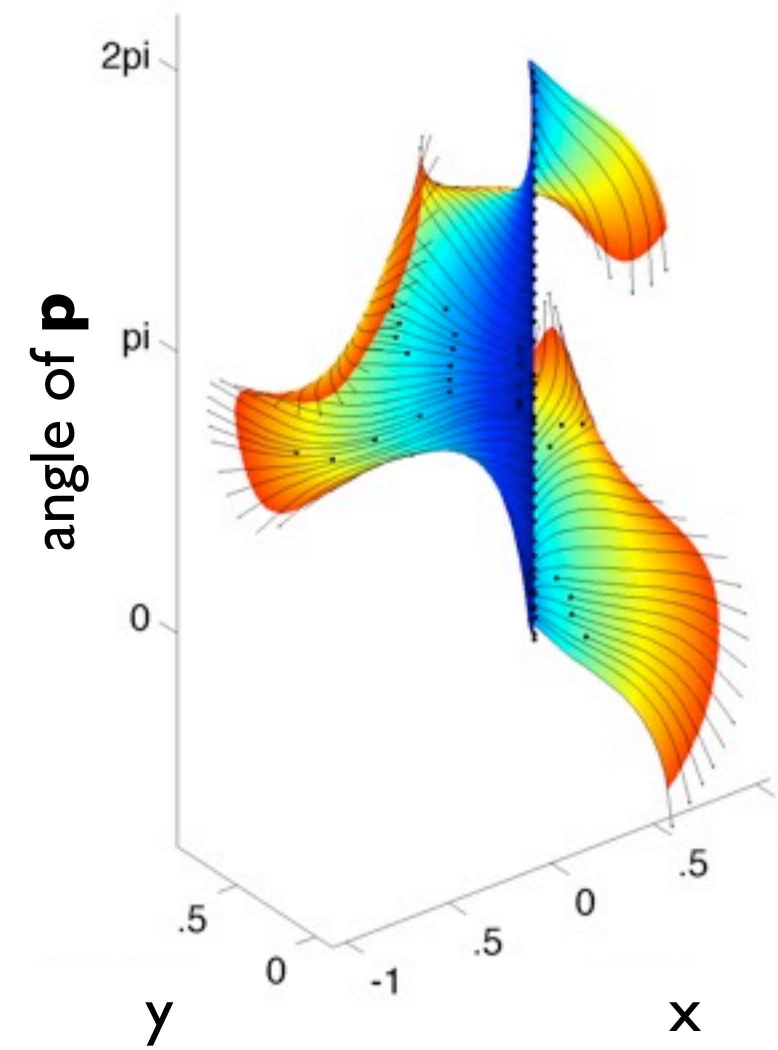

[Rhoads, et al (2010)]

(which happens to coincide with separatrix)

# Key to Lagrangian method: track state **x** *and costate* ***p*** (vector normal to front)

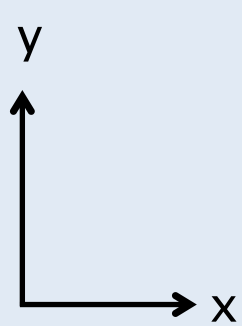
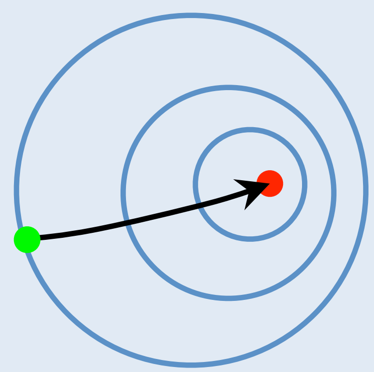
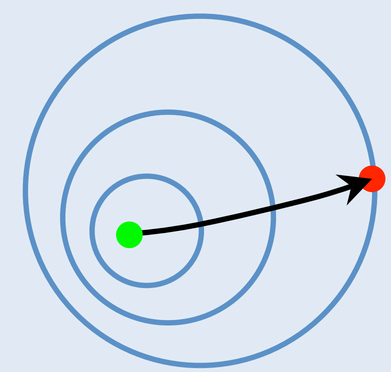
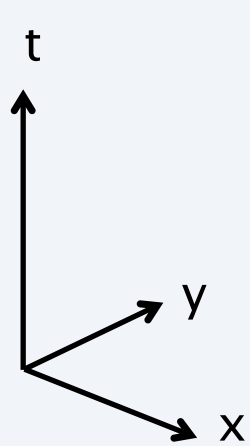
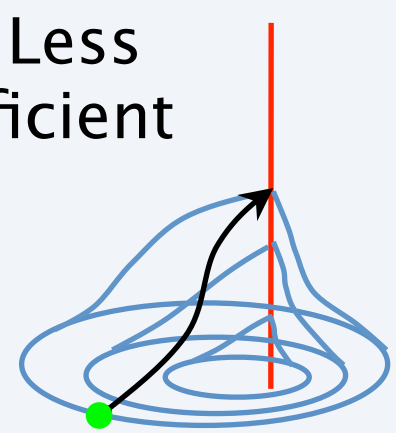
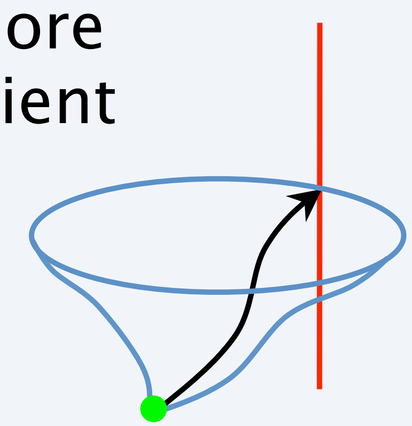
$$\dot{\mathbf{p}} = -\nabla \mathbf{v}^{\mathrm{T}} \mathbf{p}$$

Special property of minimum time control: It's sufficient to track the *angle* of **p**.

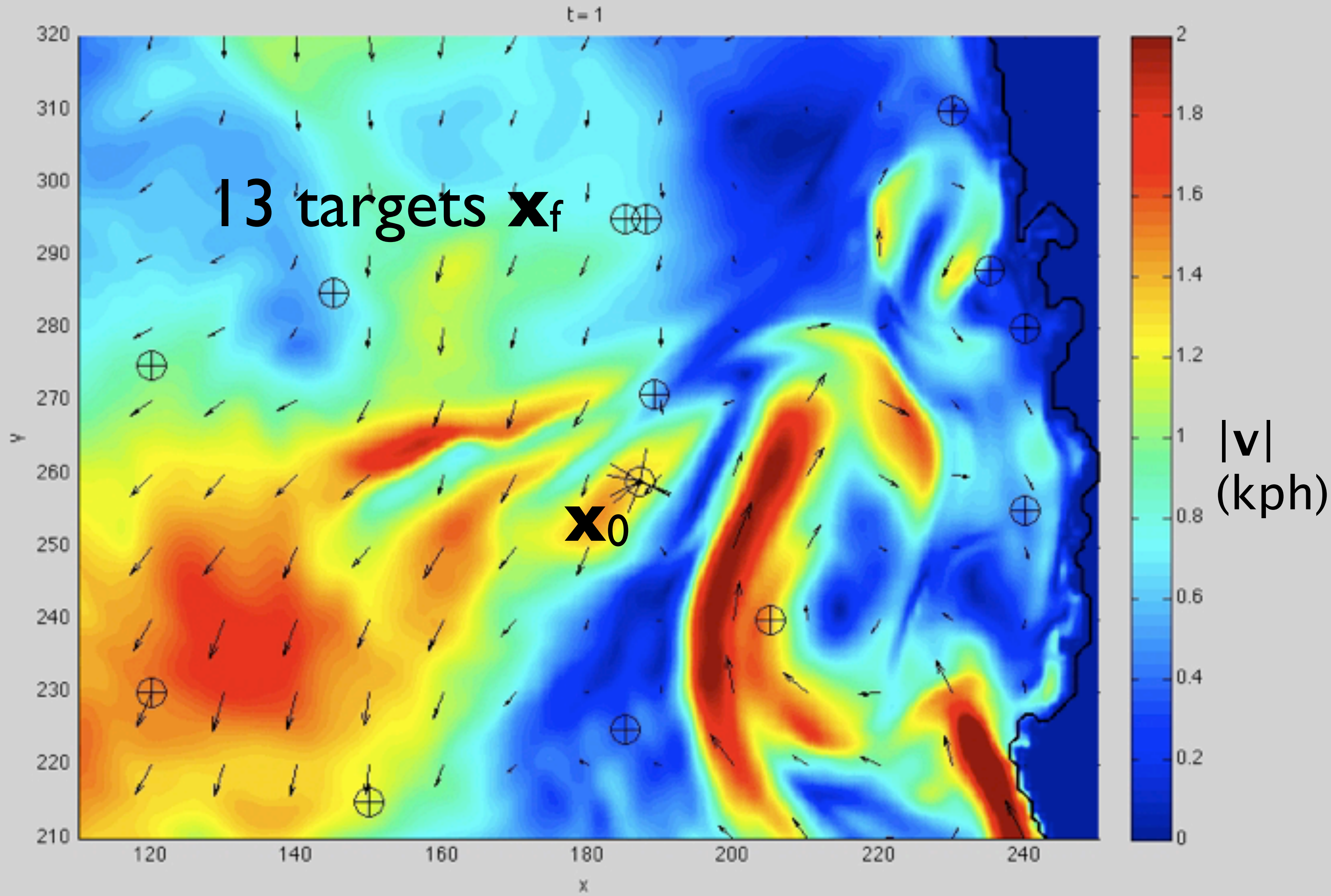



See also [Nishida, et al (2007)], [Holzhuter (2004)], [Osinga, Hauser (2006)], [Mahoney, et al (2012)], [Mitchell, Mahoney (2012)]

# Extends to forward &/or time-varying cases

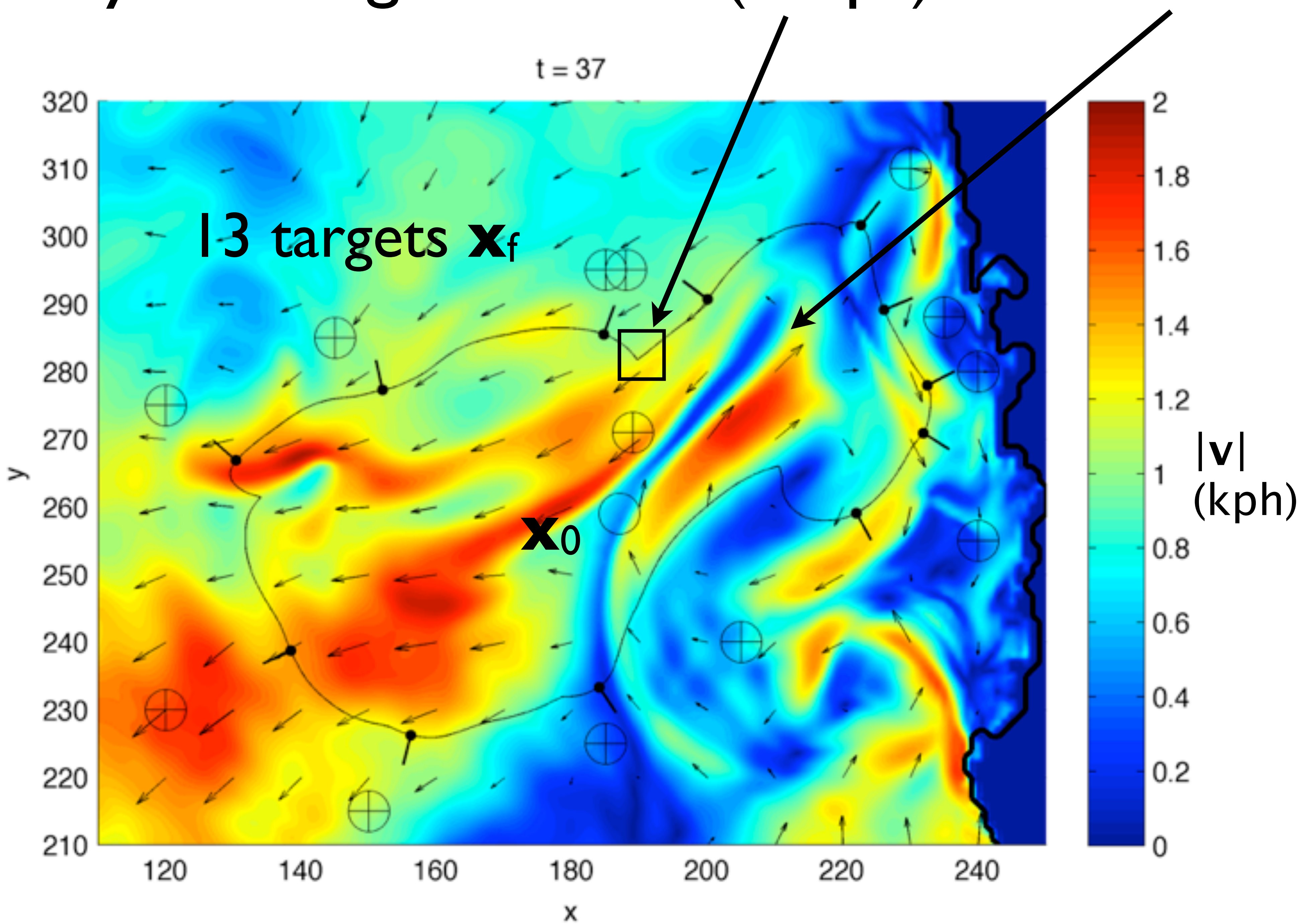
| legend: $\mathbf{x}_0$ (green dot) $\mathbf{x}_f$ (red dot) | backward: "controllability"<br><br>feedback, but… | forward: "reachability"<br><br>no feedback, but… |
|---|---|---|
| time-invariant case; $\mathbf{v}(\mathbf{x},t) = \mathbf{v}(\mathbf{x})$  |  |  |
| time-varying case  | … Less efficient  [Rhoads, et al (2010)] | … More efficient  [Rhoads, et al (2013)] |

# Ex: Adriatic Sea ($s$ = 0.9 kph, $T_{max}$ = 54 h)



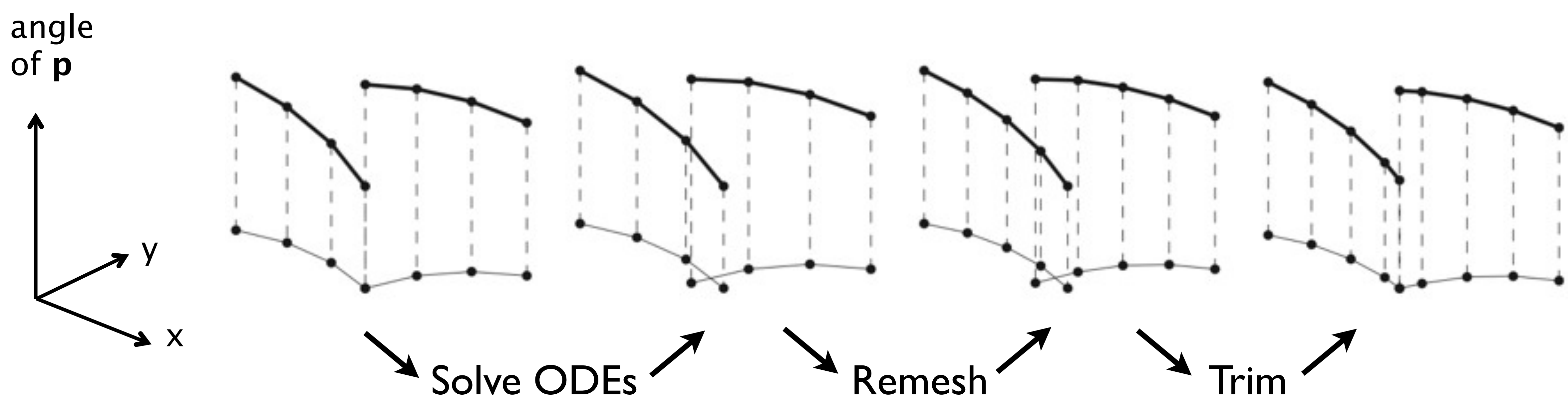

Movie of “reachability” front [Rhoads, et. al, 2013]

# Two key challenges: *shocks* (cusps) and *stretching*



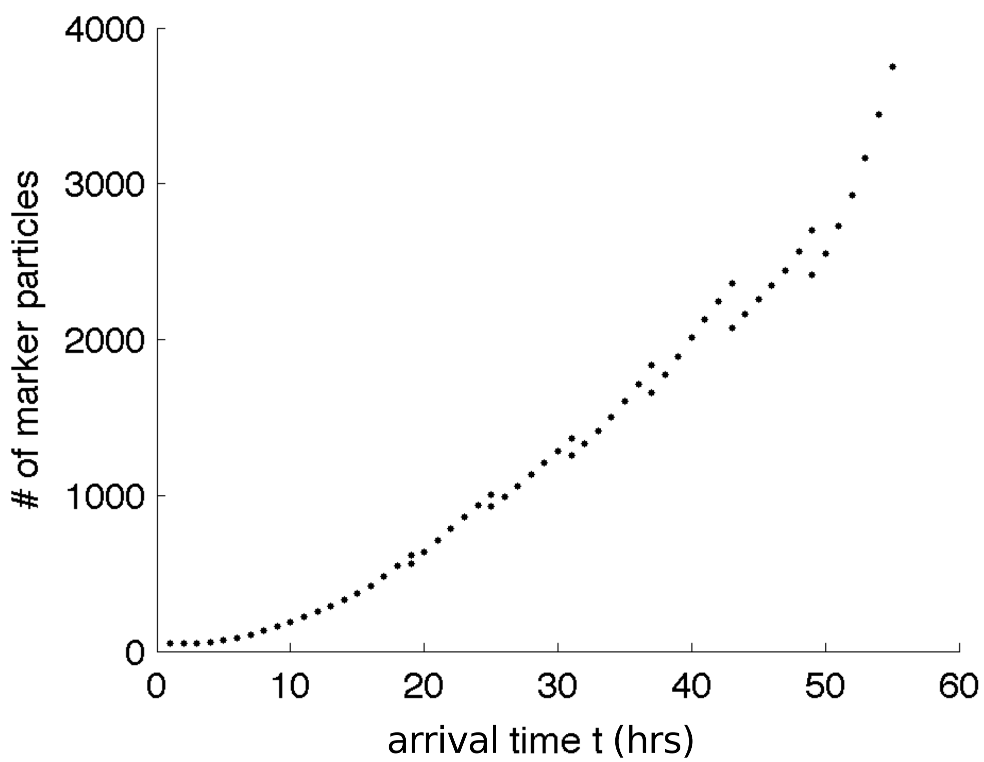

Snapshot of “reachability” front [Rhoads, et. al, 2013]

# Two key ingredients: *trimming* and *remeshing*

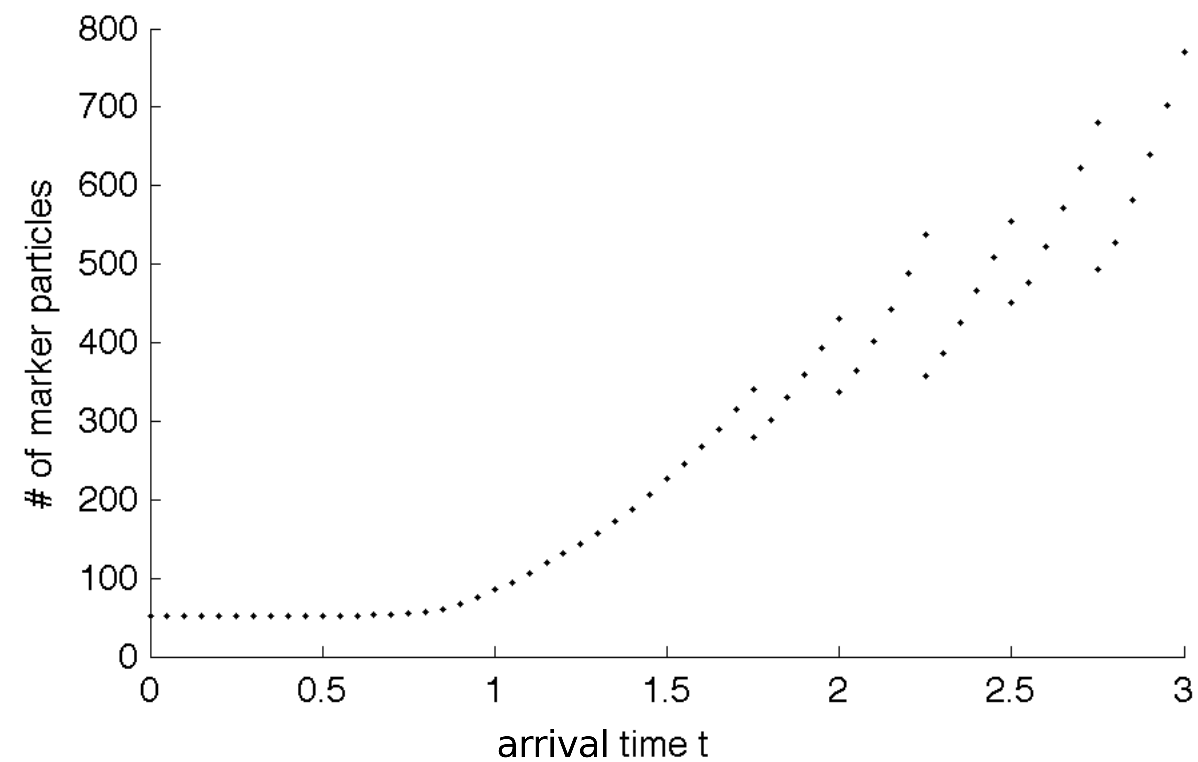



Eulerian alternative: Level Set Methods
[Lolla et al ('12)] (same application)

# Trimming increases algorithmic complexity but decreases computational complexity.

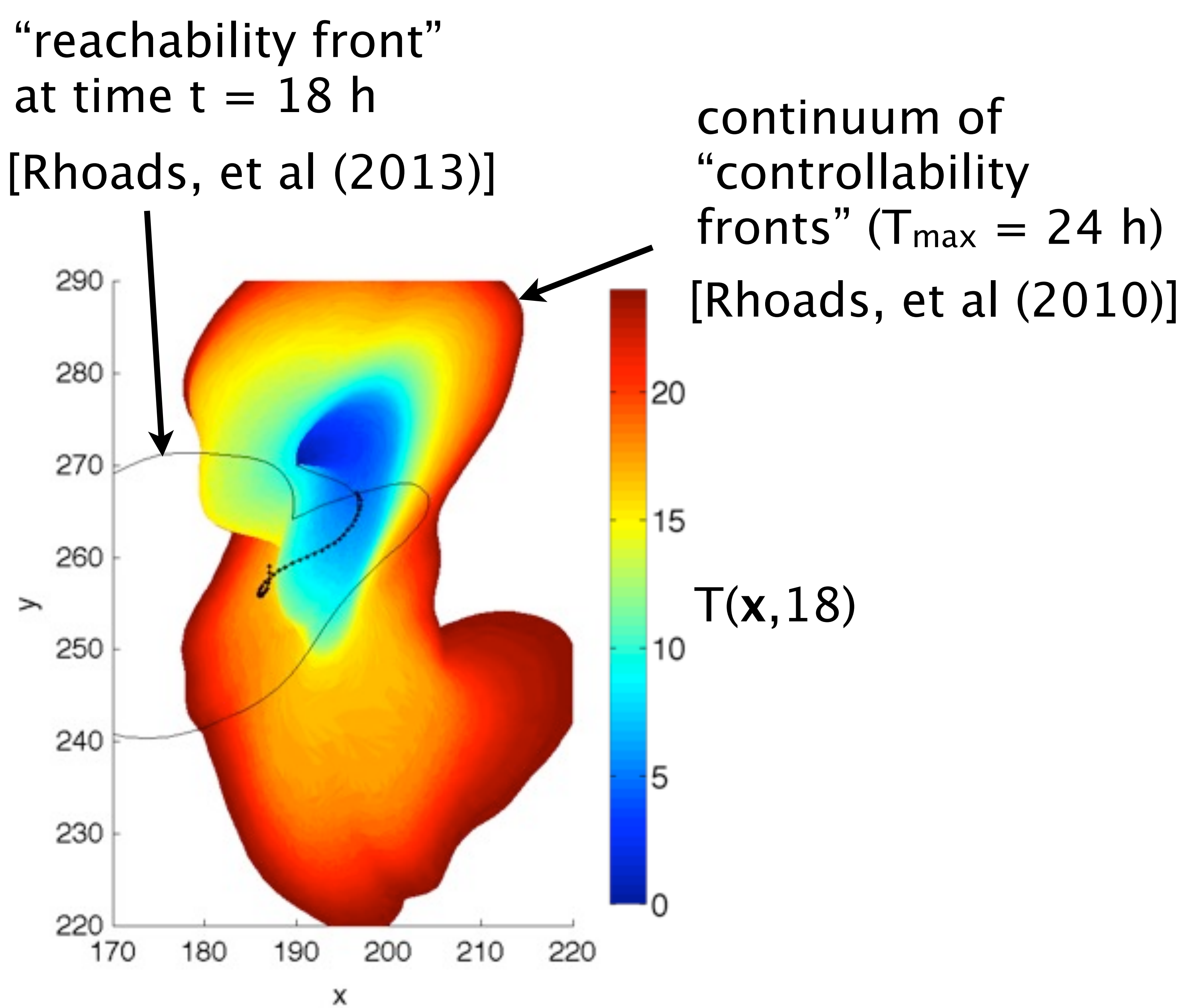



Present example (Adriatic)

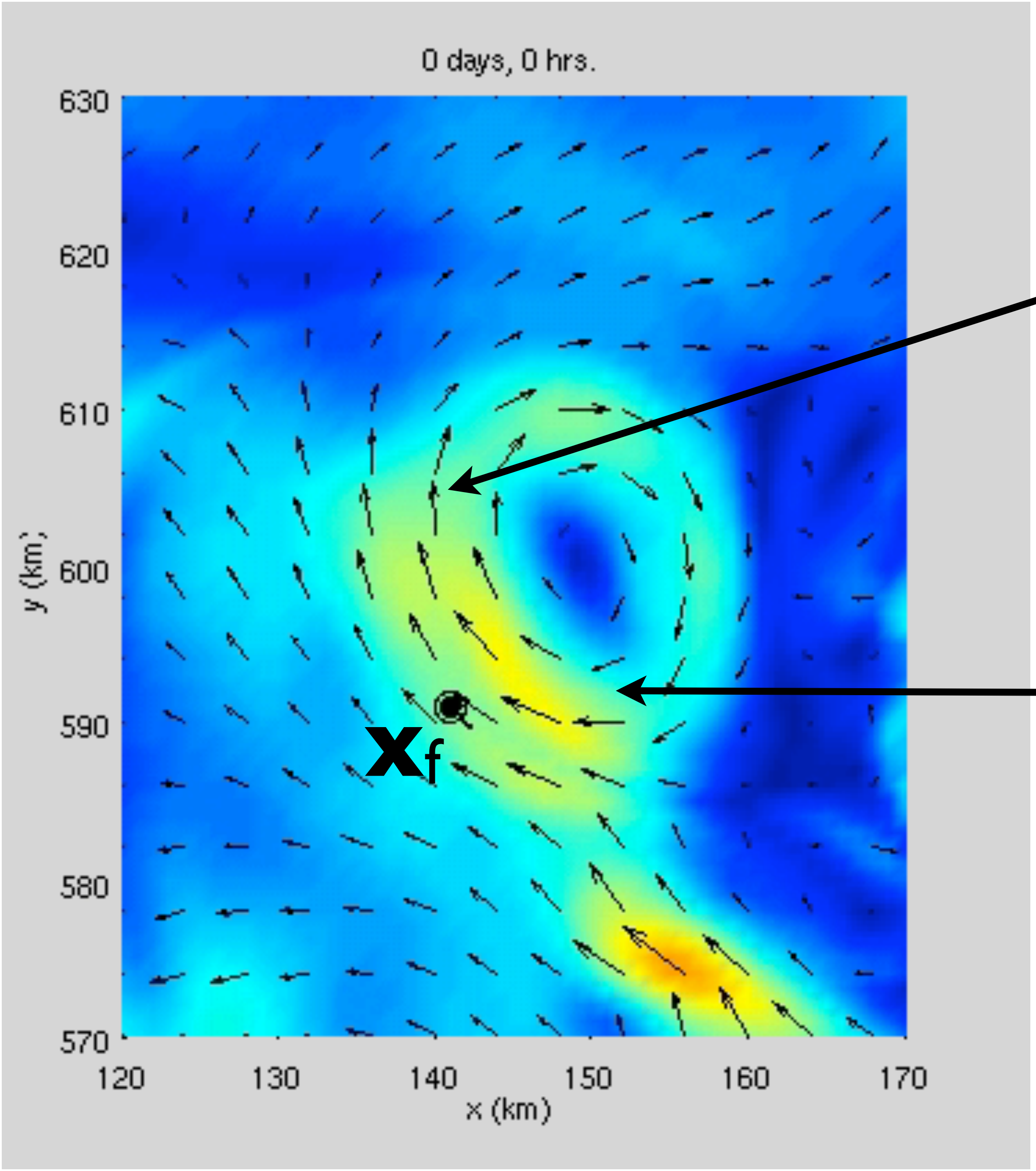



Another example (time-invariant gyre flow). Growth effectively linear instead of exponential.

# Forward method yields same trajectory as backward method… but much more efficiently



# Application: “approximate station keeping”




Naive strategy fights currents.

Min time strategy “dodges” currents (but still never rests)

# Overview

- Minimum time
  - Forward Lagrangian
  - Backward Lagrangian (briefly)
- Minimum energy
  - Backward semi-Lagrangian
- Flow map
  - Alternate perspective on minimum energy control
  - Simple greedy control

# Minimum energy problem

fixed $t_f$
no hard constraint on $\mathbf{x}(t_f)$

end cost ↓ weight on energy ↓

$$\min_{\mathbf{u}} \left\{ h(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} W|\mathbf{u}|^2 dt \right\}$$

↓

HJB: $$\frac{\partial V}{\partial t} + \min_{\mathbf{u}} \left\{ (\mathbf{v}+\mathbf{u}) \cdot \nabla V + W|\mathbf{u}|^2 \right\} = 0$$

↓

$$\mathbf{u} = -\frac{\nabla V}{2W} \quad \left( \text{if } \frac{|\nabla V|}{2W} \le s; \text{ otherwise } \mathbf{u} = -s\frac{\nabla V}{|\nabla V|} \right)$$

Boundary condition: $$V(\mathbf{x}, t_f) = h(\mathbf{x})$$

↓

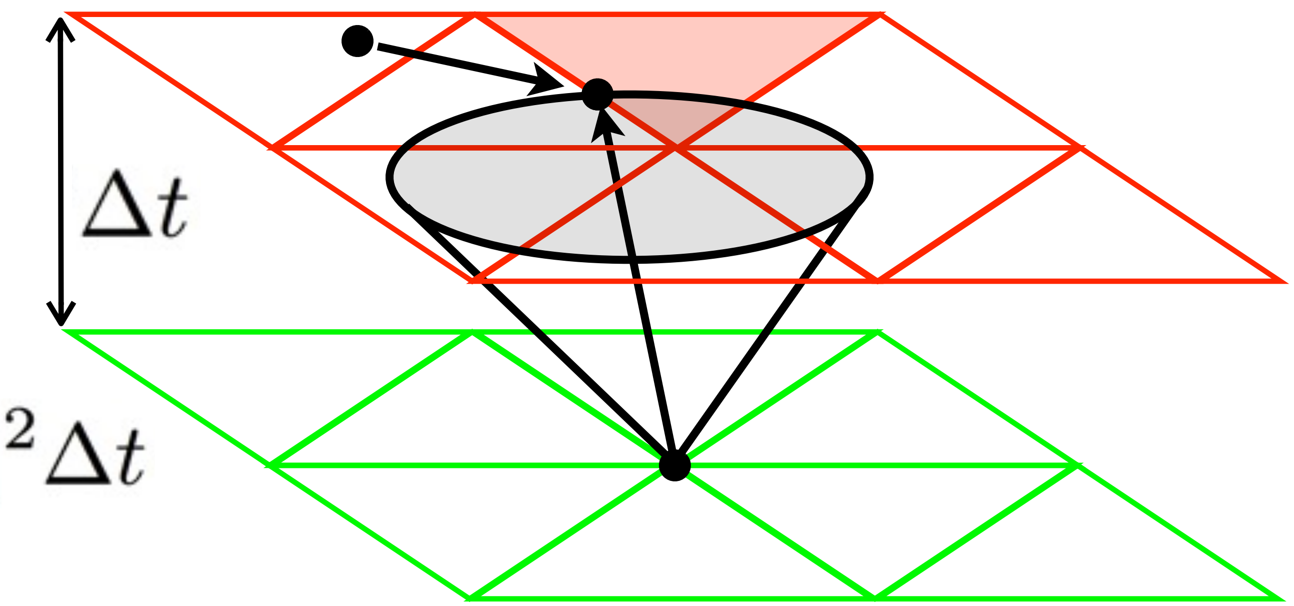
*Backward time-stepping*

# Semi-Lagrangian method (first order... exact pointwise minimization)

$$V'$$

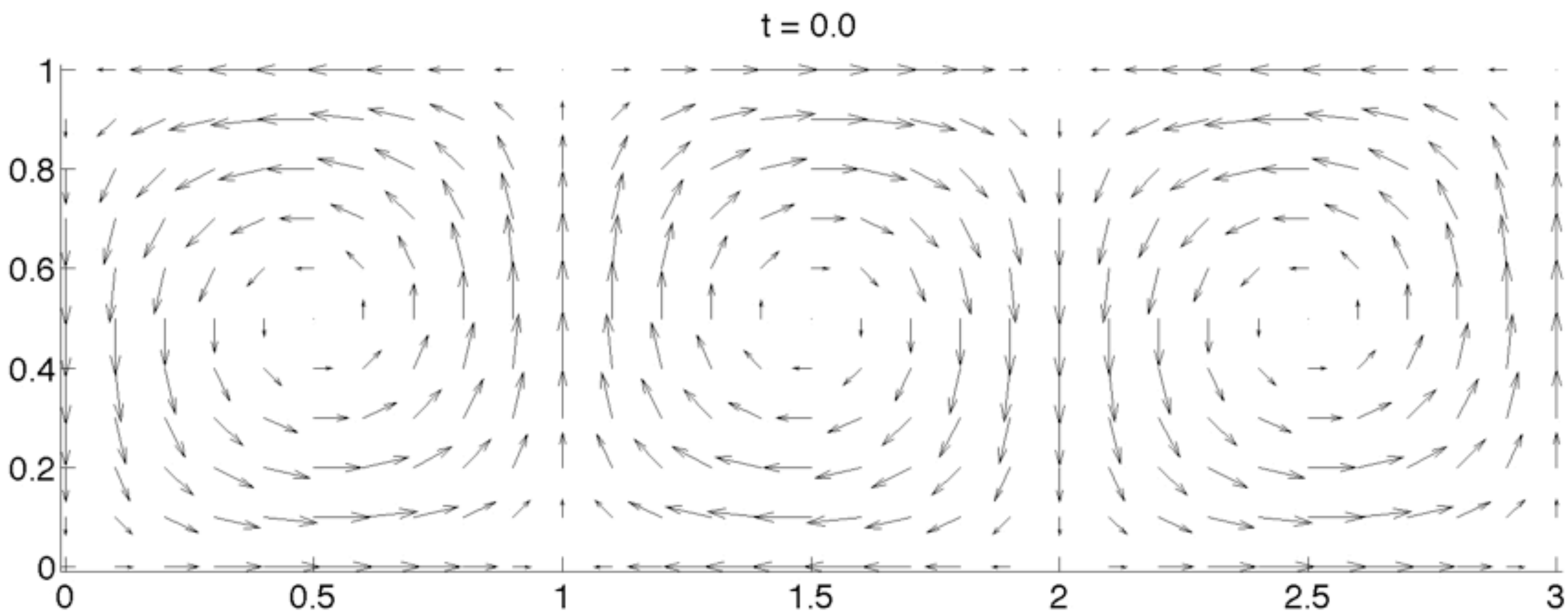
$$\downarrow$$

$$V = V' + W|\mathbf{u}|^2\Delta t$$

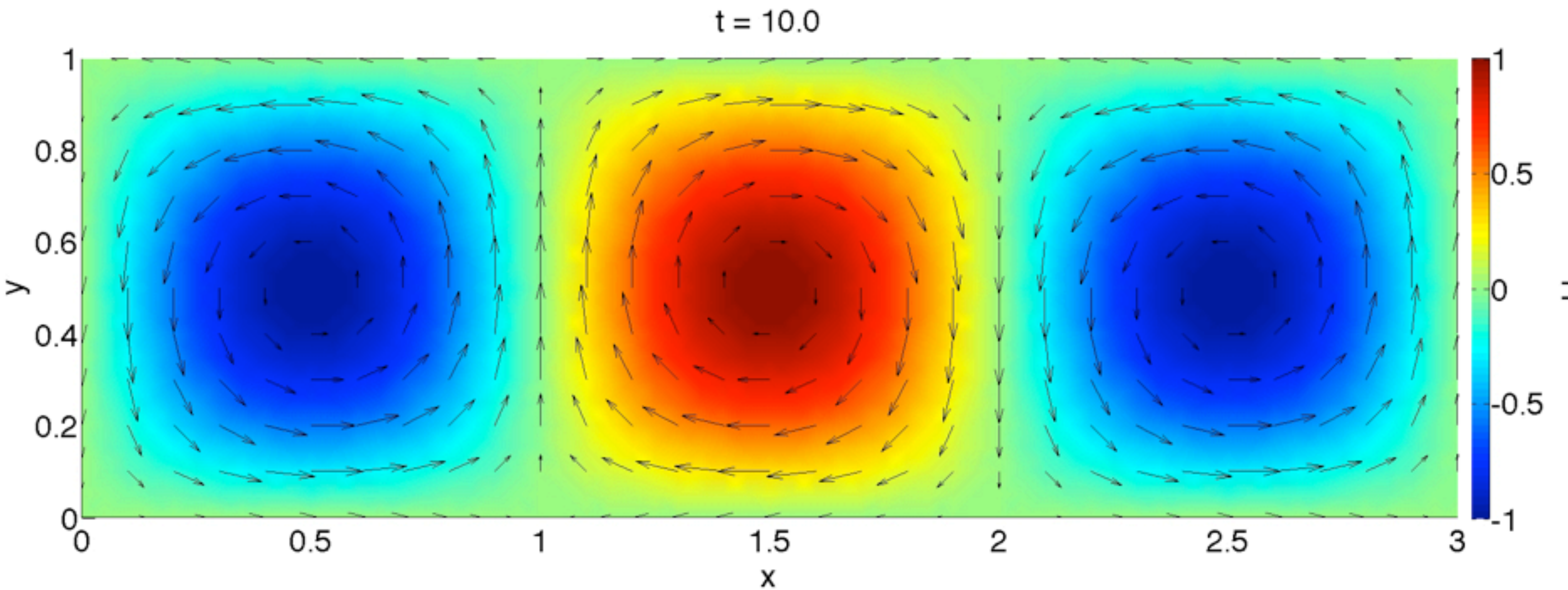



for each time slice
    for each grid point
        for each reachable triangle
            define $V'$ by linear interpolation;
            minimize $V$;
            update overall minimum;

# Ex: $\mathbf{v}(\mathbf{x},t)$ = time-varying 3-gyre flow

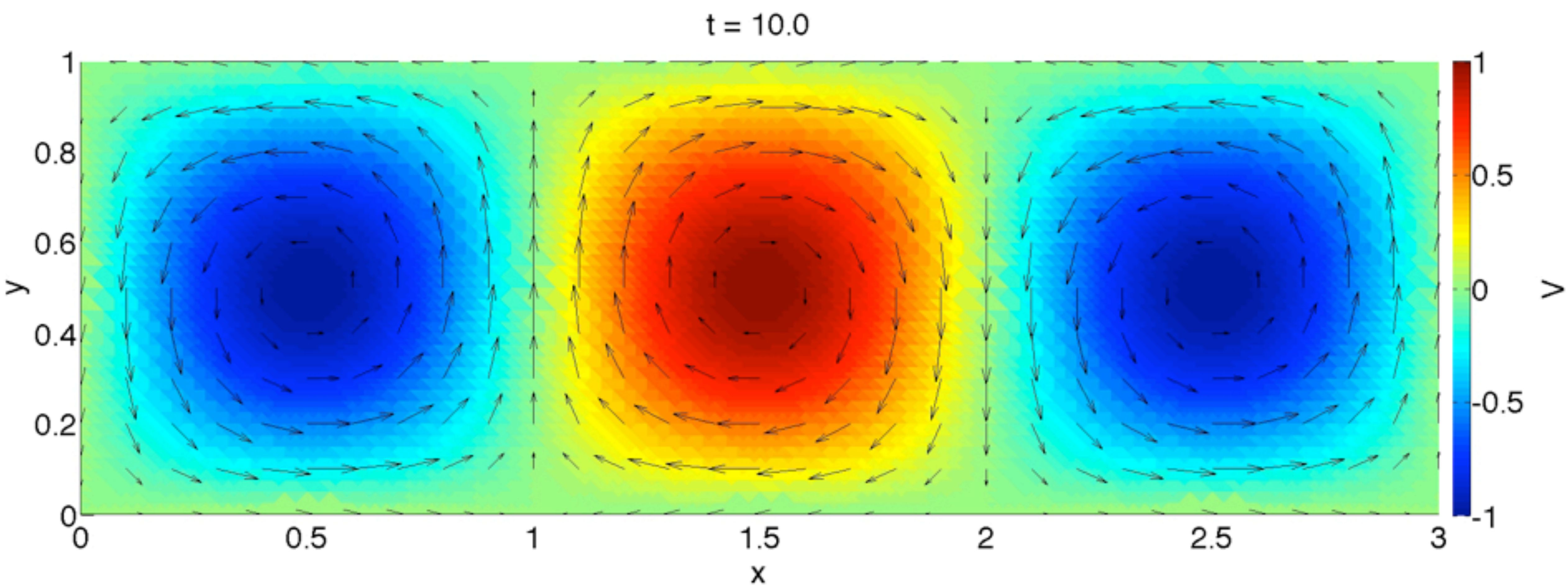


# Effect over backwards time of $\mathbf{v}(\mathbf{x},t)$ on $h(\mathbf{x})$ (a double basin)



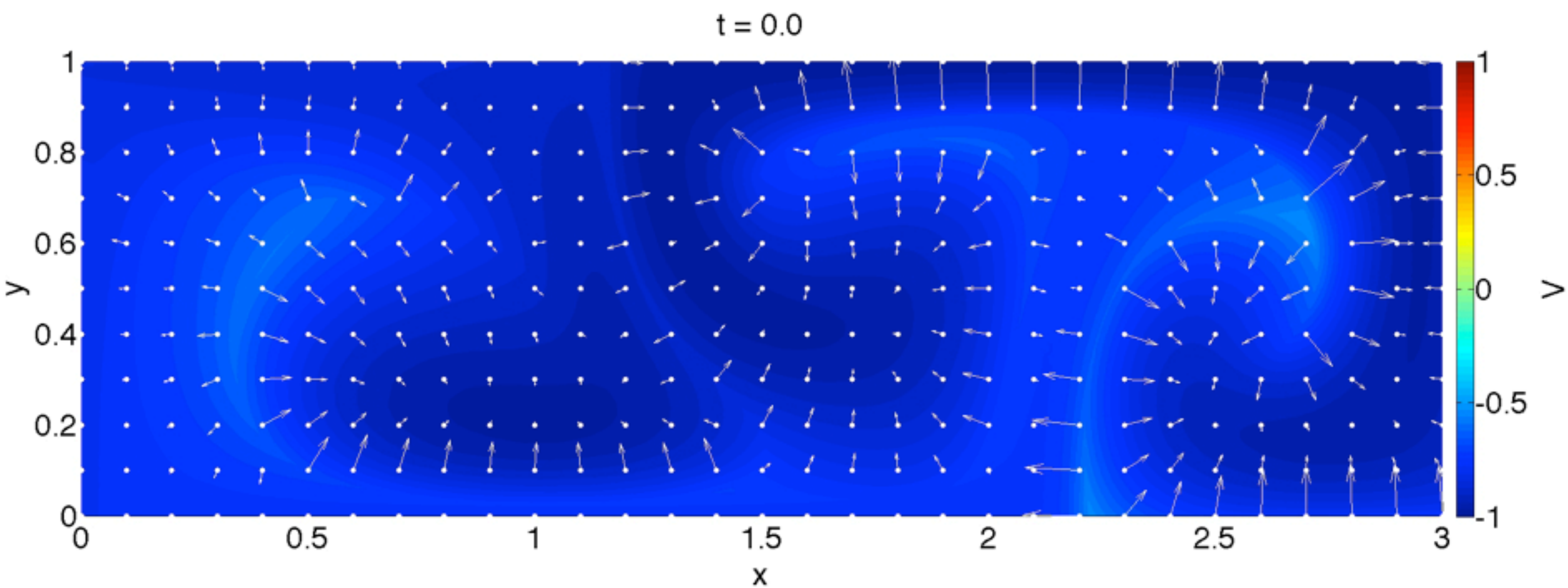

= "*pulled back end cost*" function $H(\mathbf{x},t)$.

# Solution $V(\mathbf{x},t)$ (for $W = 10$, $s = 1$, $t_f = 10$, $\Delta t = 0.1$)

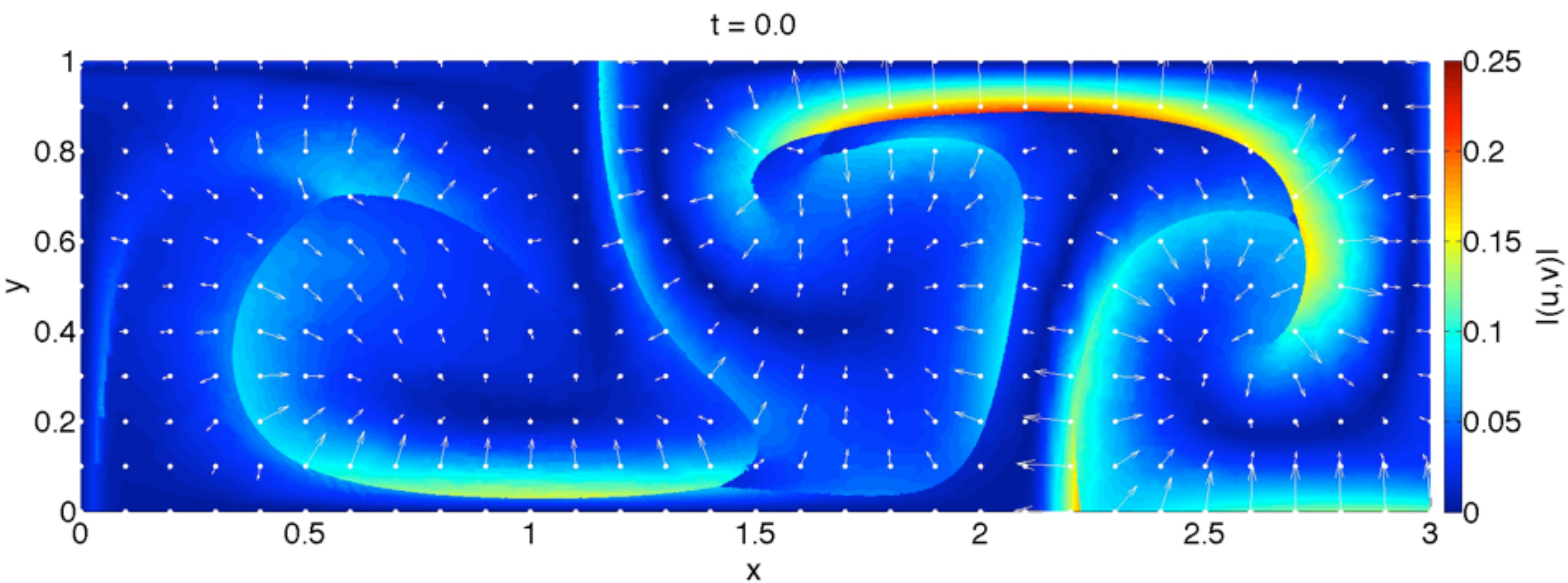



= effect over backwards time of $\mathbf{v}(\mathbf{x},t)$ $\underline{+\ \mathbf{u}(\mathbf{x},t)}$ on $V(\mathbf{x},t_f) = h(\mathbf{x})$

# Simulation forward in time of **u**(**x**,*t*) yields closed-loop **u**(*t*) (white arrows) and **x**(*t*) (white dots)



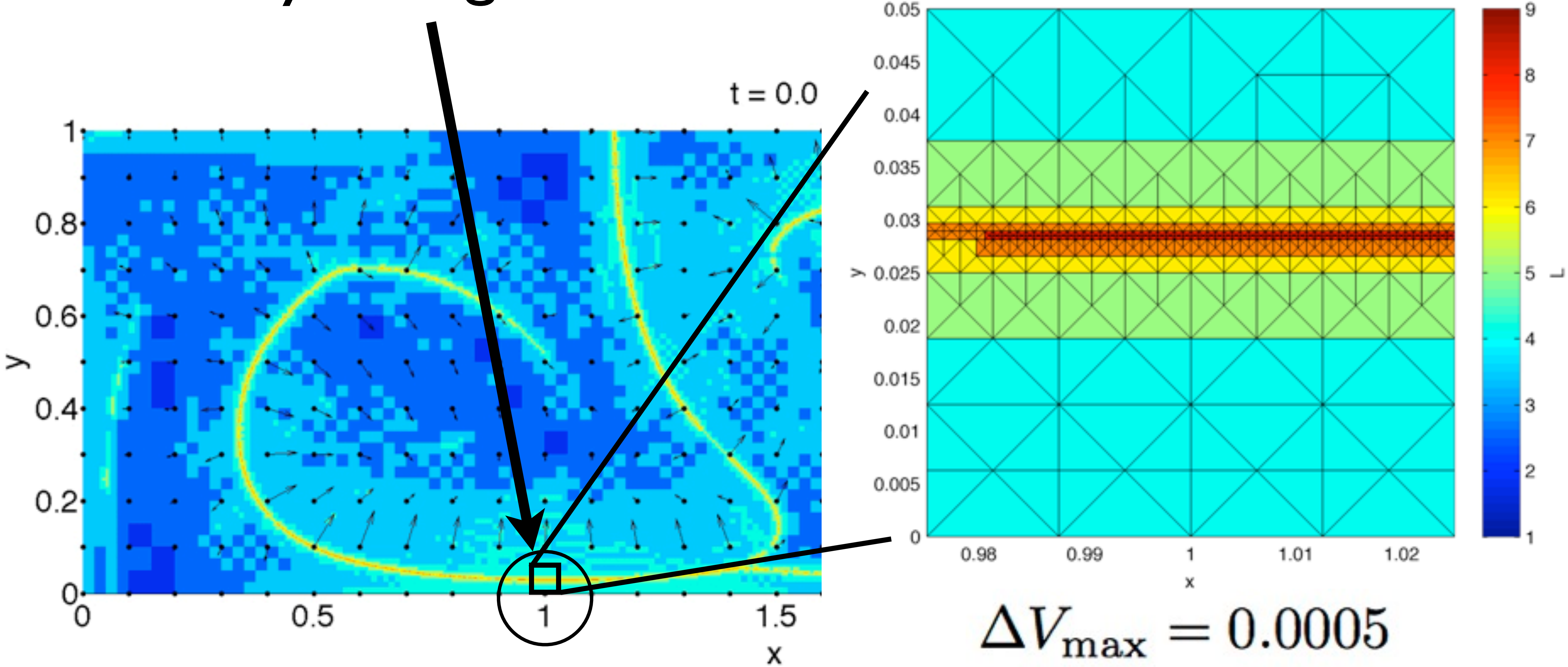

Shocks = cusps in *V*(**x**,t)...




... or discontinuities in **u**(**x**,t)...

Radius = $s \; \Delta t$ = (1.0)(0.1) = 0.1

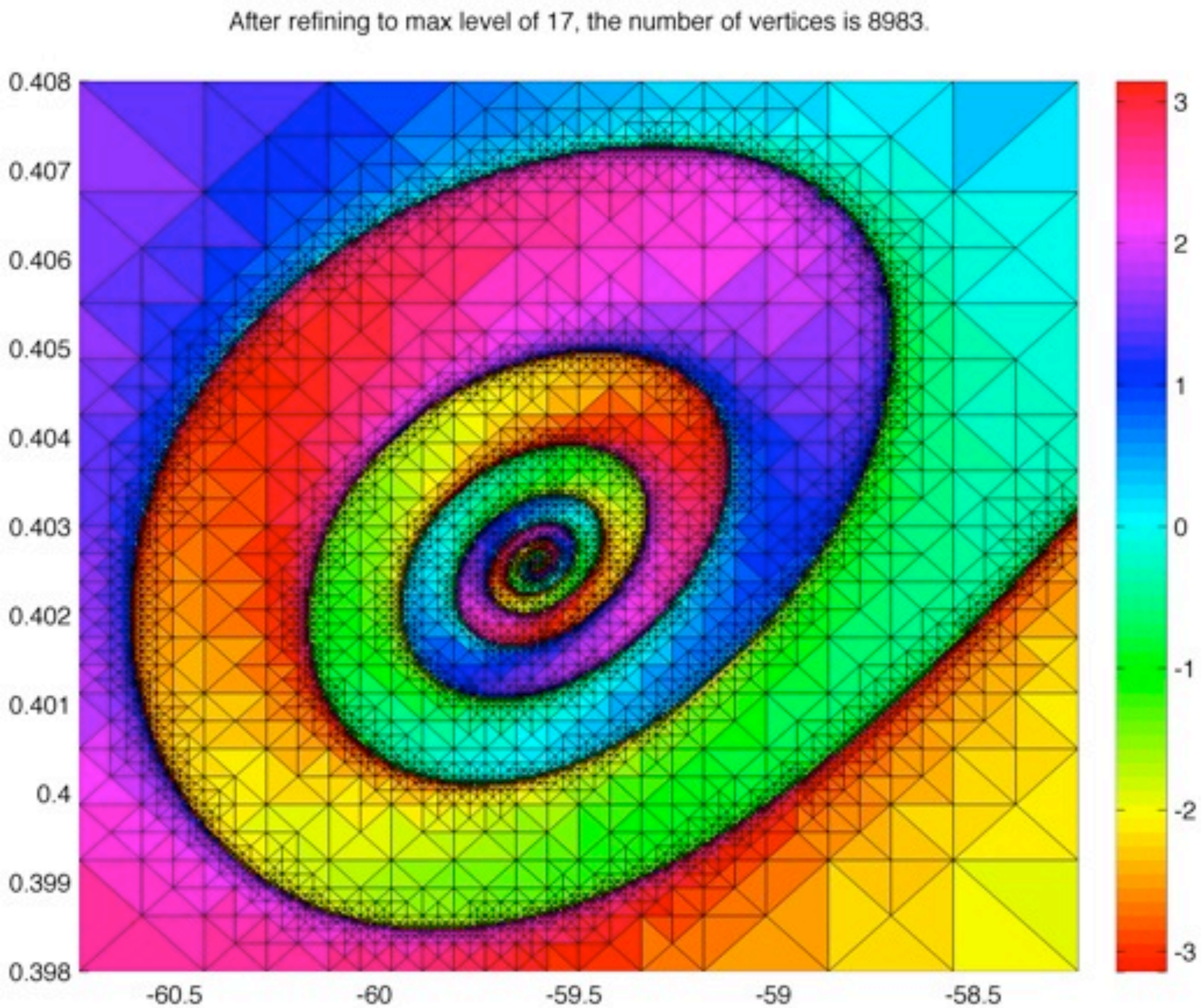
... many triangles



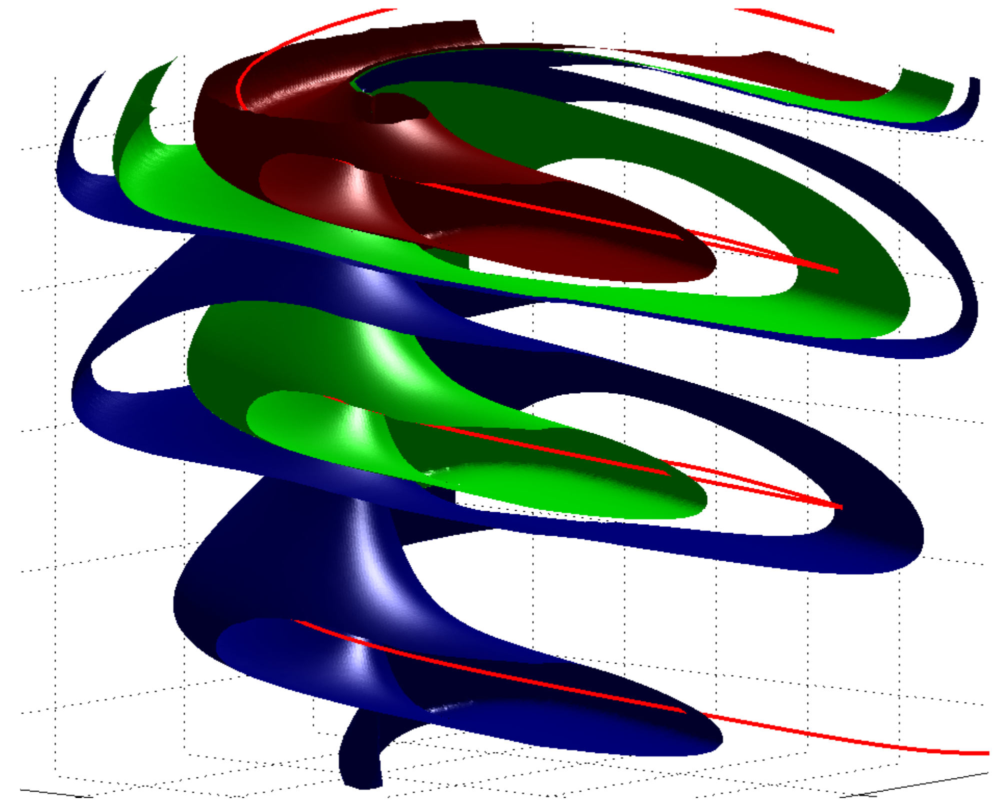

$\Delta V_{\max} = 0.0005$

... or very sharp ridges in grid resolution level *L*

# Another application: neuron phase dynamics




Phase function near unstable fixed point of 2D Hodgkin–Huxley neuron model, via **quadtree (nd–tree)**



3 Isochrons (level sets of phase function) near limit cycle (red) of Hindmarsh – Rose 3D bursting neuron model, via **octree (nd–tree)** adaptive mesh, **Delaunay tetrahedralization**, and custom **marching tetrahedra** isosurface function

# Adaptive grid captures shocks very well but computational complexity issues remain.

- $n_t$ time slices
- $\sim n_x$ grid points (per time slice)
- $\sim(s/n_t)^2 n_x$ reachable triangles (per grid point)
- Suggests $O(s^2 n_x^2 / n_t)$ overall complexity, but:
    - Grid points and triangles are concentrated in a very small area.
    - Memory complexity is $O(n_x n_t)$.

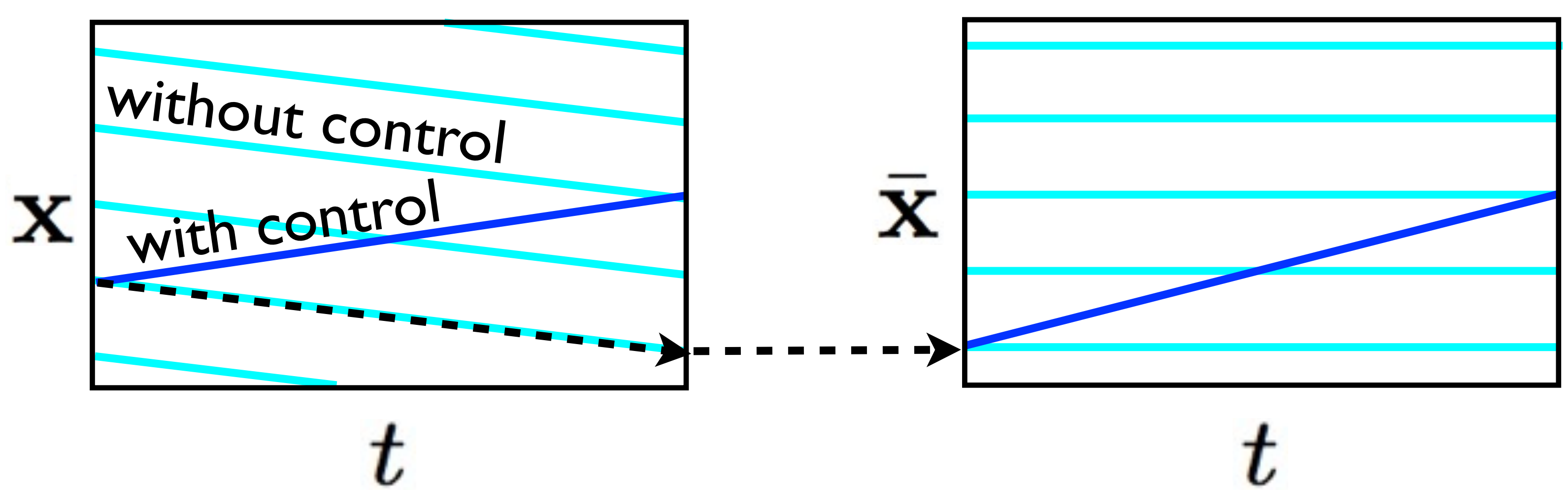
# Overview

- Minimum time
  - Forward Lagrangian
  - Backward Lagrangian (briefly)
- Minimum energy
  - Backward semi-Lagrangian
- Flow map
  - Alternate perspective on minimum energy control
  - Simple greedy control

# Fixed final time flow map $F_t^{t_f}$ yields alternate state coordinates where: (a) "flow" is zero but (b) control is multiplied by time-varying matrix

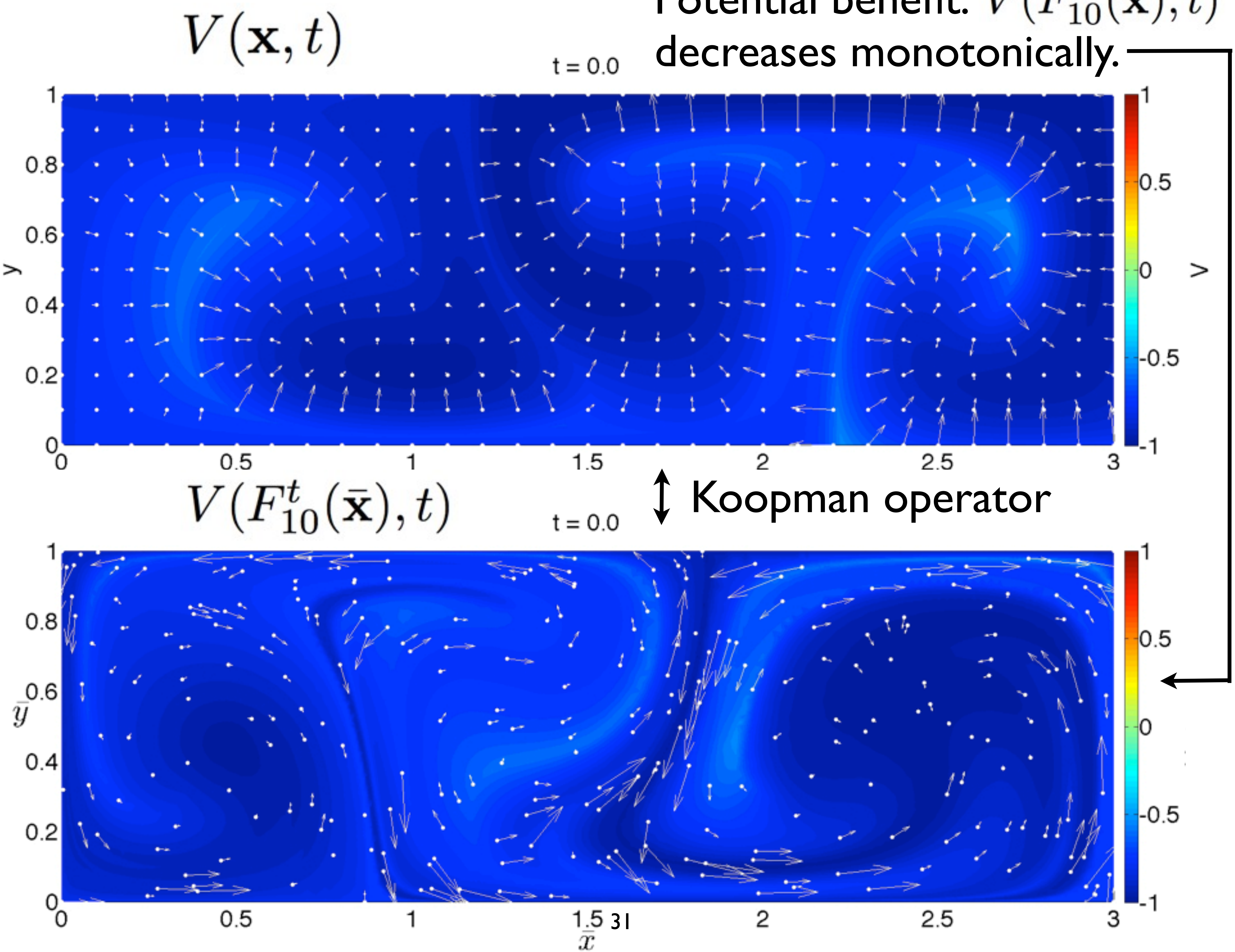
$$\dot{\mathbf{x}} = \mathbf{v} + \mathbf{u} \nearrow \bar{\mathbf{x}} := F_t^{t_f}(\mathbf{x}) \searrow \dot{\bar{\mathbf{x}}} = J\mathbf{u}$$

where $J(\bar{\mathbf{x}}, t) := \nabla F_t^{t_f}(F_{t_f}^{t}(\bar{\mathbf{x}})) = \nabla F_t^{t_f}(\mathbf{x})$
is the flow map Jacobian.




Very simple example (in 1D, with constant v < 0, and thus J = identity):

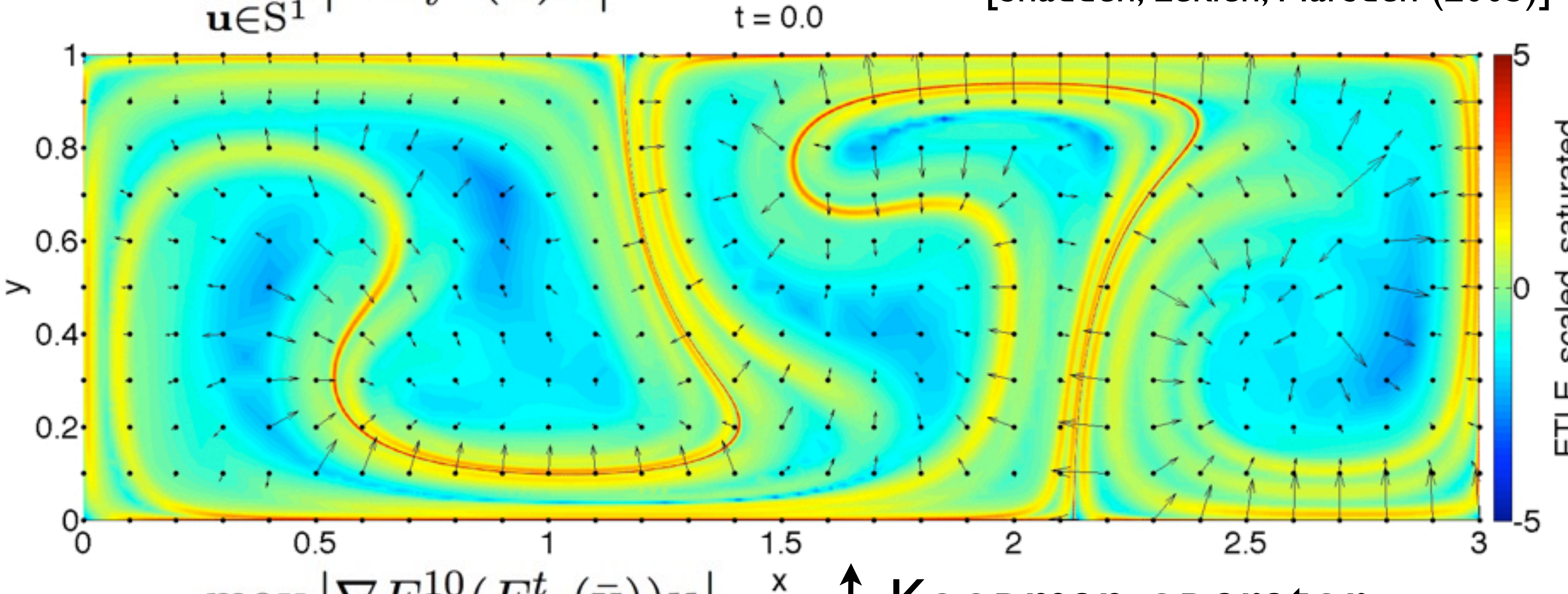
Potential benefit: $V(F_{10}^t(\bar{\mathbf{x}}), t)$ decreases monotonically.




Koopman operator

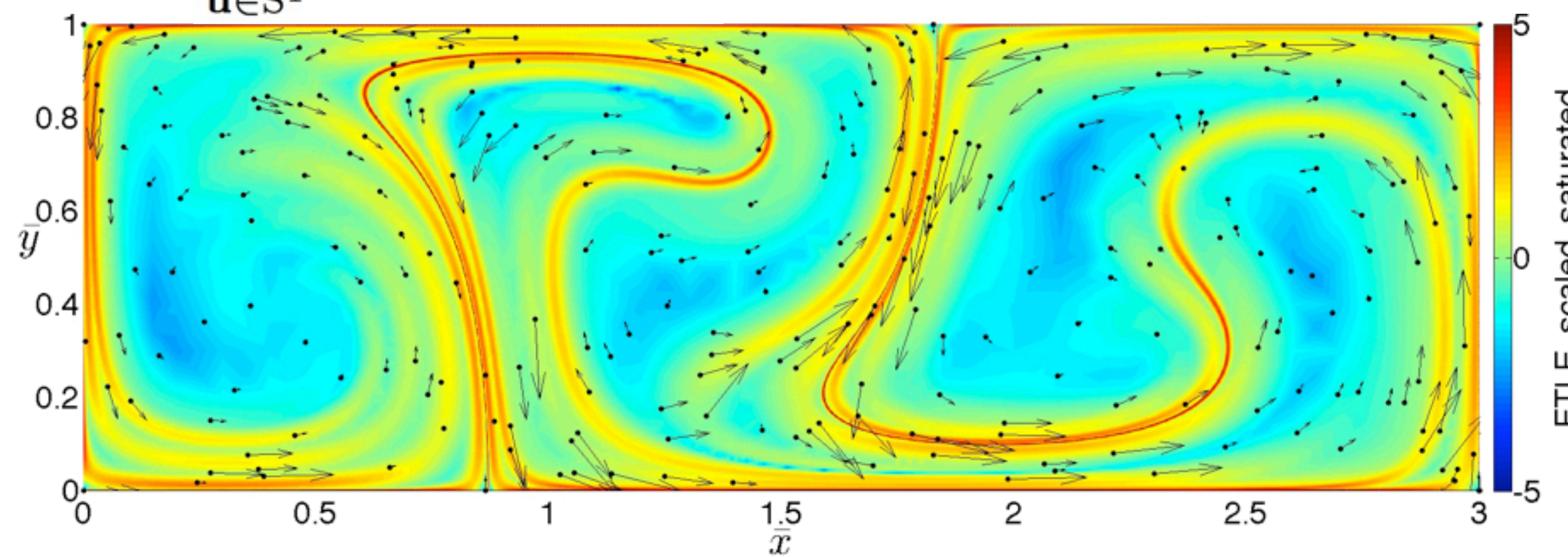
i.e. finite time Lyapunov exponents
[Shadden, Lekien, Marsden (2005)]

$$\max_{\mathbf{u}\in S^1} \left|\nabla F_t^{10}(\mathbf{x})\mathbf{u}\right|$$



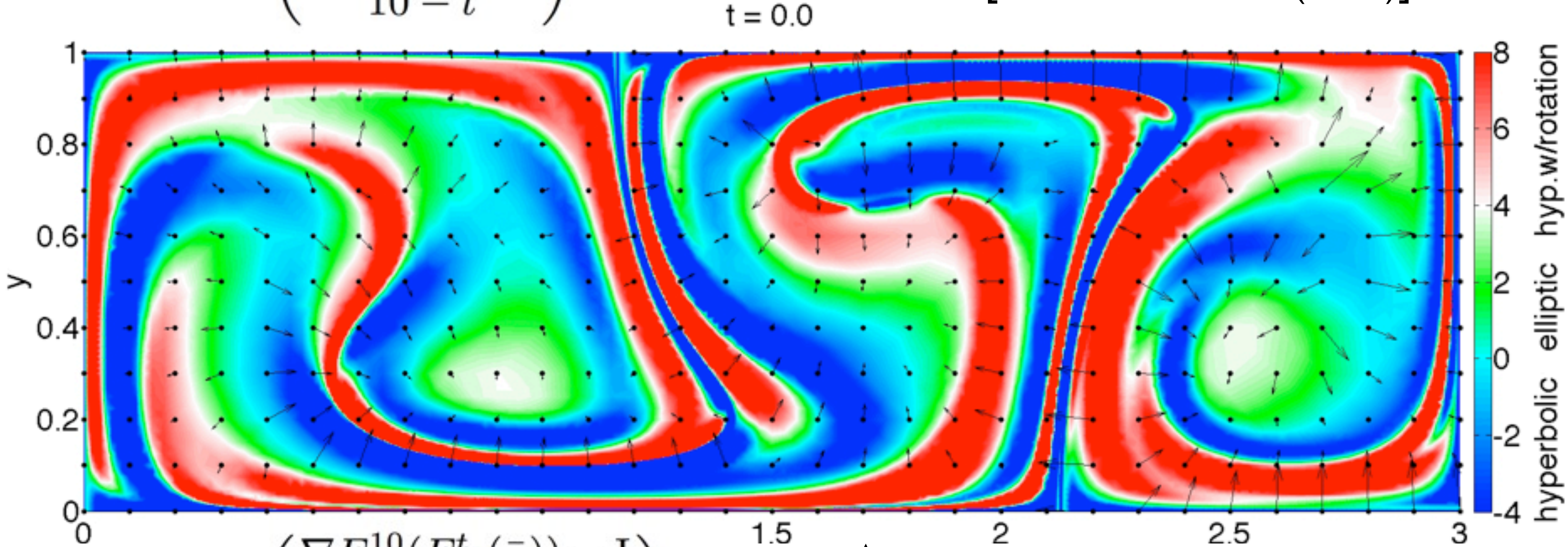

$\updownarrow$ Koopman operator

$$\max_{\mathbf{u}\in S^1} \left|\nabla F_t^{10}(F_{10}^t(\bar{\mathbf{x}}))\mathbf{u}\right|$$



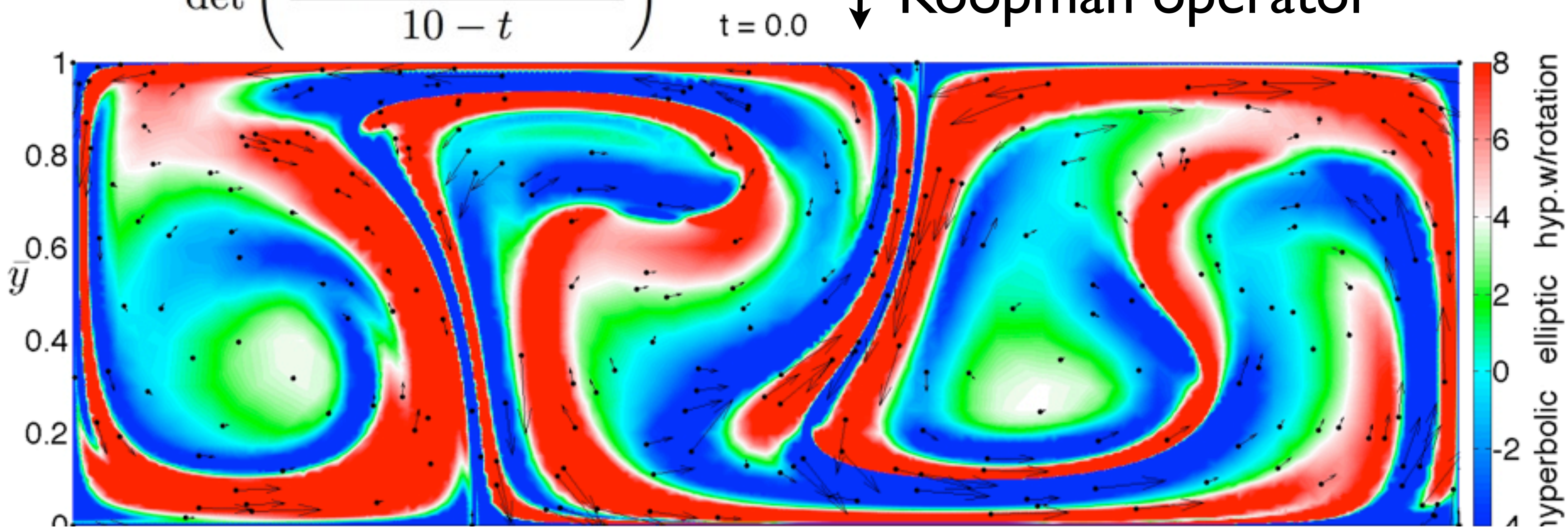
i.e. mesohyperbolicity
[Mezic, Loire, et al (2010)]

$$\det\left(\frac{\nabla F_t^{10}(\mathbf{x}) - \mathrm{I}}{10-t}\right)$$




$$\det\left(\frac{\nabla F_t^{10}(F_{10}^t(\bar{\mathbf{x}})) - \mathrm{I}}{10-t}\right)$$

↕ Koopman operator



# Why not incorporate the control objective?

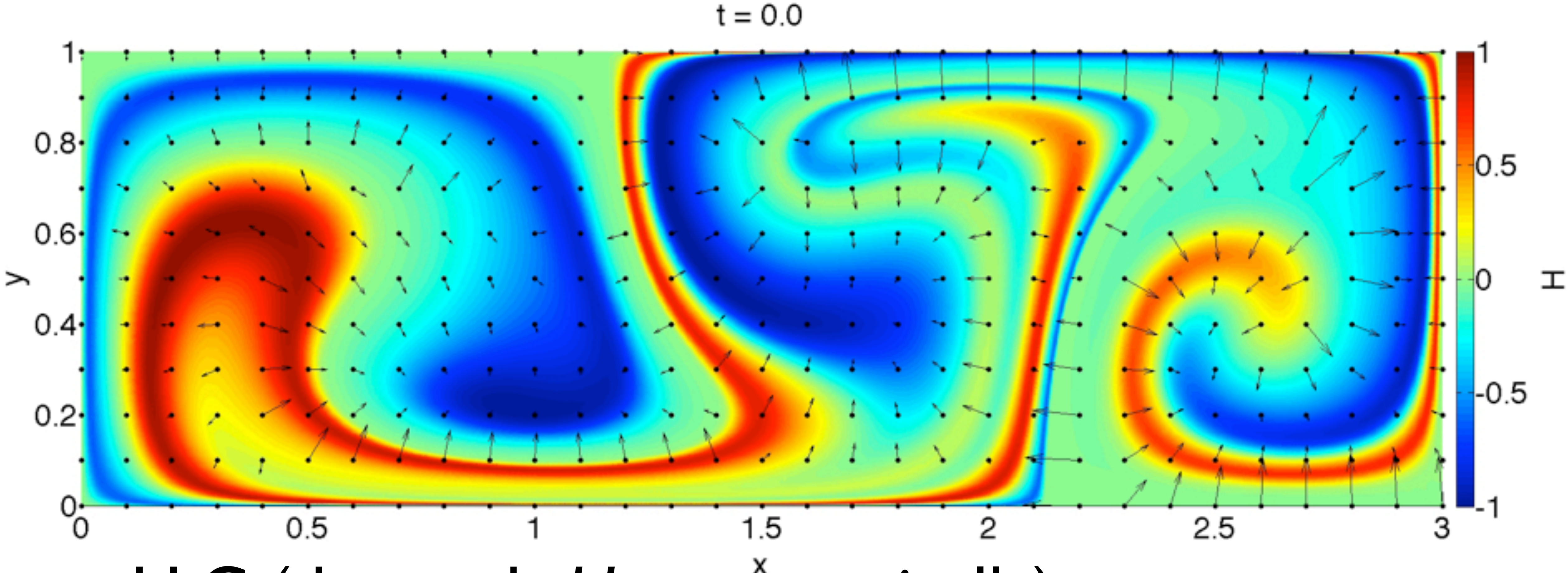
*Pulled back end cost* function $H(\mathbf{x}, t) := h(\bar{\mathbf{x}})$ yields "local Lagrangian control" (LLC)

- Good: $\mathbf{u} = -\frac{\nabla H}{2W} \quad \left(\text{if } \frac{|\nabla H|}{2W} \leq s; \text{ otherwise } \mathbf{u} = -s\frac{\nabla H}{|\nabla H|}\right)$
- Better: $\mathbf{u} = -\frac{\nabla H}{2\bar{W}} \quad \left(\text{if } \frac{|\nabla H|}{2\bar{W}} \leq s; \text{ otherwise } \mathbf{u} = -s\frac{\nabla H}{|\nabla H|}\right),$

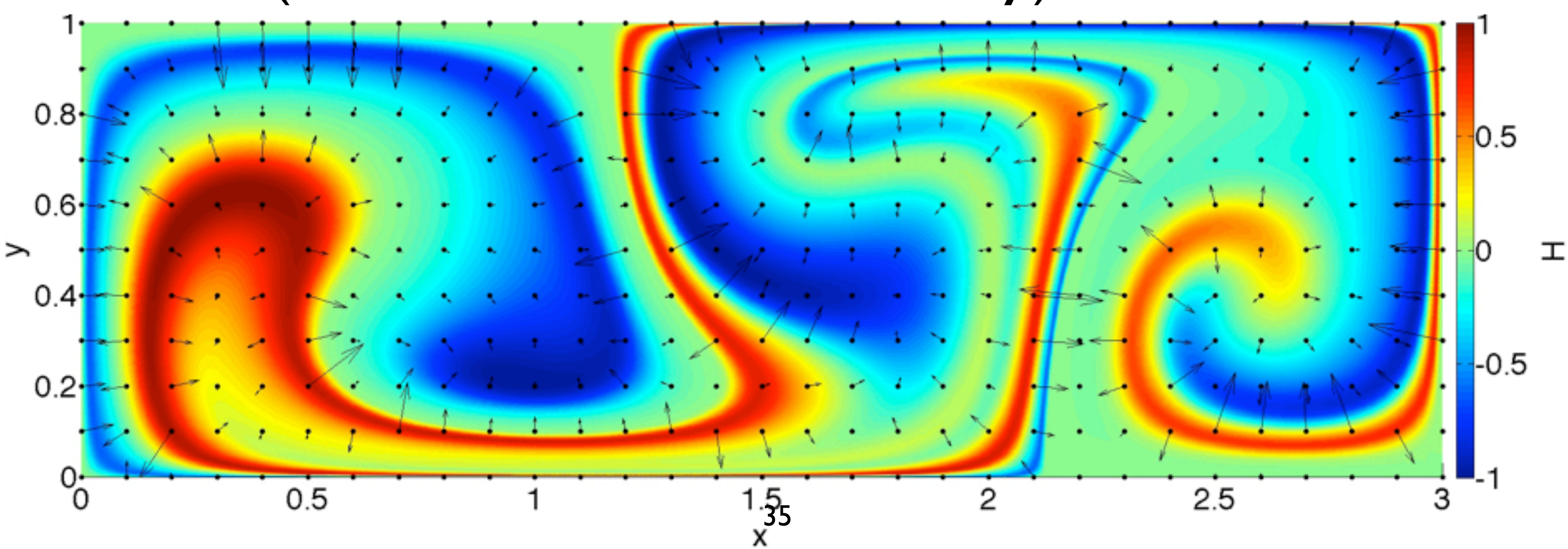
$$\text{where } \bar{W} = \bar{W}(t) := W + t_f - t.$$

For control based on FTLE see [Inanc, Shadden, Marsden (2005)], [Senatore, Ross (2008)]

# Optimal (crosses red ridges in *H*)

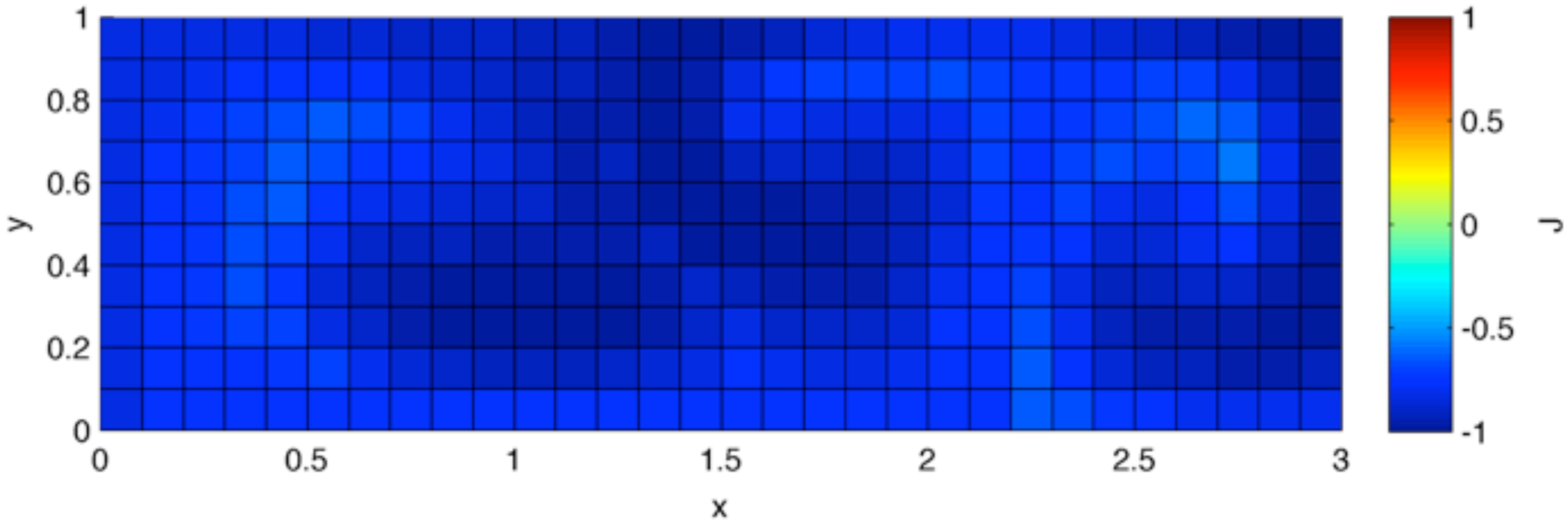



# LLC (descends *H* monotonically)

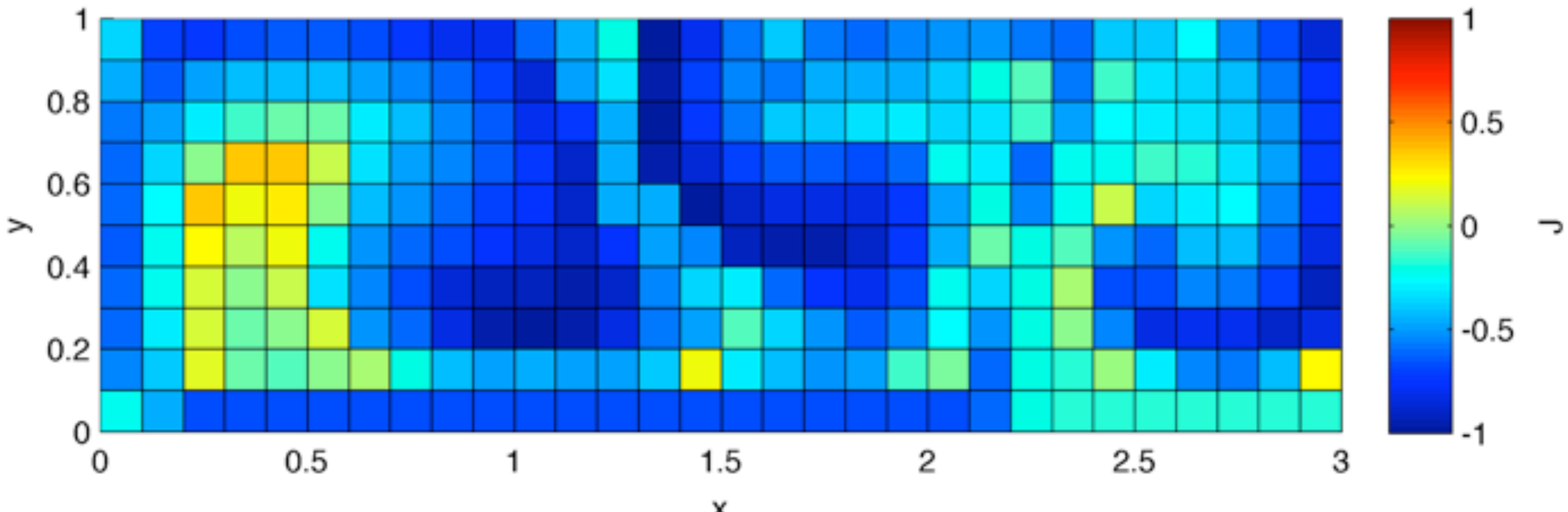


# How suboptimal is LLC?

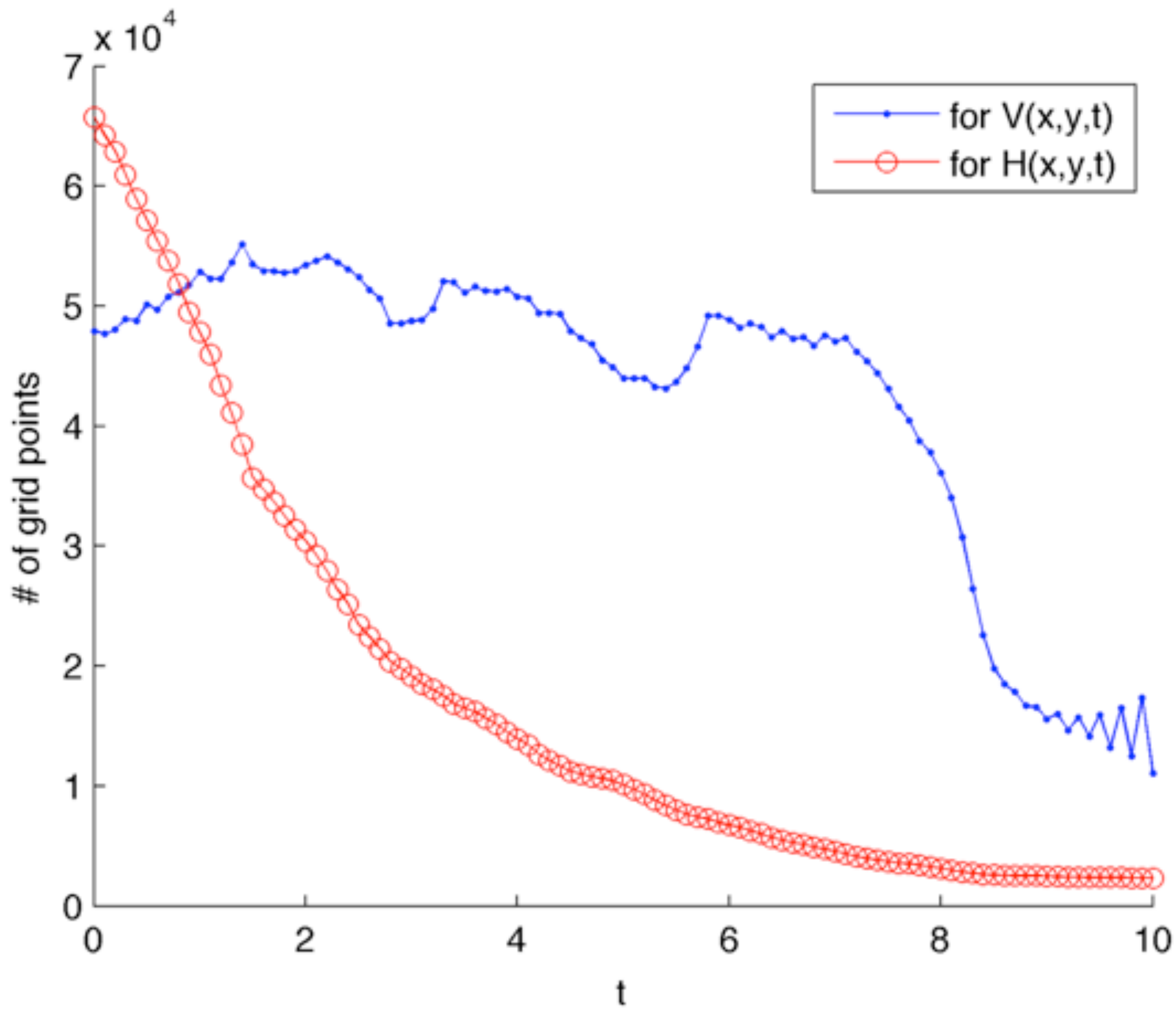
## Optimal trajectory costs




## LLC trajectory costs



# Present *global* method for LLC does not demonstrate the advantage of its being defined *locally*.



| Problem, Method | Minimum time, Forward Lagrangian | Minimum energy, Bwd. semi-Lagrangian | Minimum energy, Using flow map |
|---|---|---|---|
| Related Work | [Lolla et al '12], [Nishida et al '07], [Holzhuter '04], [Osinga Hauser '06] | [Falcone Ferretti '02], [Kumar Vladimirsky '09], [Vladimirsky Zheng '13] | [Inanc Shadden Marsden '05], [Senatore Ross '08], [Garth et al '07] [Miron et al '12] |
| Contribution(s) | Captures not only shocks but **discontinuities** in time-varying case [Rhoads et al '13].<br><br>More efficient than backward Lagrangian method of [Rhoads et al '10]. | **Adaptive grid** and **exact pointwise minimization**. | New perspective on fixed final time control and simple greedy algorithm in LLC. [Rhoads et al '13]<br><br>Adaptive grid works very well for efficient flow map computation. |
| Details | ~15 min, Matlab (vs ~3 hrs for backward method) | ~1 day, C++ | ~30 min, C++ |
| Issues | trimming | computational complexity, first order accuracy | "Zig-zagging" if LLC is computed locally with standard gradient discretizations. |
| Recommendations | Better incorporate into minimum energy w/ W=0 and more general boundary conditions. | Higher order discretization & iterative minimization.<br><br>Lagrangian or Eulerian method. | (a) Solve HJB in new coordinates, (b) better local LLC computation, (c) combine adaptive grid w/ flow map composition. |

# Acknowledgements


- Igor Mezić
- Frederic Gibou, Jeff Moehlis, Andy Teel
- Drew Poje, Alexandre Mauroy
- Brad Paden, Bassam Bamieh, Gustavo Ponce, Raya Feldman
- Sophie Loire, Bryan Eisenhower, Gunjan Thakur, Michael Georgescu, Ryan Mohr, Hassan Arbabi, Marin Sigurdson
- Marko Budisic, George Mathew
- All my other friends
- ME Departmental staff
- Brett Rhoads, Stephanie Fleig, Kathy Fleig, Tom Fleig
- Charlotte Becker
- Office of Naval Research

Thank you.

# Some references


[1] Bryson, A., Ho, Y., 1975. Applied optimal control. Taylor & Francis Bristol, PA.

[2] Falcone, M., Ferretti, R., 2002. Semi-lagrangian schemes for hamilton–jacobi equations, discrete representation formulae and godunov methods. Journal of computational physics 175, 559–575.

[3] Holzhüter, T., 2004. Optimal regulator for the inverted pendulum via Euler–Lagrange backward integration. Automatica 40, 1613–1620.

[4] Nishida, T., Sugihara, K., Kimura, M., 2007. Stable marker-particle method for the Voronoi diagram in a flow field. Journal of Computational and Applied Mathematics 202, 377–391.

[5] Osinga, H., Hauser, J., 2006. The geometry of the solution set of nonlinear optimal control problems. Journal of Dynamics and Differential Equations 18, 881–900.

[6] Petres, C., Pailhas, Y., Patron, P., Petillot, Y., Evans, J., Lane, D., 2007. Path planning for autonomous underwater vehicles. IEEE Transactions on Robotics 23, 331–341.

[7] Rhoads, B., Mezić, I., Poje, A., 2010. Minimum time feedback control of autonomous underwater vehicles, in: Decision and Control (CDC), 2010 49th IEEE Conference on, IEEE. pp. 5828–5834.

[8] Sethian, J., 1996. A fast marching level set method for monotonically advancing fronts. Proceedings of the National Academy of Sciences of the United States of America 93, 1591.

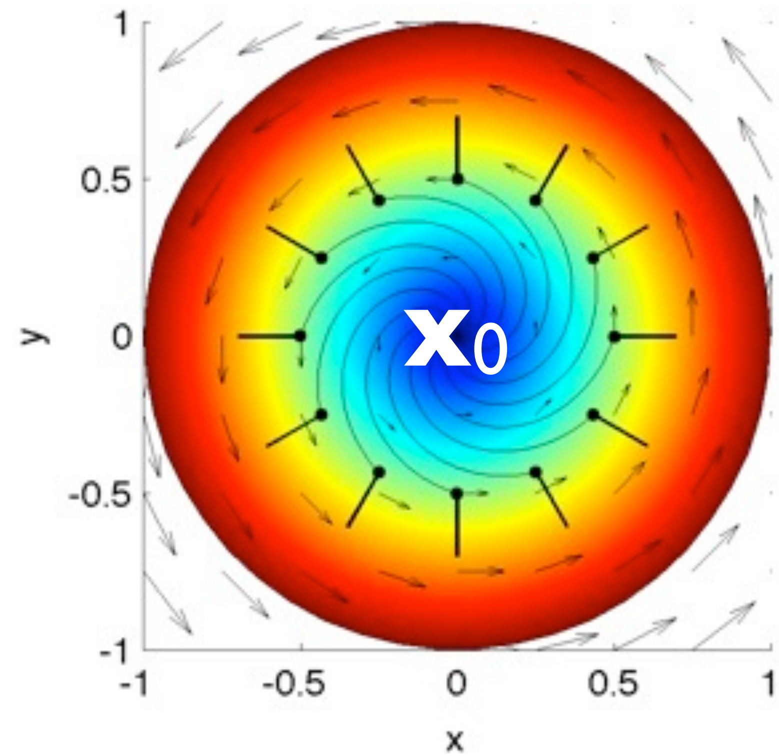
[9] Sethian, J., Vladimirsky, A., 2001. Ordered upwind methods for static Hamilton–Jacobi equations. Proceedings of the National Academy of Sciences of the United States of America 98, 11069.

[10] Tsitsiklis, J., 1995. Efficient algorithms for globally optimal trajectories. IEEE Transactions on Automatic Control 40, 1528–1538.

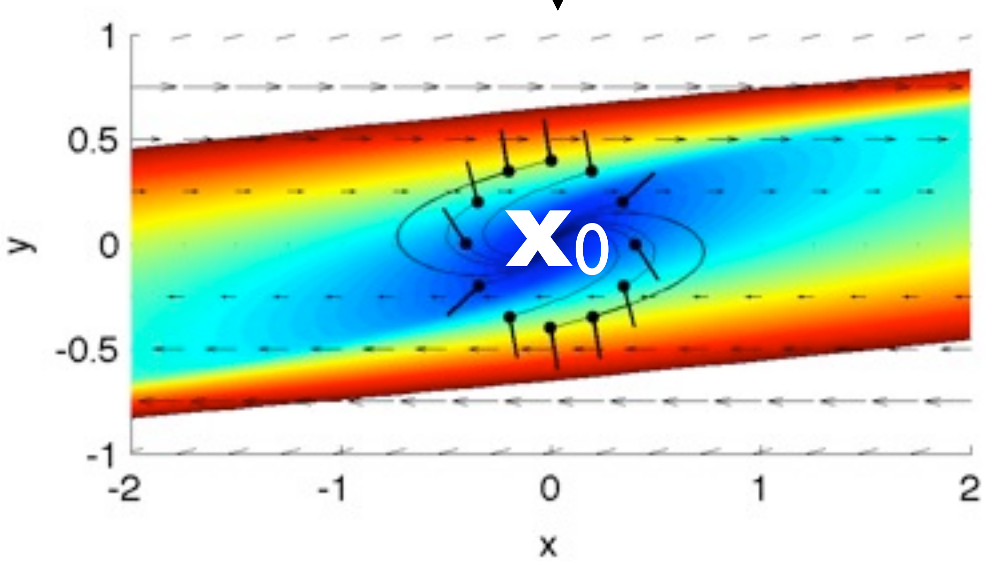
# Supplemental Slides

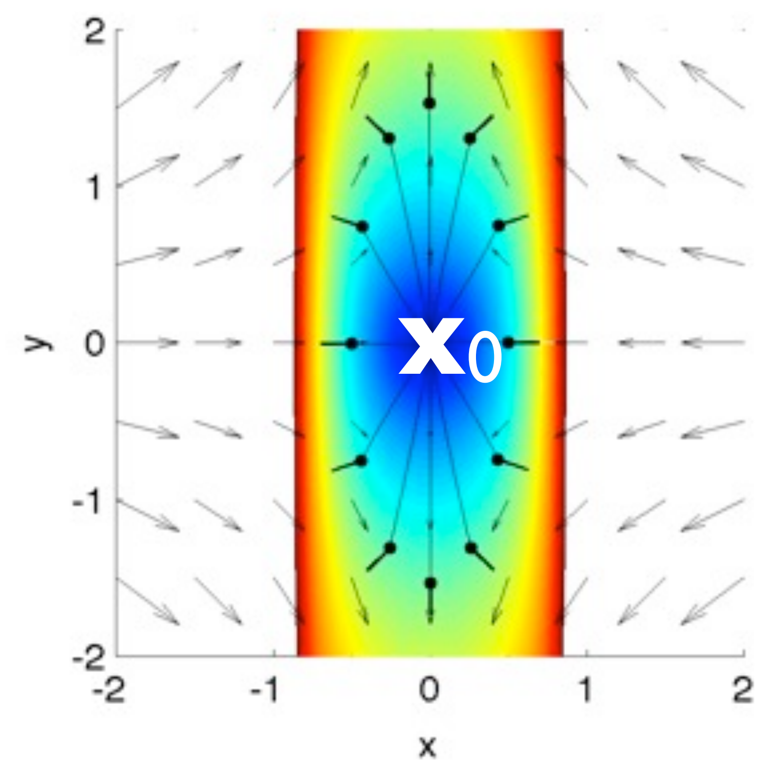
# Effect of simple LTI flows

Elliptic:
swim outward.

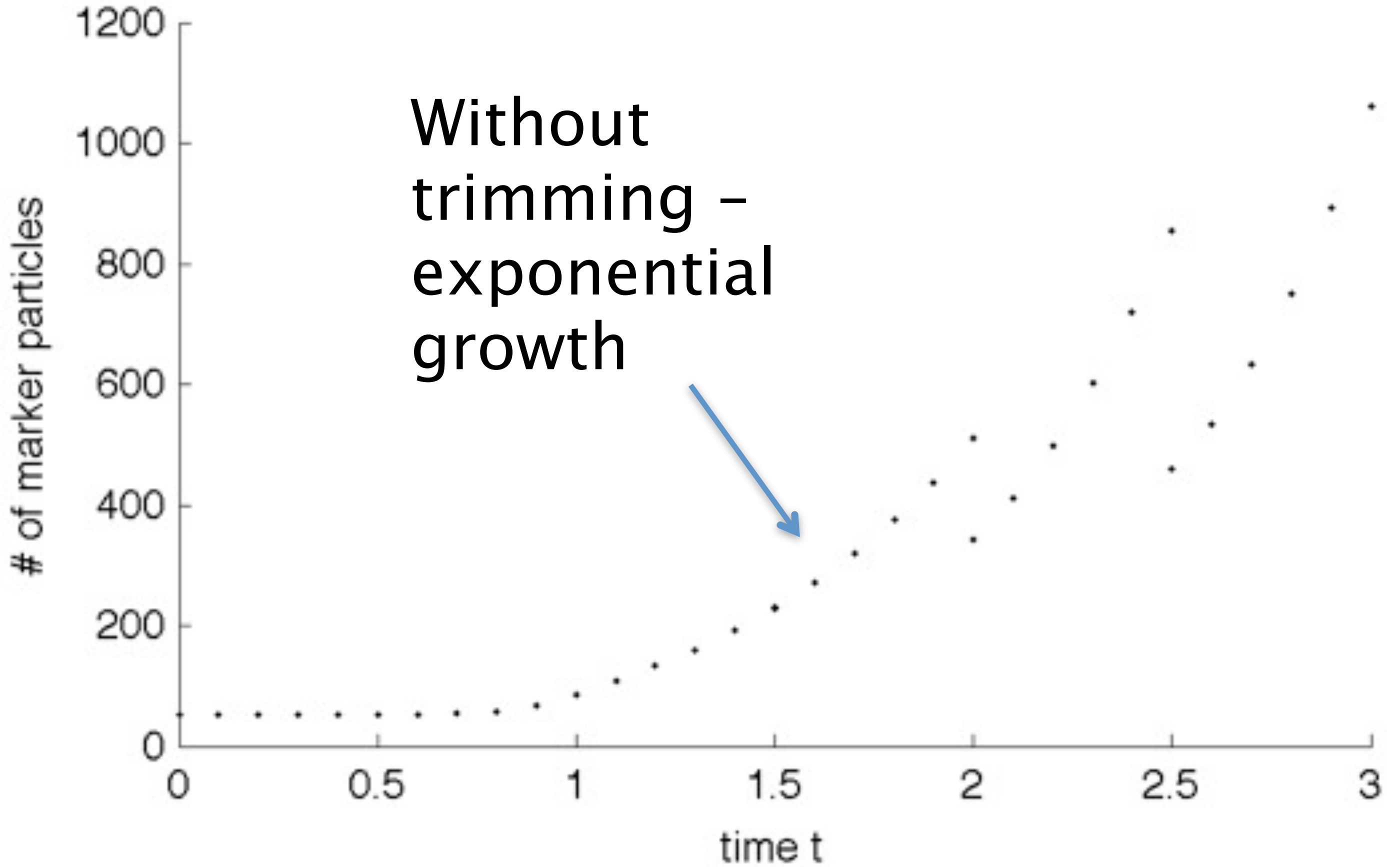


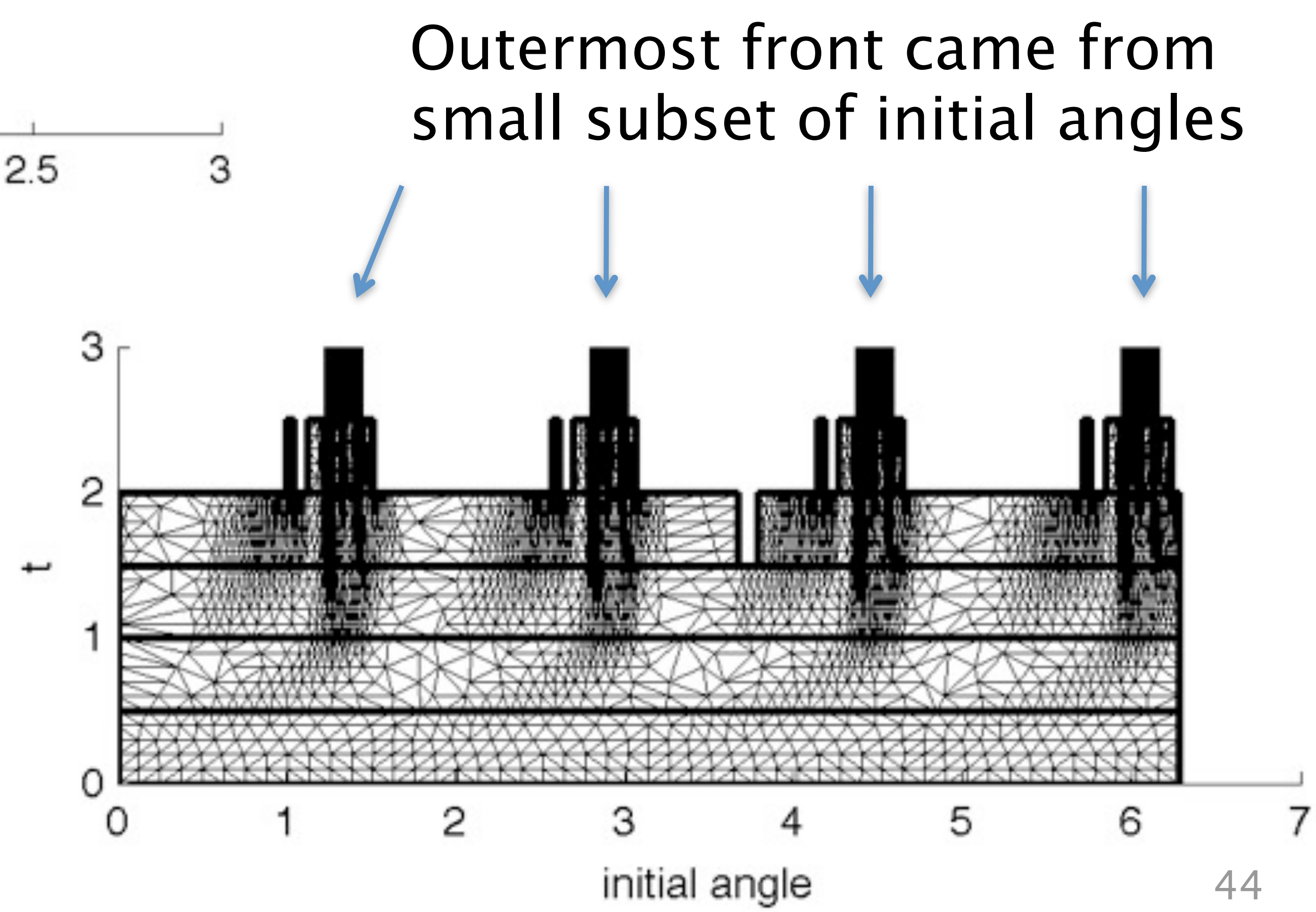

Hyperbolic:
use structure.



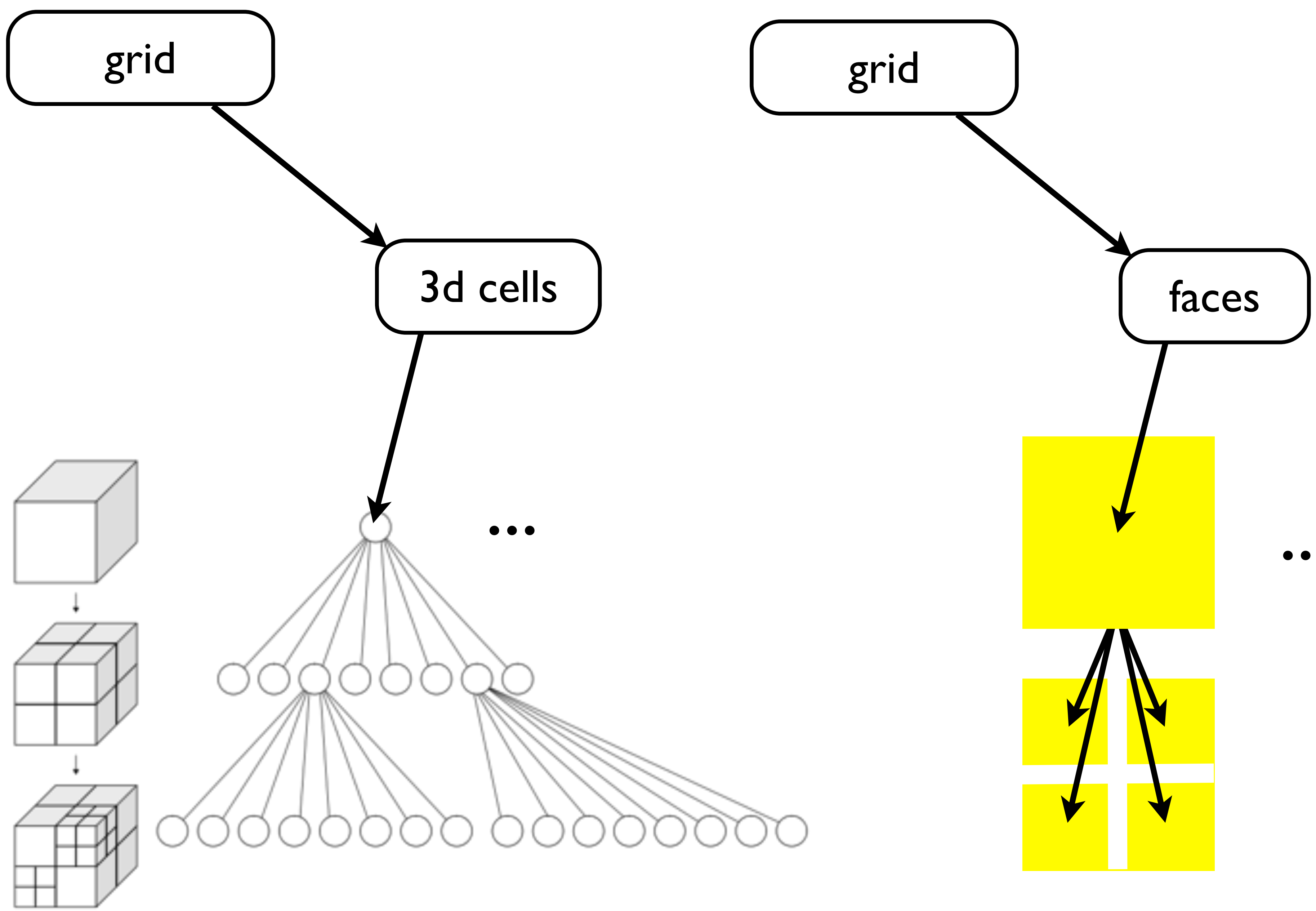



# Many locally optimal trajectories trimmed away



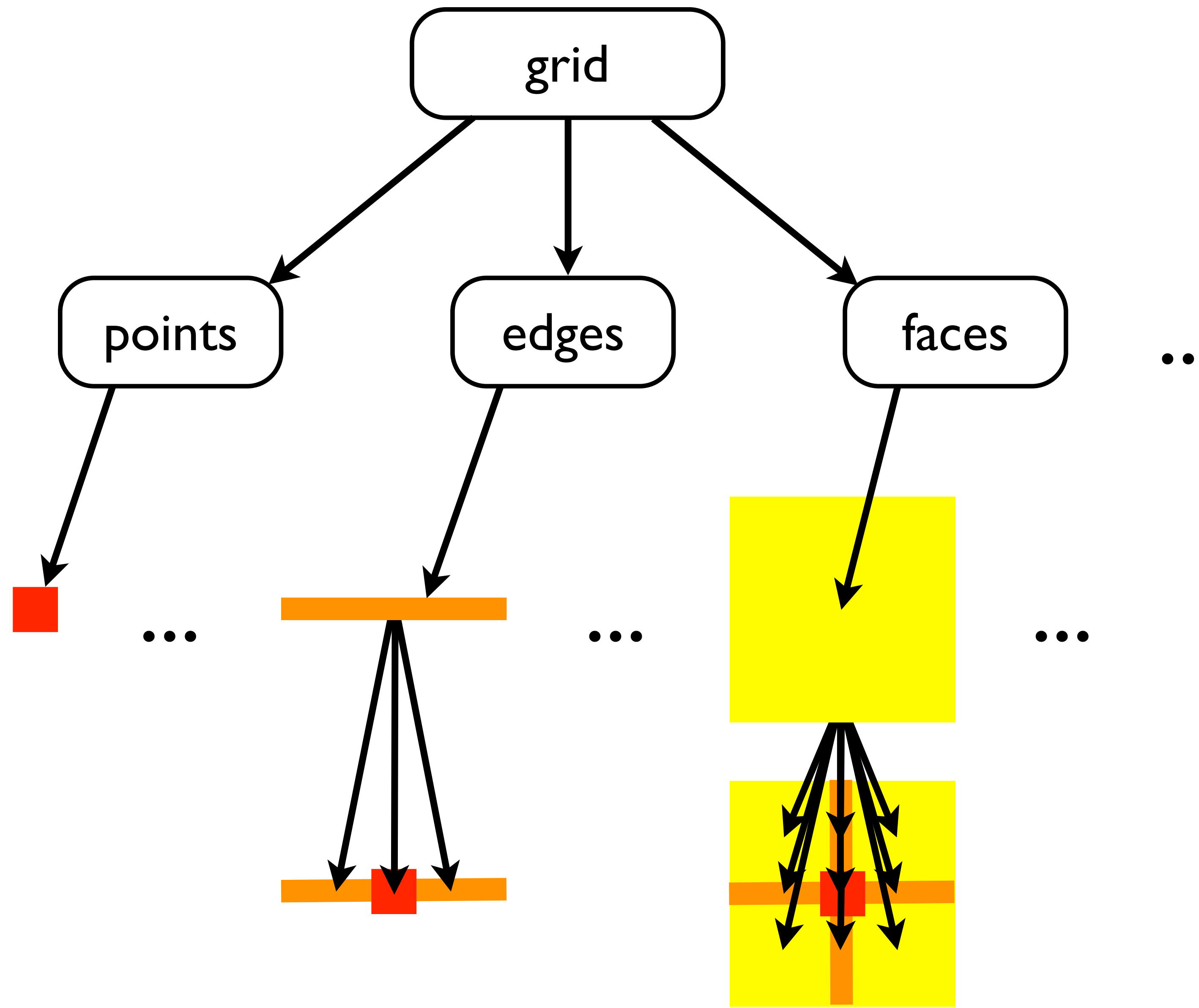



# Adaptive grid structures:
# basic octree, quadtree



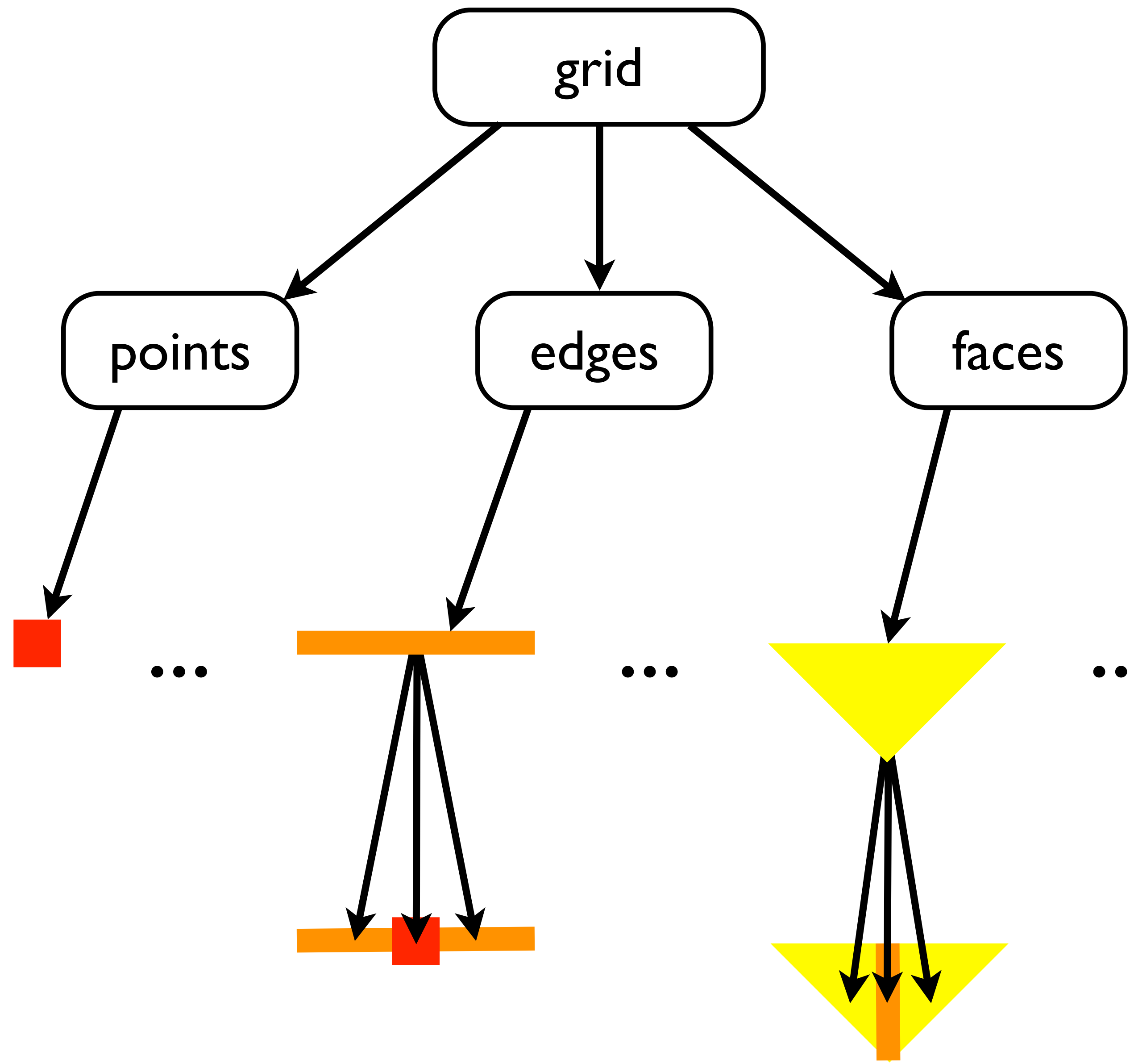
# Adaptive grid structures: general "nd-tree"

- Grid is kept *graded*, via recursive refinement of neighbors.
- For visualization and computation of level sets: *Delaunay triangulation/ tetrahedralization*

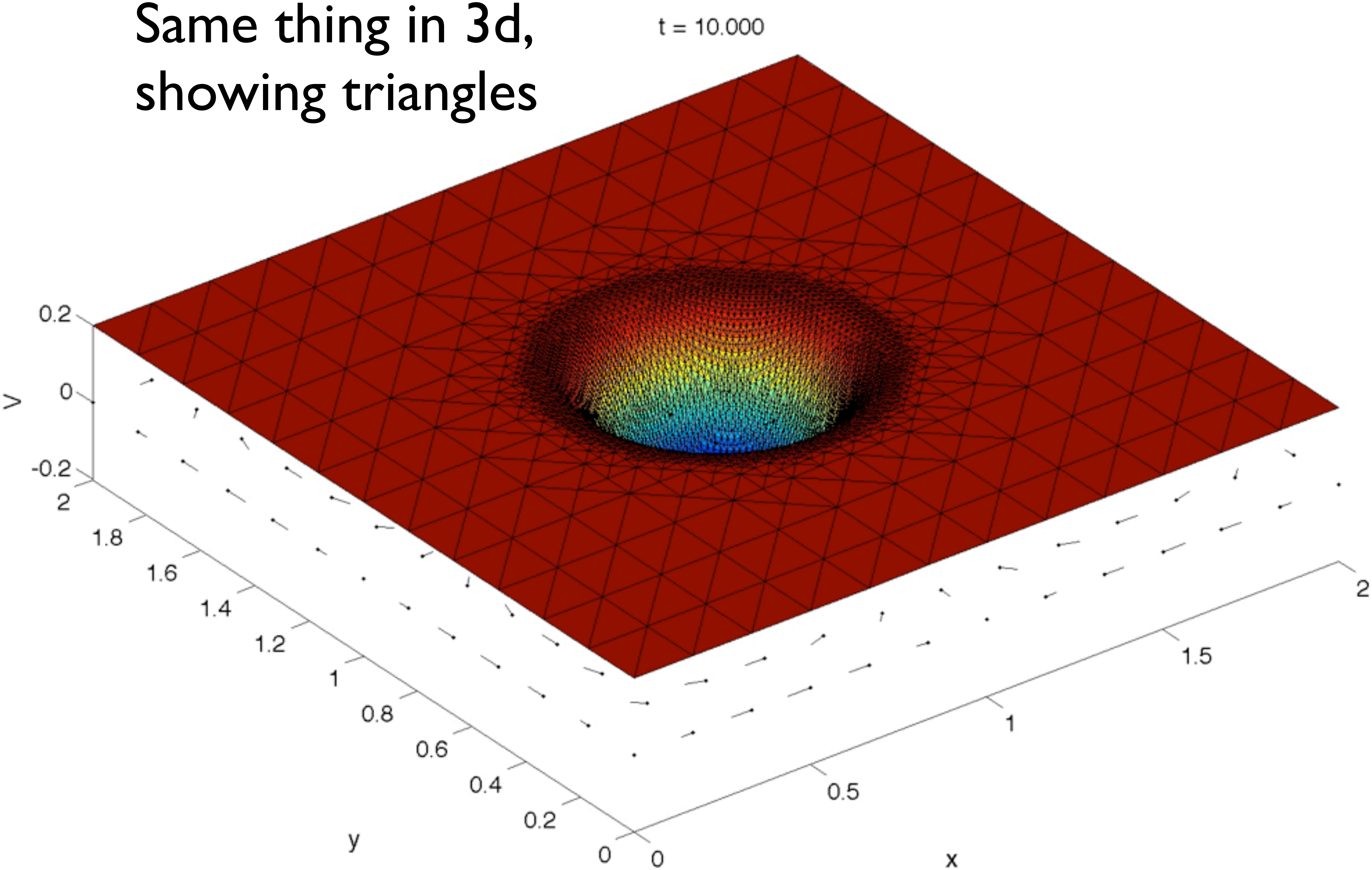


# Adaptive grid structures: triangles

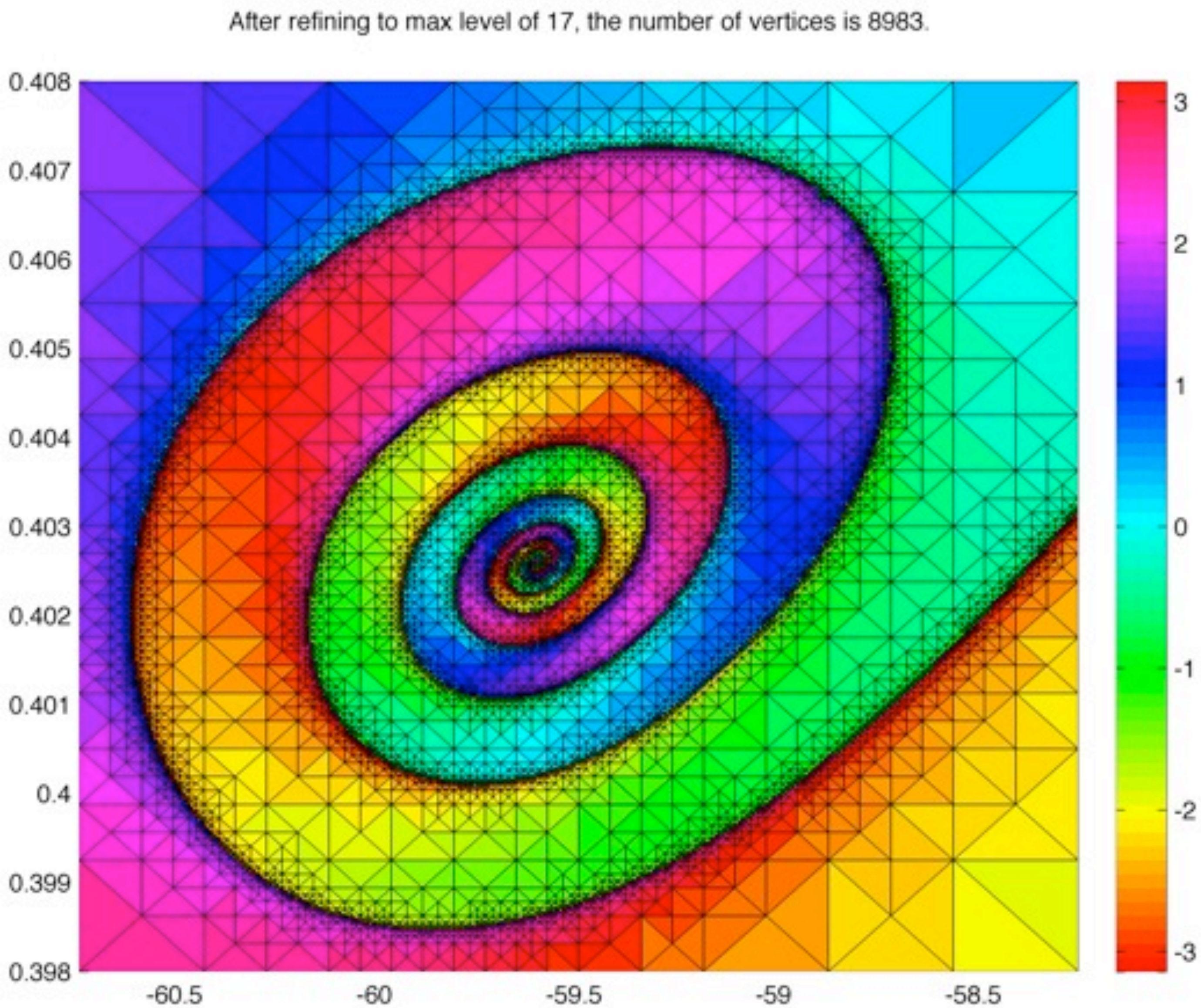
- Grid refinement initiated by edge refinement.
- Edge refinement based on error estimate.
- Grid kept *graded* by simply refining hypoteneuse first.
- Fast binary triangle search



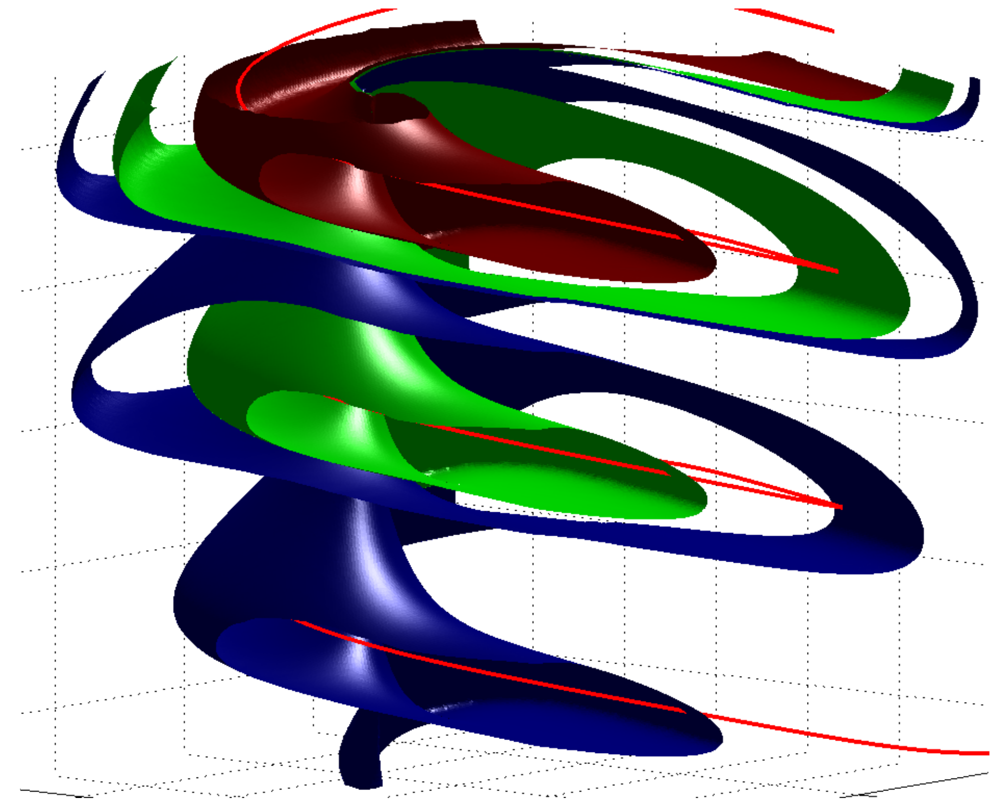
# Same thing in 3d, showing triangles



# Another application: neuron phase dynamics

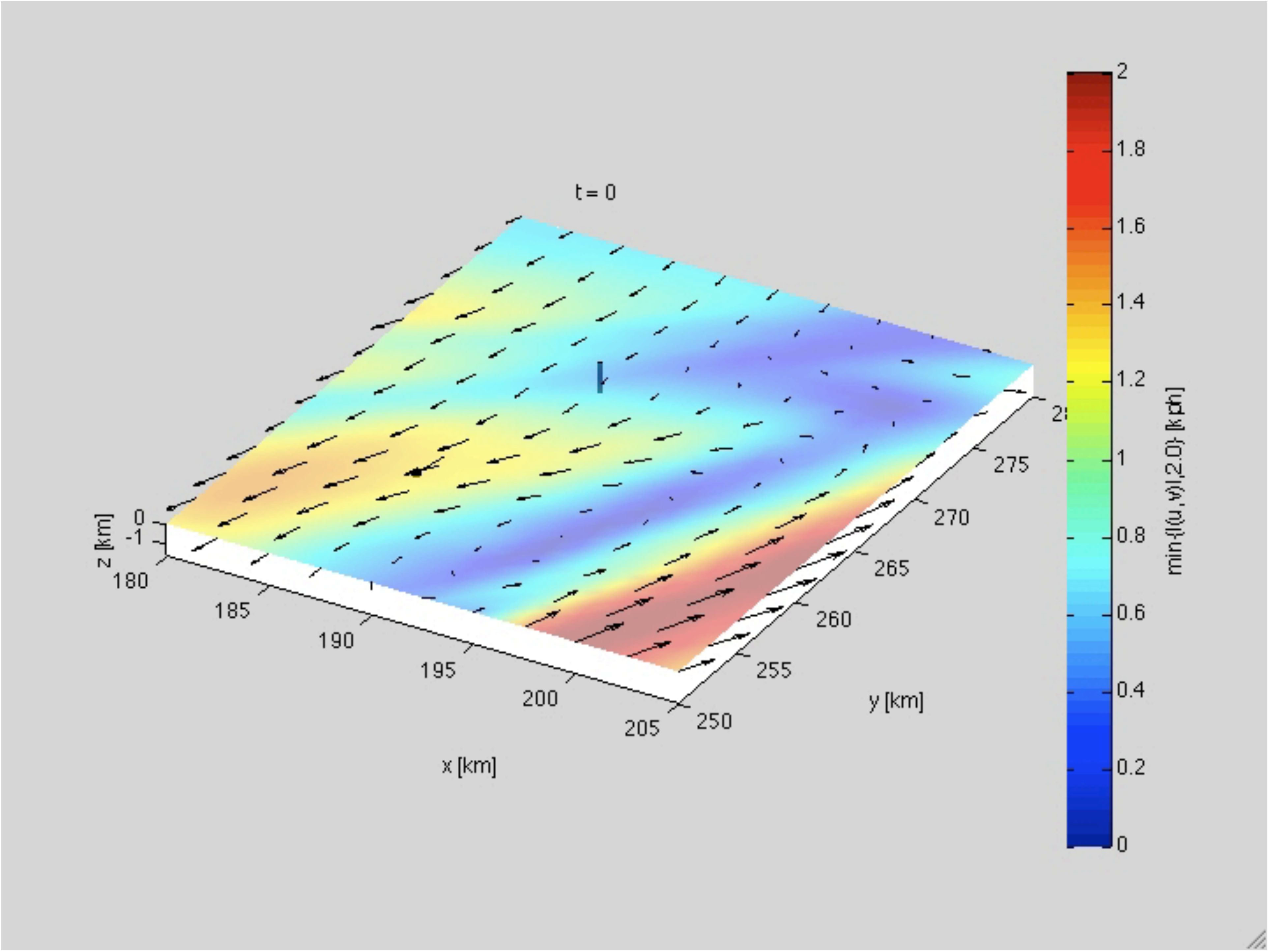



Phase function near unstable fixed point of 2D Hodgkin–Huxley neuron model, via **quadtree (nd–tree)**

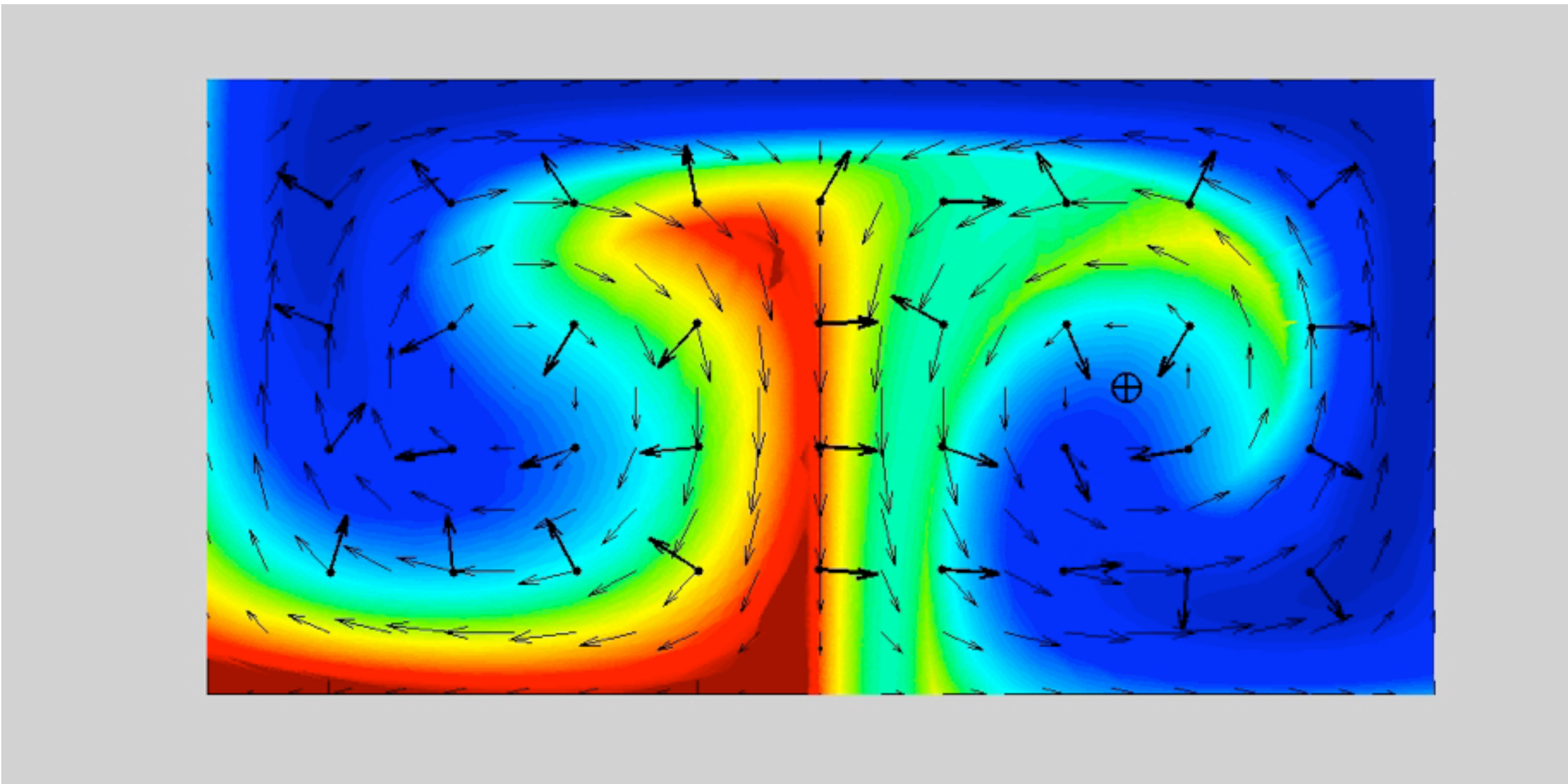


3 Isochrons (level sets of phase function) near limit cycle (red) of Hindmarsh – Rose 3D bursting neuron model, via **octree (nd–tree)** adaptive mesh, **Delaunay tetrahedralization**, and custom **marching tetrahedra** isosurface function

# 3d, time-varying flow

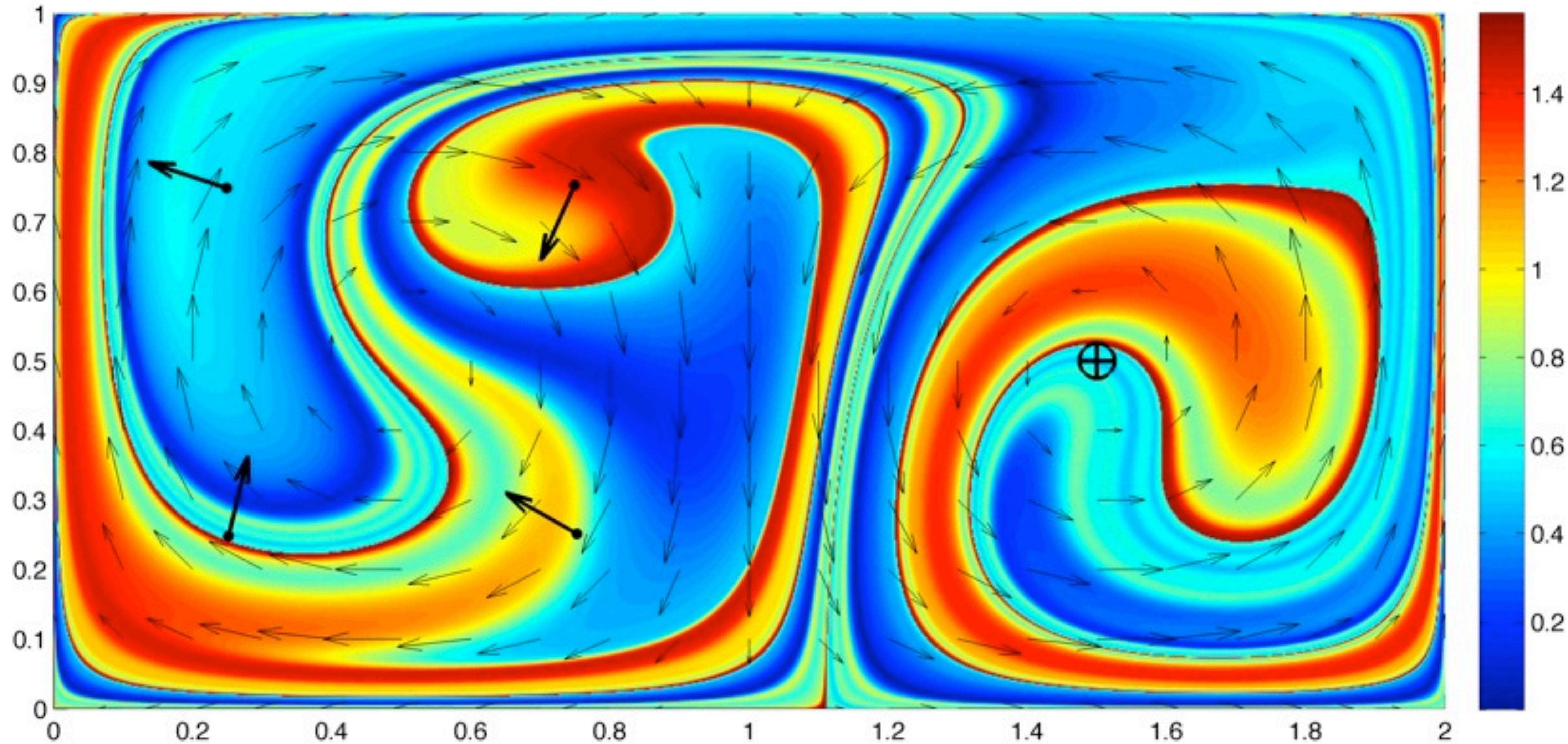


# Problem 3 – Minimum energy: time-varying double gyre

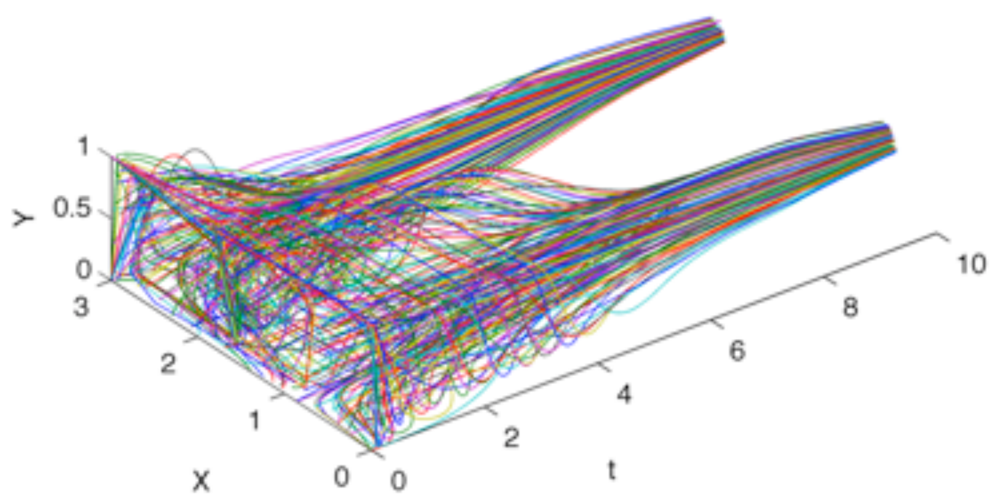


$W = 1.5$, $h(\mathbf{x}) := 0.5|\mathbf{x} - \mathbf{x}_f|^2$, $V_{max} = 0.05$
(max speed that results is about 0.1)

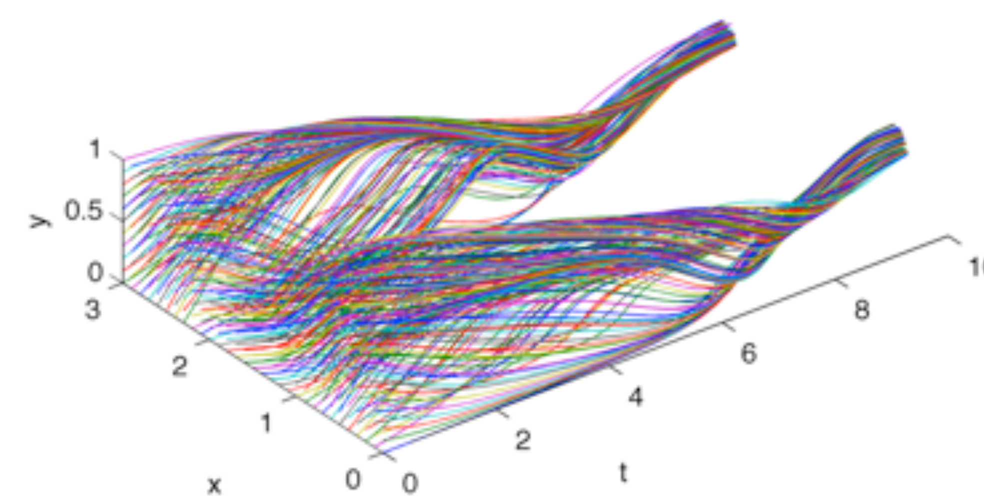
# Longer times (T = 20): even steeper gradients



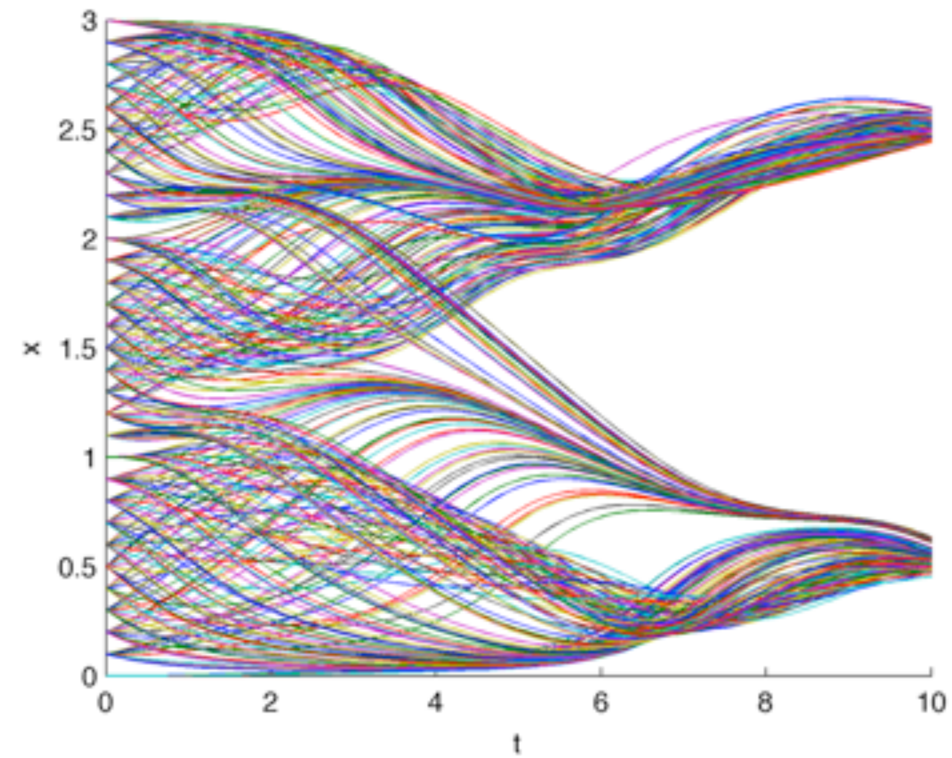

… challenging robust feedback problem?



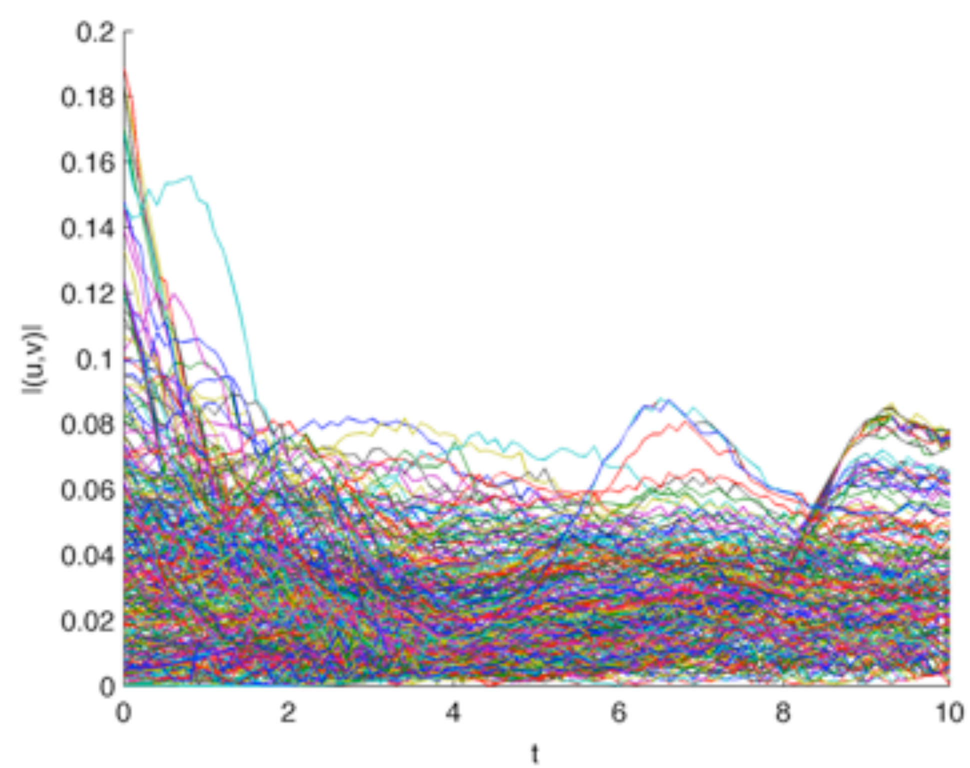
y
1
0.5
0
3
2
1
0
X
0
2
4
6
8
10
t

y
1
0.5
0
3
2
1
0
x
0
2
4
6
8
10
t



x
0
0.5
1
1.5
2
2.5
3
0
2
4
6
8
10
t

|(u,v)|
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
0
2
4
6
8
10
t