CSCAMM, UMd, April 2007

Some examples of equation-free computation

An *Input-Output* Approach to Multiscale Computation

I. G. Kevrekidis **-and several collaborators-**

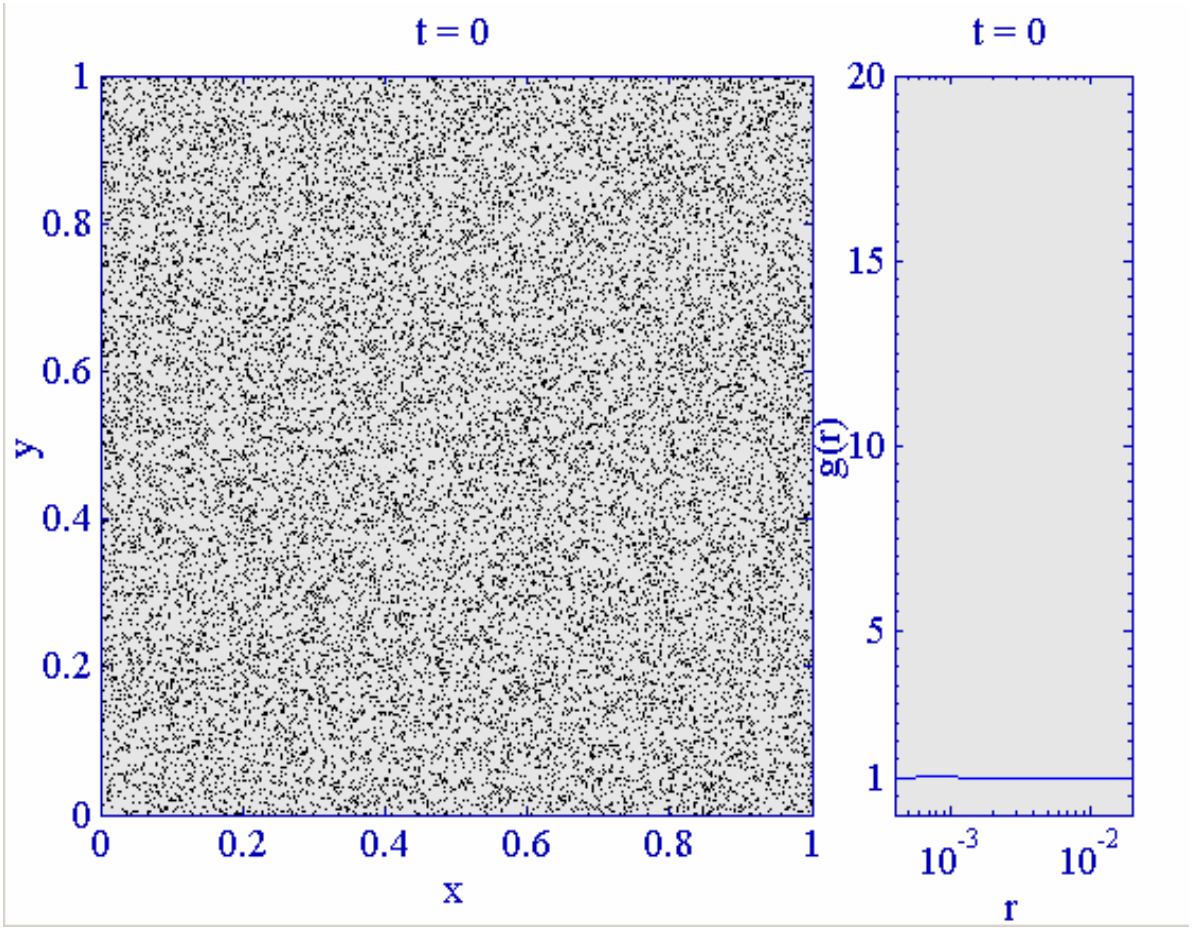
Department of Chemical Engineering & PACM
Princeton University, Princeton, NJ 08544

SIAM– July, 2004

Clustering and stirring in a plankton model

Young, Roberts and Stuhne, *Nature* 2001

Dynamics of System with convection



Simulation Method

- Random (equal) birth and death, probability: $\lambda = \mu$.
- Brownian motion.
- Advective stirring. (φ, θ are random phases)

$$x'_k = x_k + \delta x_k(t); \left\langle \delta^2 x_k \right\rangle = 2\tau D$$

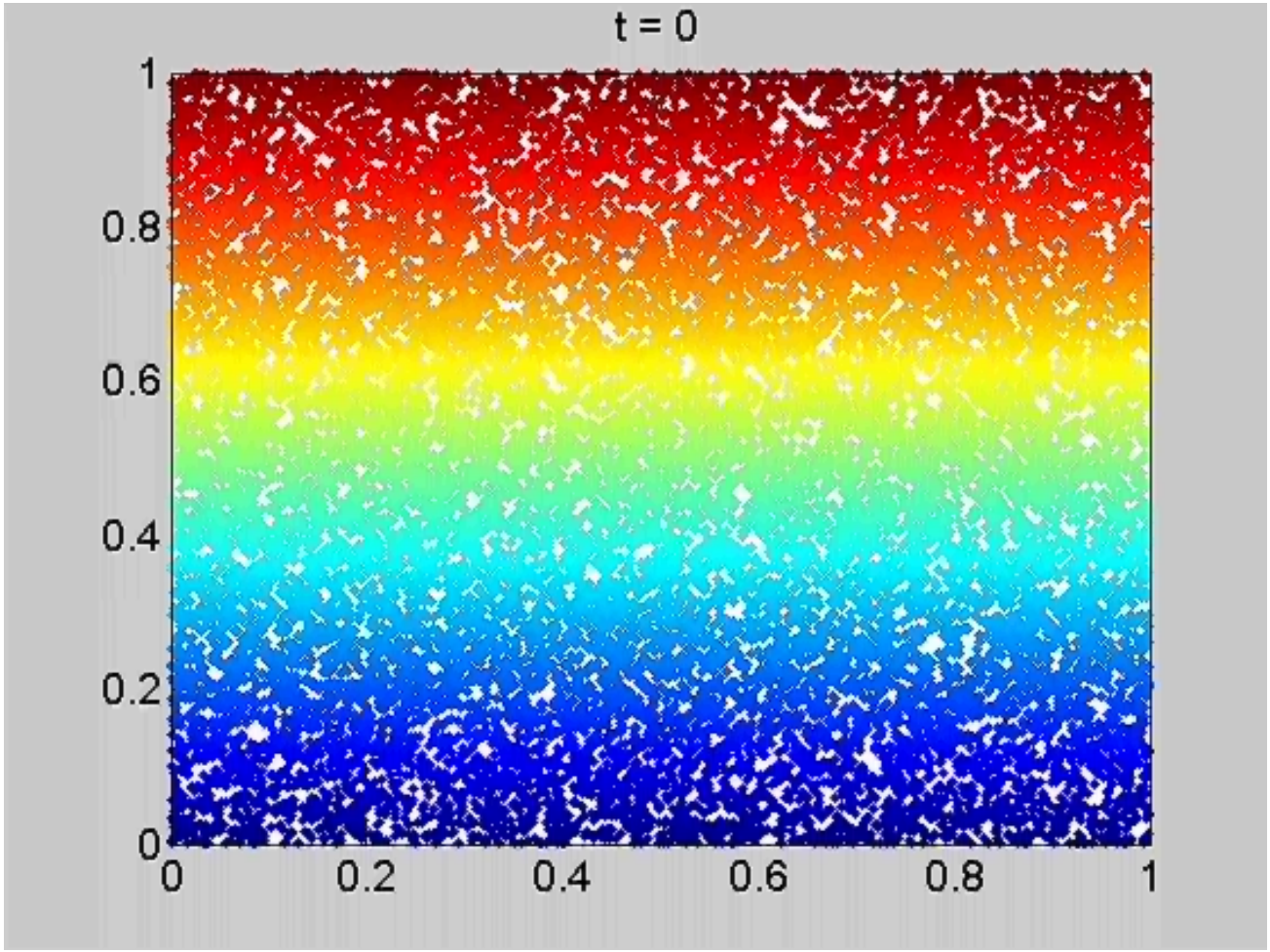
$$x_k(t+\tau) = x'_k(t) + U\,\tau/2 \cos[k y'_k(t) + \varphi(t)]$$

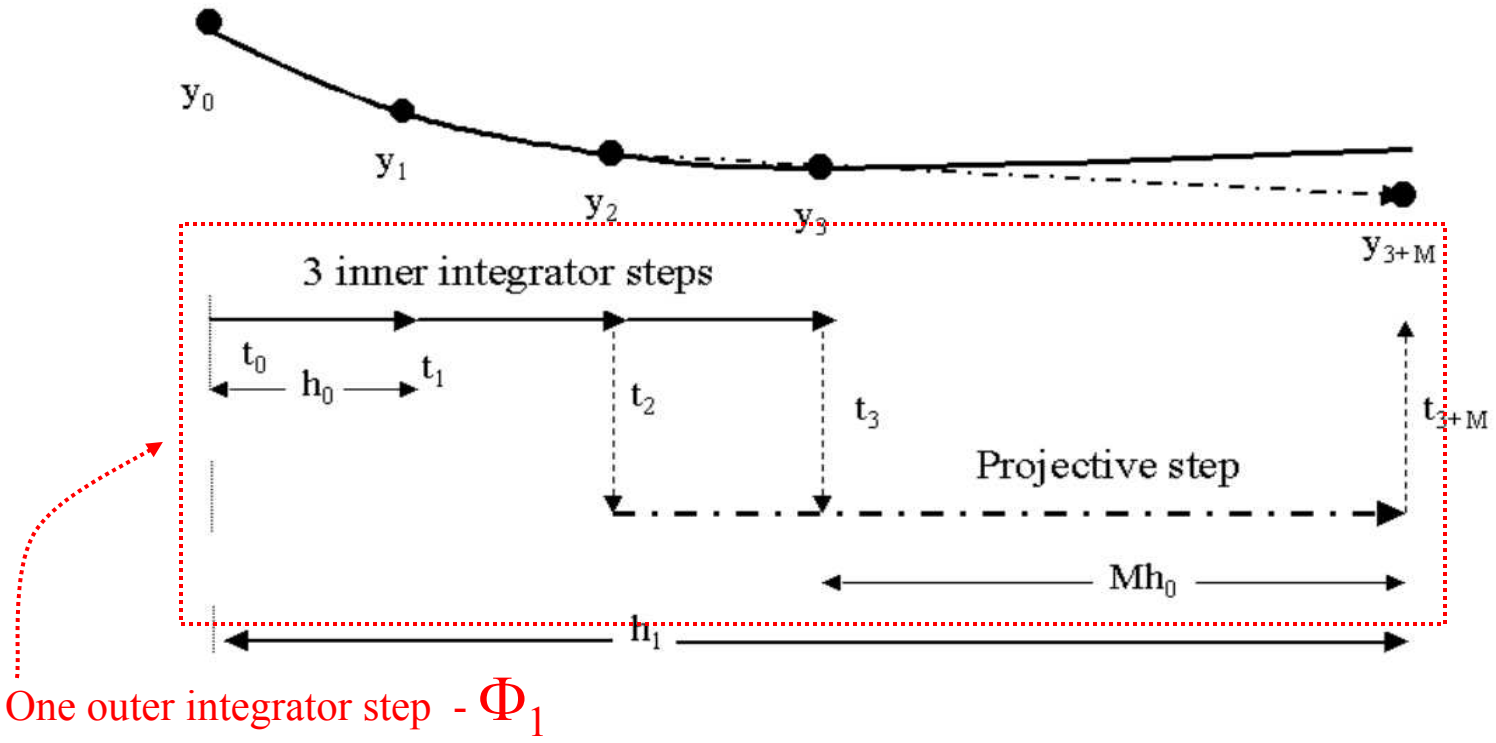
$$y_k(t+\tau) = y'_k(t) + U\,\tau/2 \cos[k x'_k(t) + \theta(t)]$$

$$G_t = 2D\frac{1}{r}(rG_r)_r + 2(\lambda-\mu)G + \gamma\frac{1}{r}(r^3 G_r)_r + 2\lambda C\delta(\mathbf{r})$$

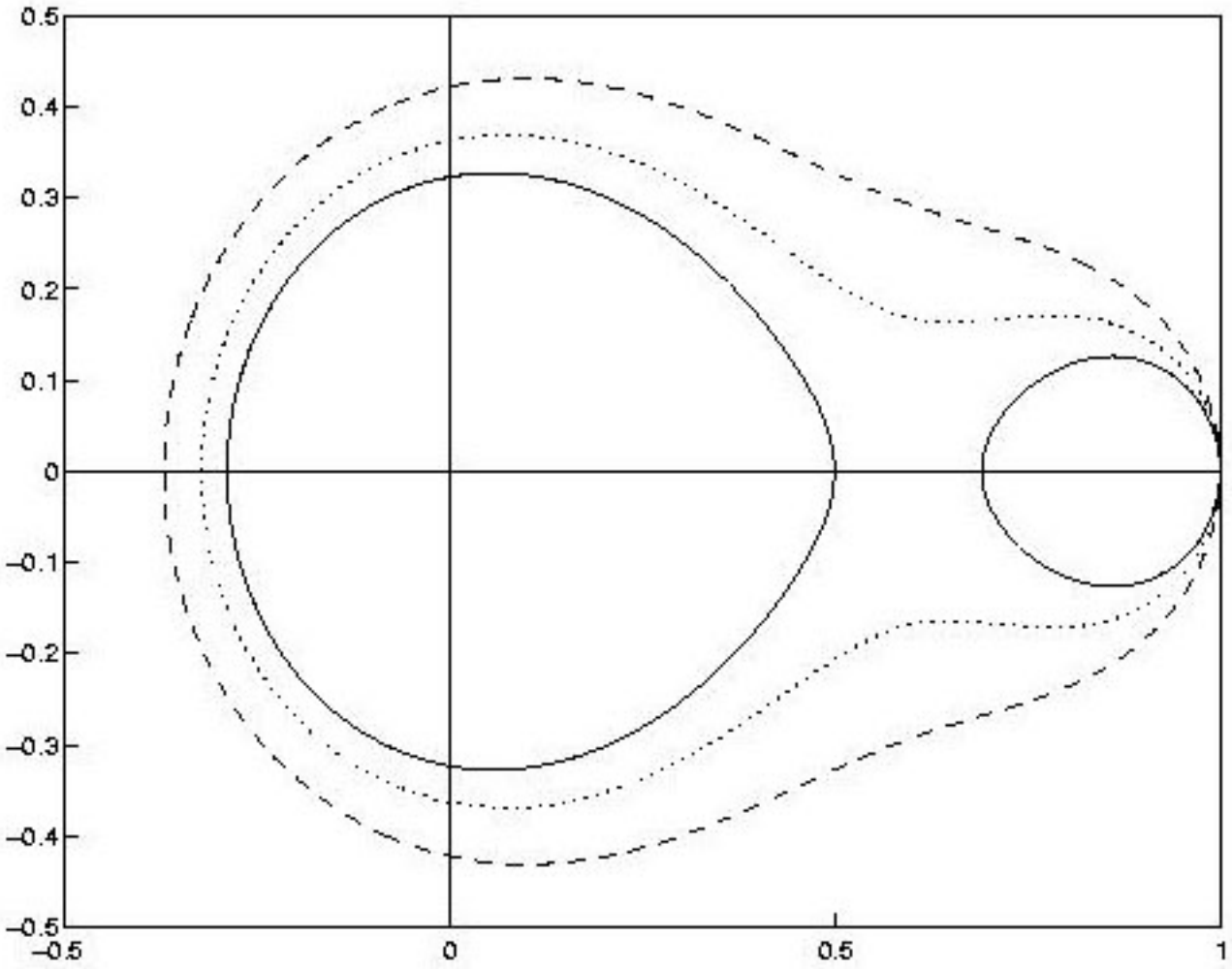
- IC: 20000 particles randomly placed in 1*1 box

Stirring by a random field (color = y)

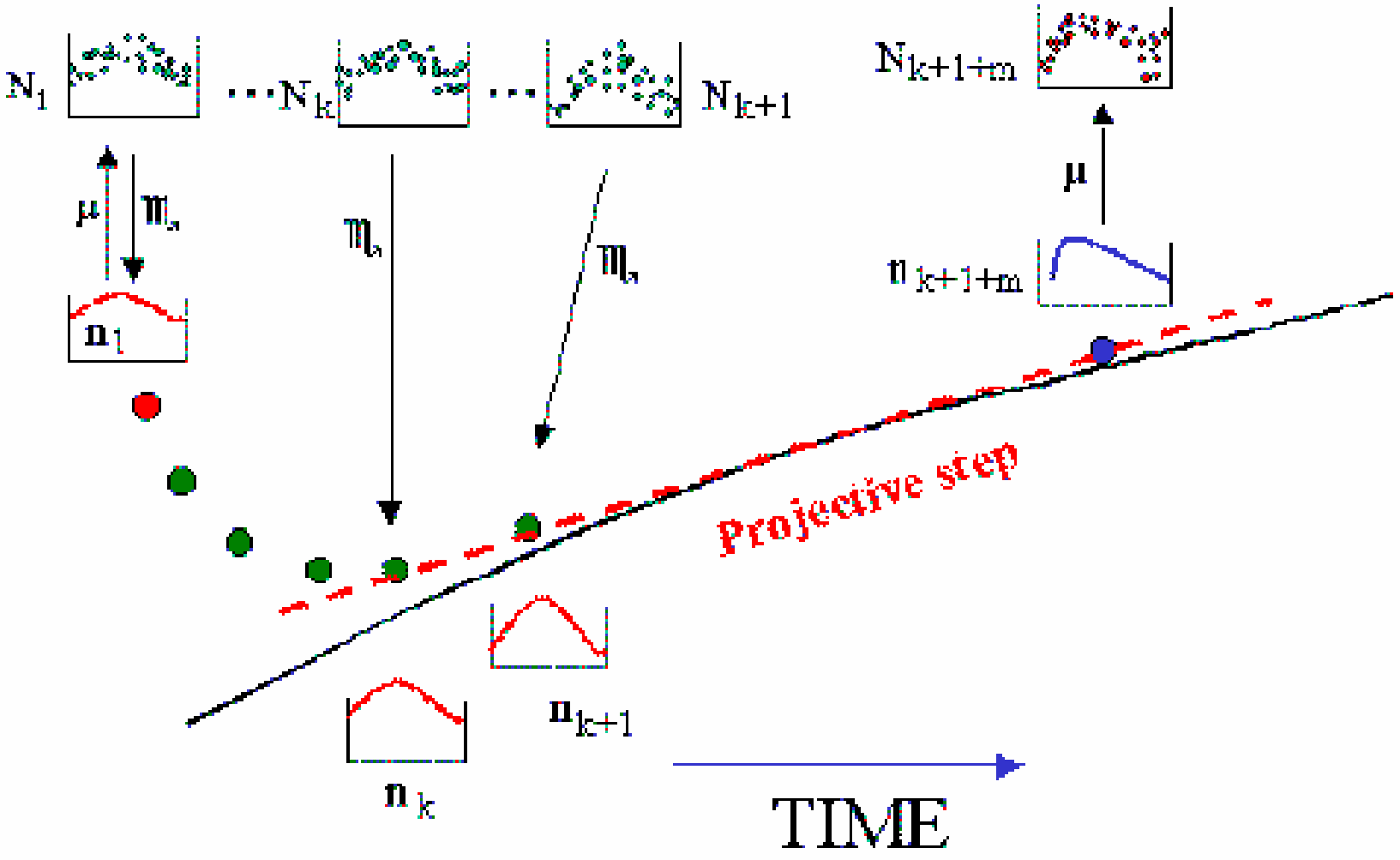




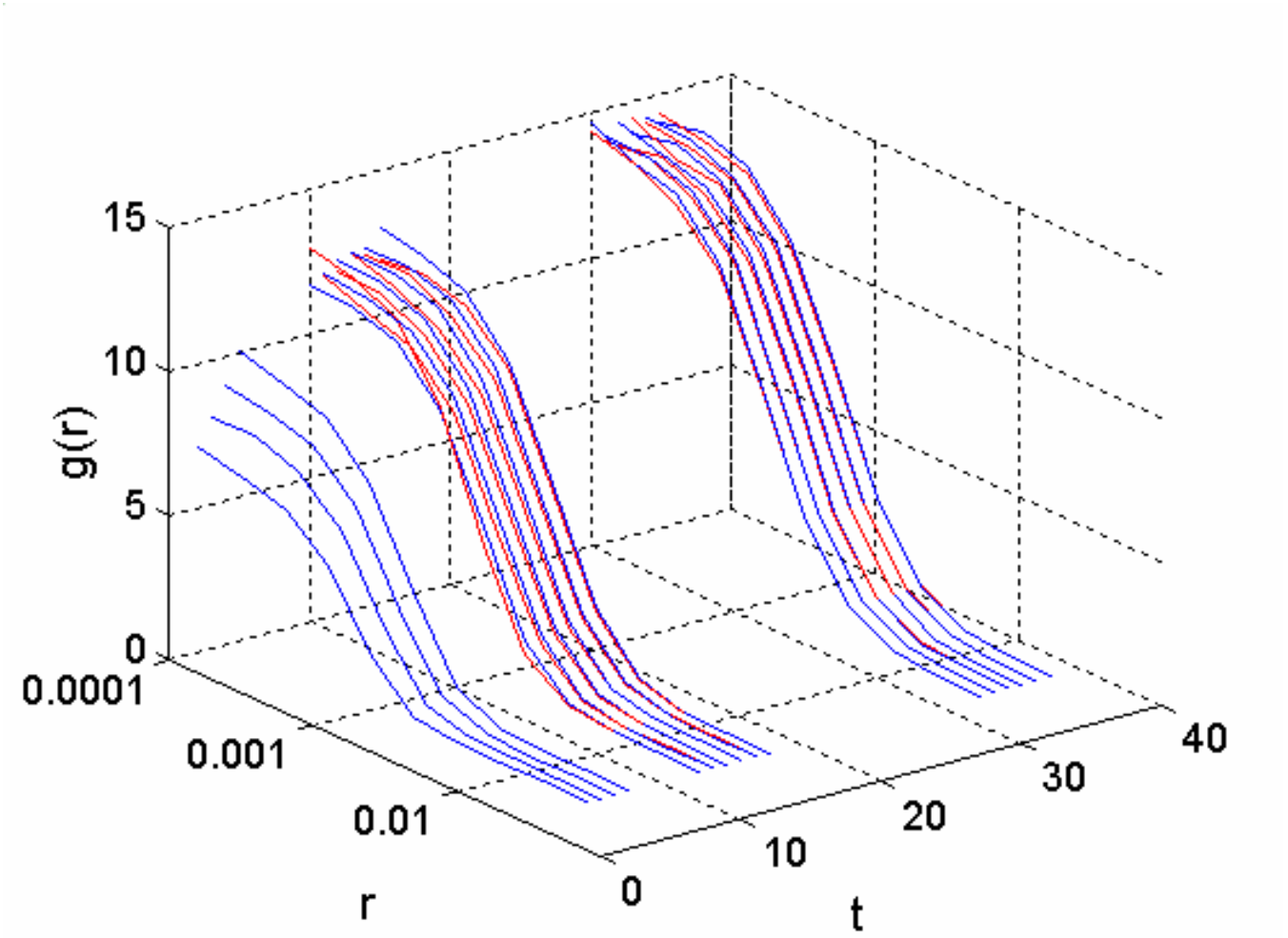
Projective Forward Euler Method - linear fit to last two points



k=2, M=5, 7, and 9


N1
Nk
Nk+1
Nk+1+m
μ
μ
nk+1+m
n1
nk
nk+1
Projective step
TIME

Projective Integration: From t=2,3,4,5 to 10



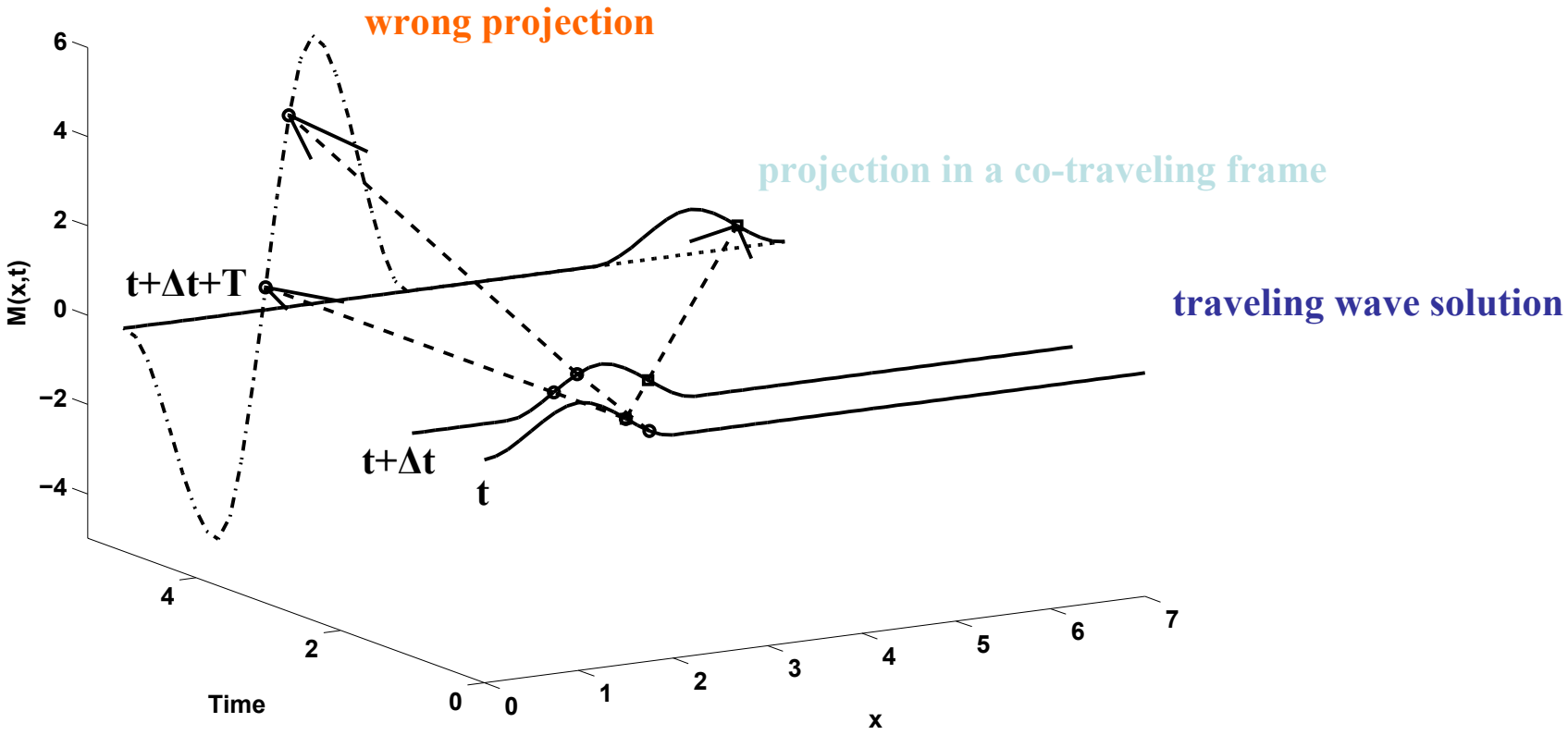
RESTRICTION - a *many-one* mapping from a high-dimensional description (such as a collection of particles in Monte Carlo simulations) to a low-dimensional description - such as a finite element approximation to a *distribution* of the particles.

LIFTING - a *one-many* mapping from low- to high-dimensional descriptions.

We do the step-by-step simulation in the high-dimensional description.

We do the macroscopic tasks in the low-dimensional description.

Projective Integration in a co-traveling frame



Coarse Self-Similar Solutions:

Dynamic Renormalization using Coarse Timesteppers.

An analogy: problems with traveling solutions (translational invariance)

Move along with the solution – it appears steady

Original Equation $\frac{\partial u}{\partial t} = L(\mathrm{u})$

Transformed equation $\frac{\partial v}{\partial t} = -c(\mathrm{v})\frac{\partial \mathrm{v}}{\partial \mathrm{x}} + L(\mathrm{v})$

(template-based reduction/reconstruction, Rowley and Marsden, 2000)

problems with focusing or collapsing solutions (scale invariance)

explode along with the solution -→ it appears steady

Original Equation $\frac{\partial u}{\partial t} = L(\mathrm{u})$

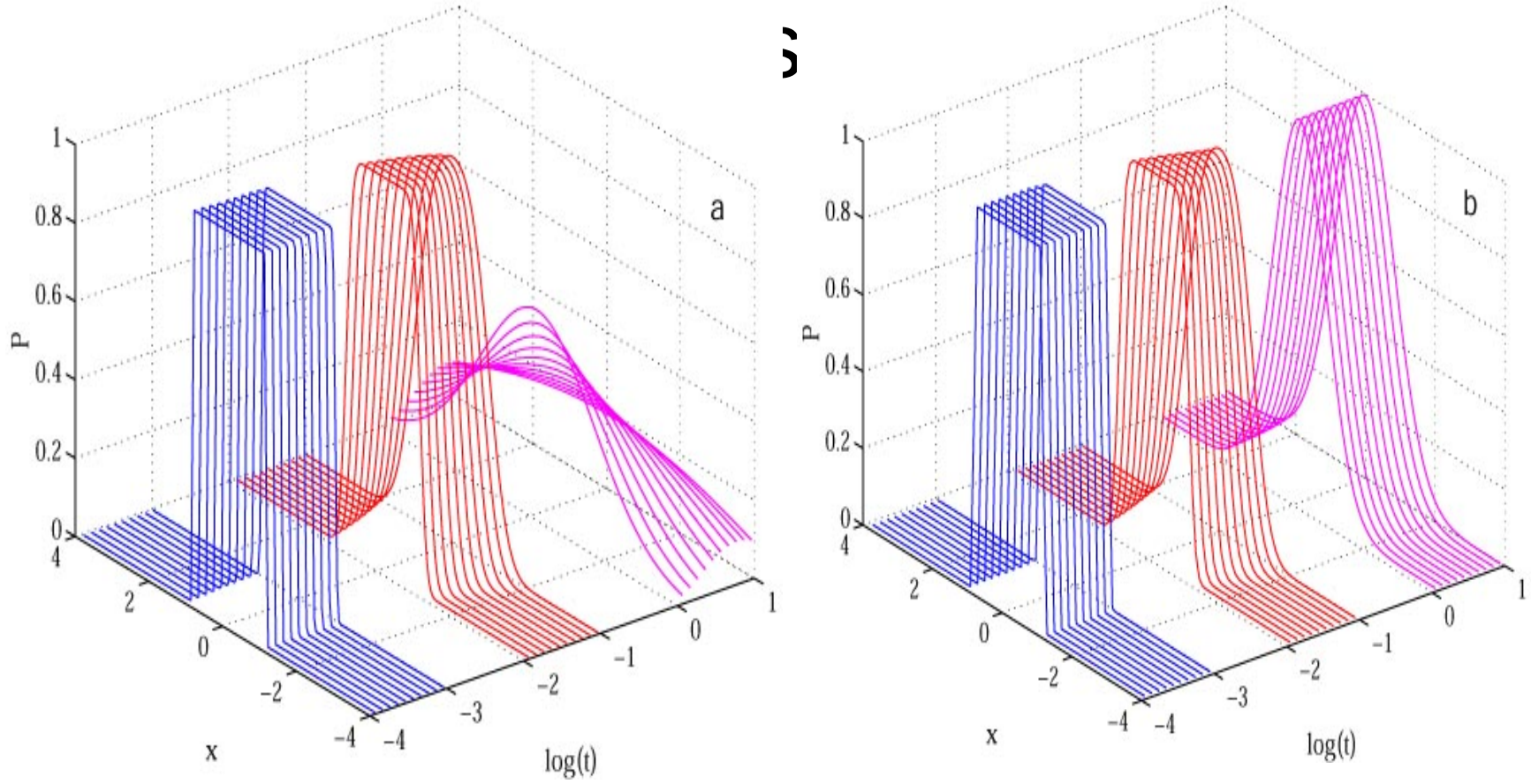
Transformed equation $\frac{\partial v}{\partial t} = -G(v)(v + \xi\frac{\partial v}{\partial \xi}) + L(v)$

Mid-80's: Lemesurier, the Sulems,, Papanicolaou…..

Templates: Aronson, Betelu and Kevrekidis, 2001
Rowley, Kevrekidis, Marsden and Lust, 2003)

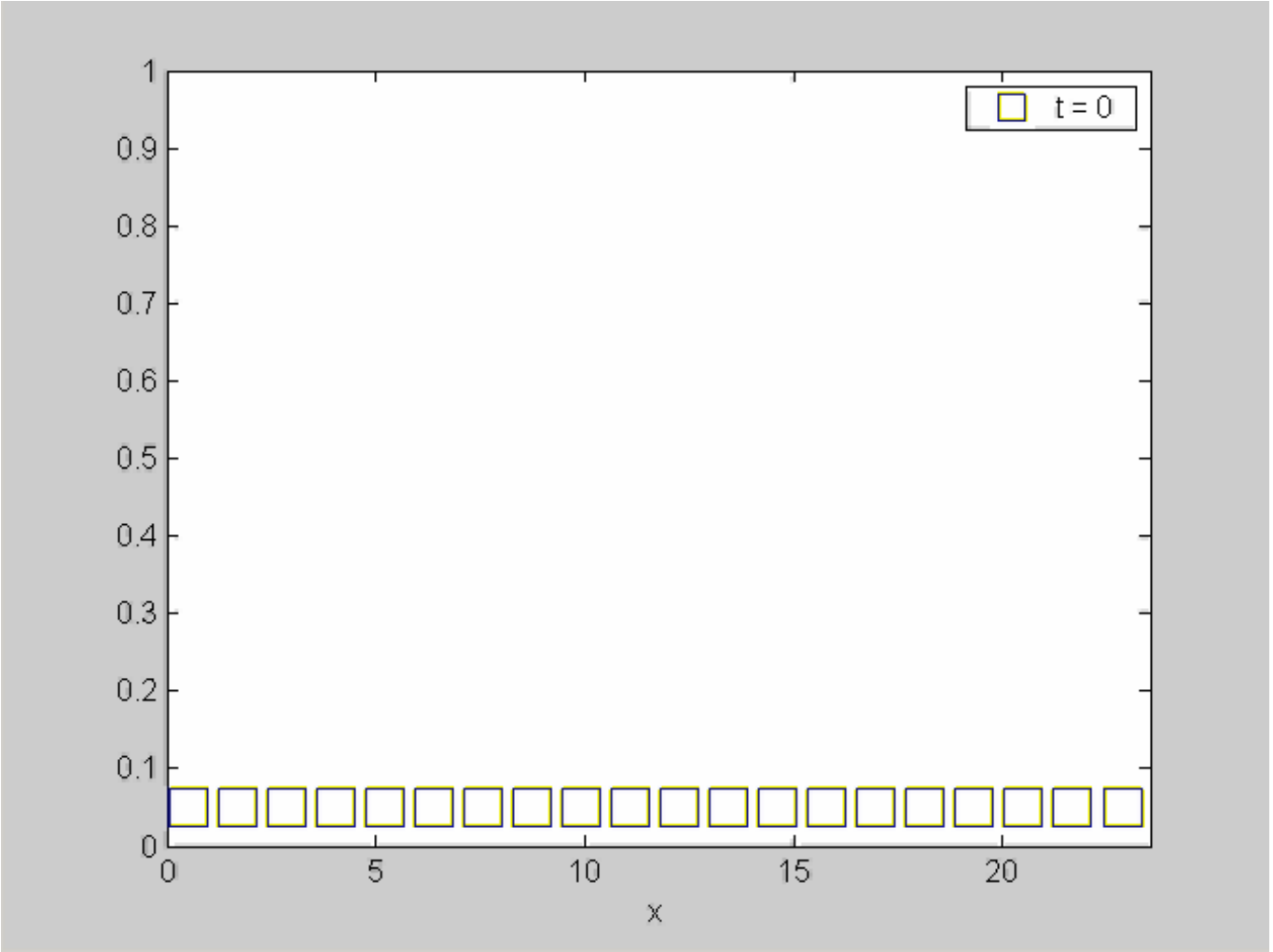
Coarse renormalization flow integration / bifurcation analysis

Time evolution of an initial rectangular density profile



Regular *Dynamically renormalized*

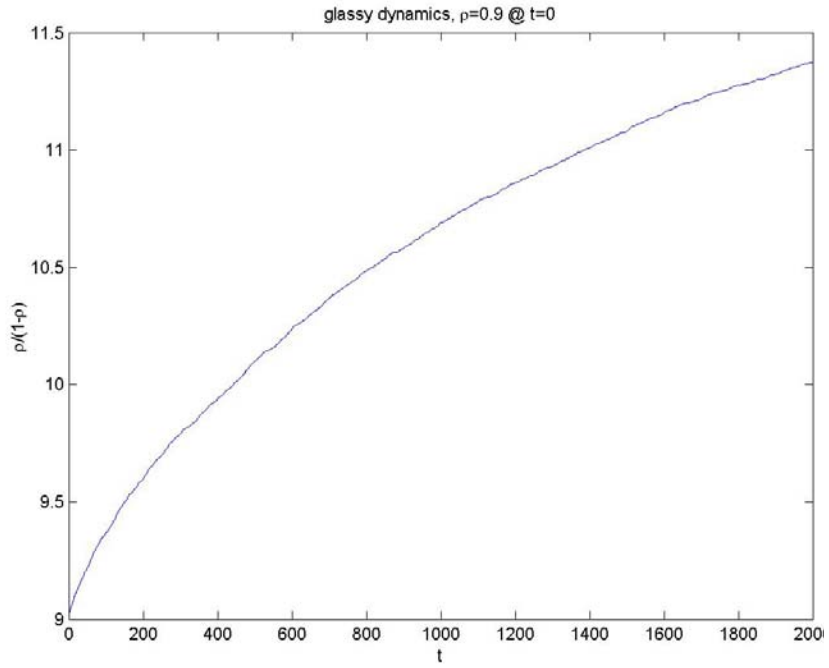
1D glass compaction model



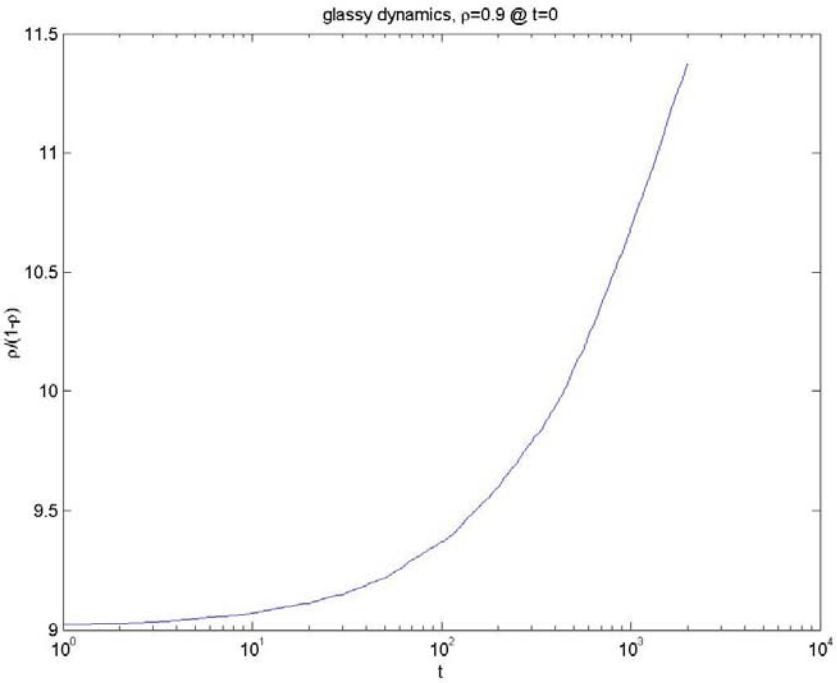
Simulation Method

- 100000 particles at given density placed in 1D simulation box with periodic boundary condition.
- Particles interact through hard-core potential.
- Monte Carlo random walk is performed in each step.
- Once a gap of unit size opened up between two adjacent particles, an additional block will deposit.

Glassy dynamics



Dynamics in linear time

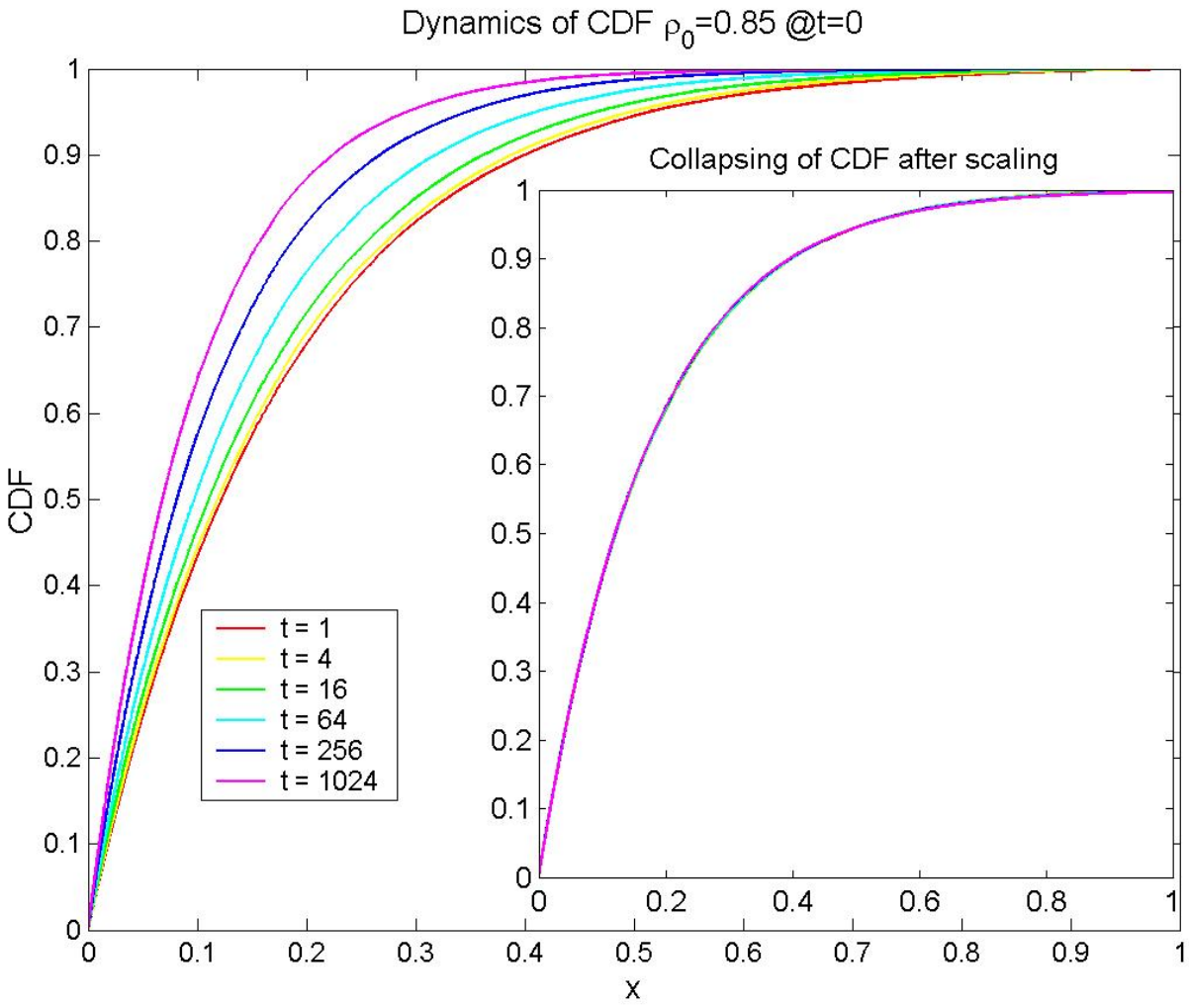


Dynamics in logarithmic time

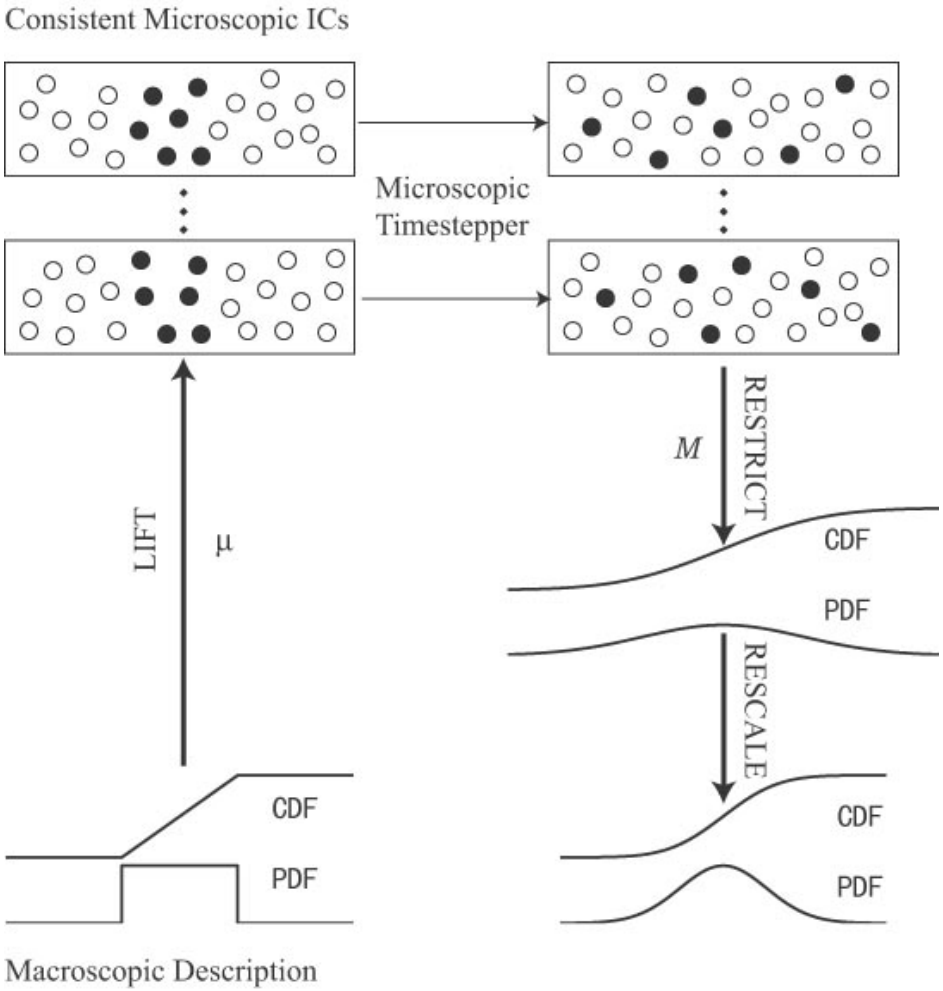
$$\left\langle \frac{\rho}{1-\rho} \right\rangle \sim \ln(t)$$

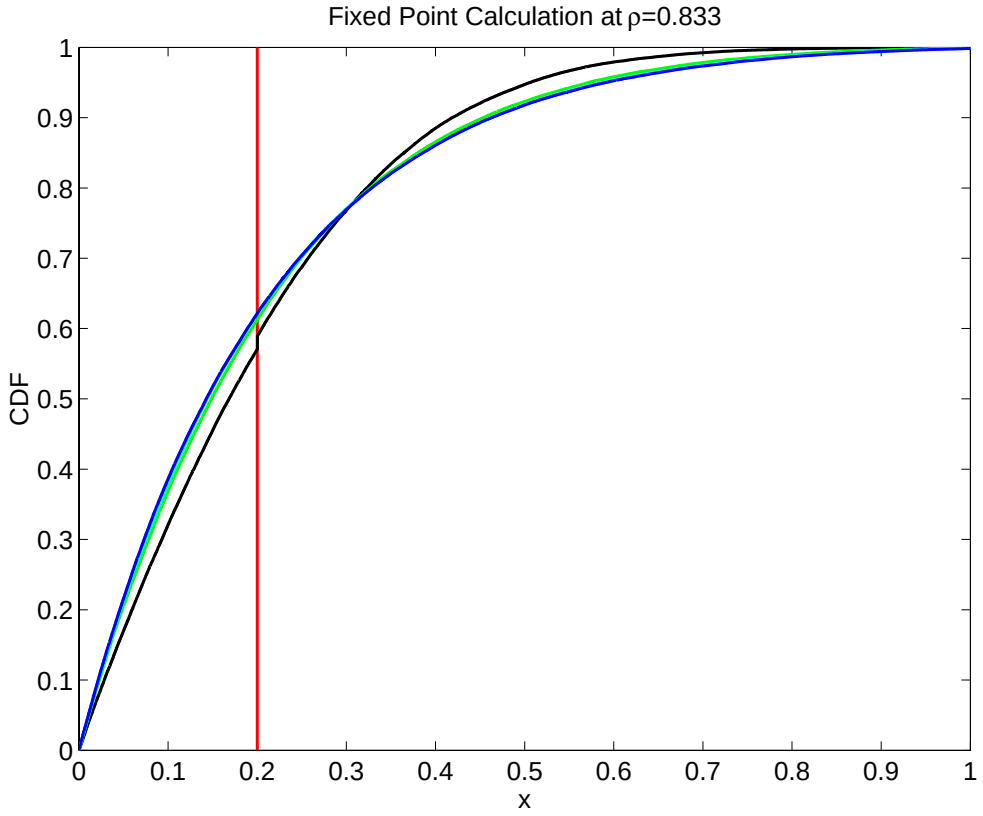
Eq. 7 Stinchcombe
PRL 88, 125701 (2002)

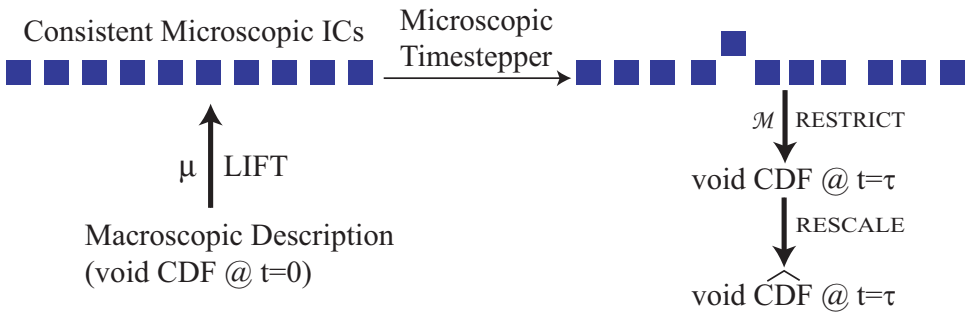
Void Distribution: Self-Similar Dynamics



Schematic view of the dynamically renormalized coarse timestepper




Fixed Point Calculation at ρ=0.833
CDF
x
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

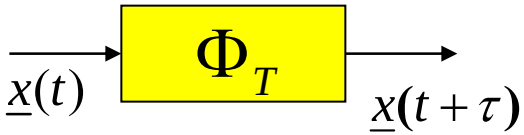

Consistent Microscopic ICs
Microscopic
Timestepper
μ LIFT
Macroscopic Description
(void CDF @ t=0)
$\mathcal{M}$ RESTRICT
void CDF @ t=τ
RESCALE
void CDF @ t=τ

THE CONCEPT: What else can I do with an integration code ?

Have equation $\dot{\underline{x}} = \underline{f}(\underline{x})$

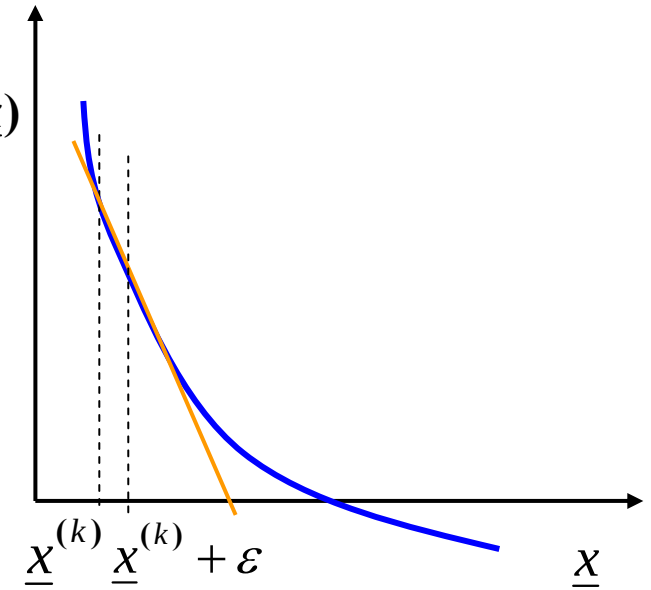
Write Simulation

Compile



Do Newton on $\underline{x} - \Phi(\underline{x}) = 0$

Do Newton



Estimate

matrix-vector product

⇩

Matrix free iterative linear algebra

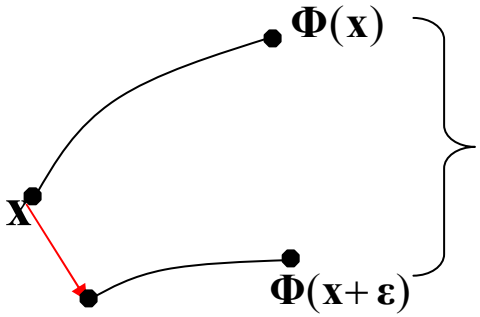
⇩

The World

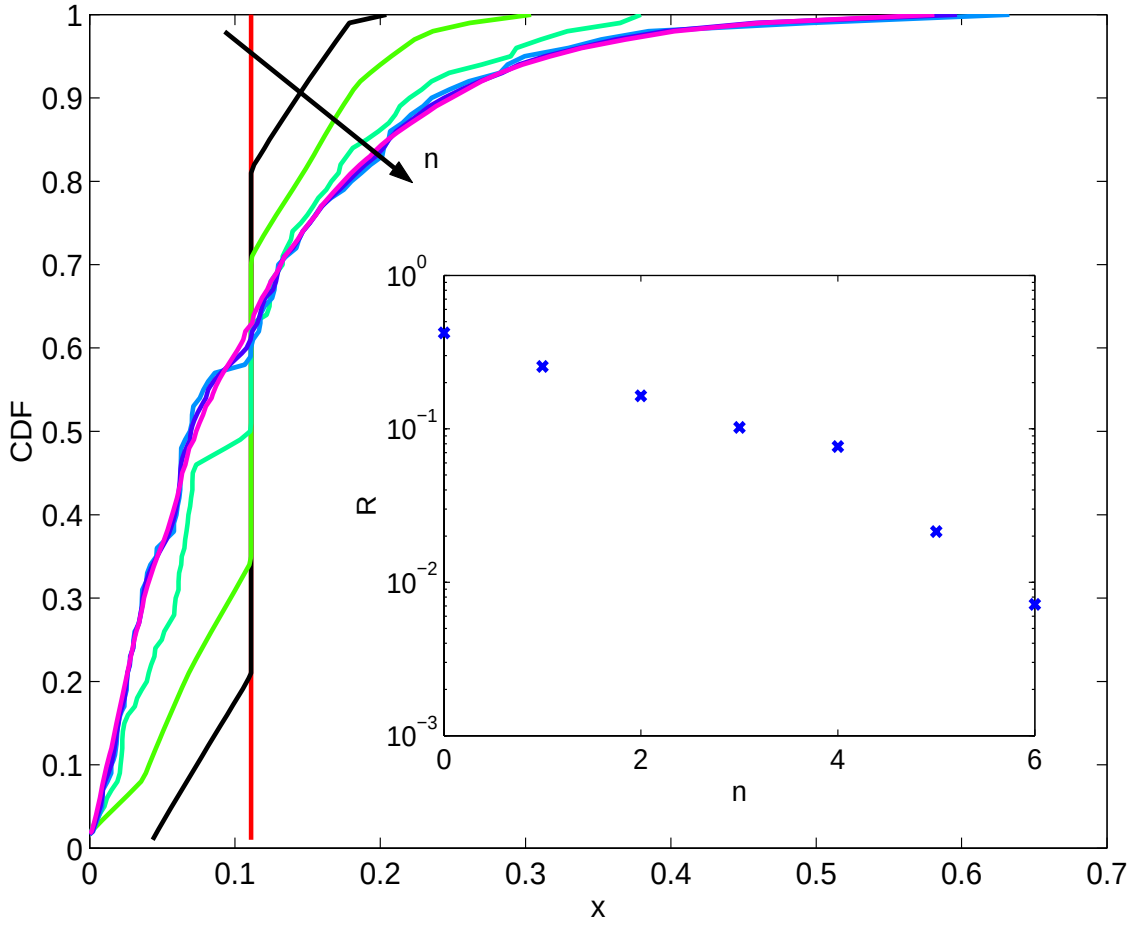
⟺

CG, GMRES Newton-Krylov

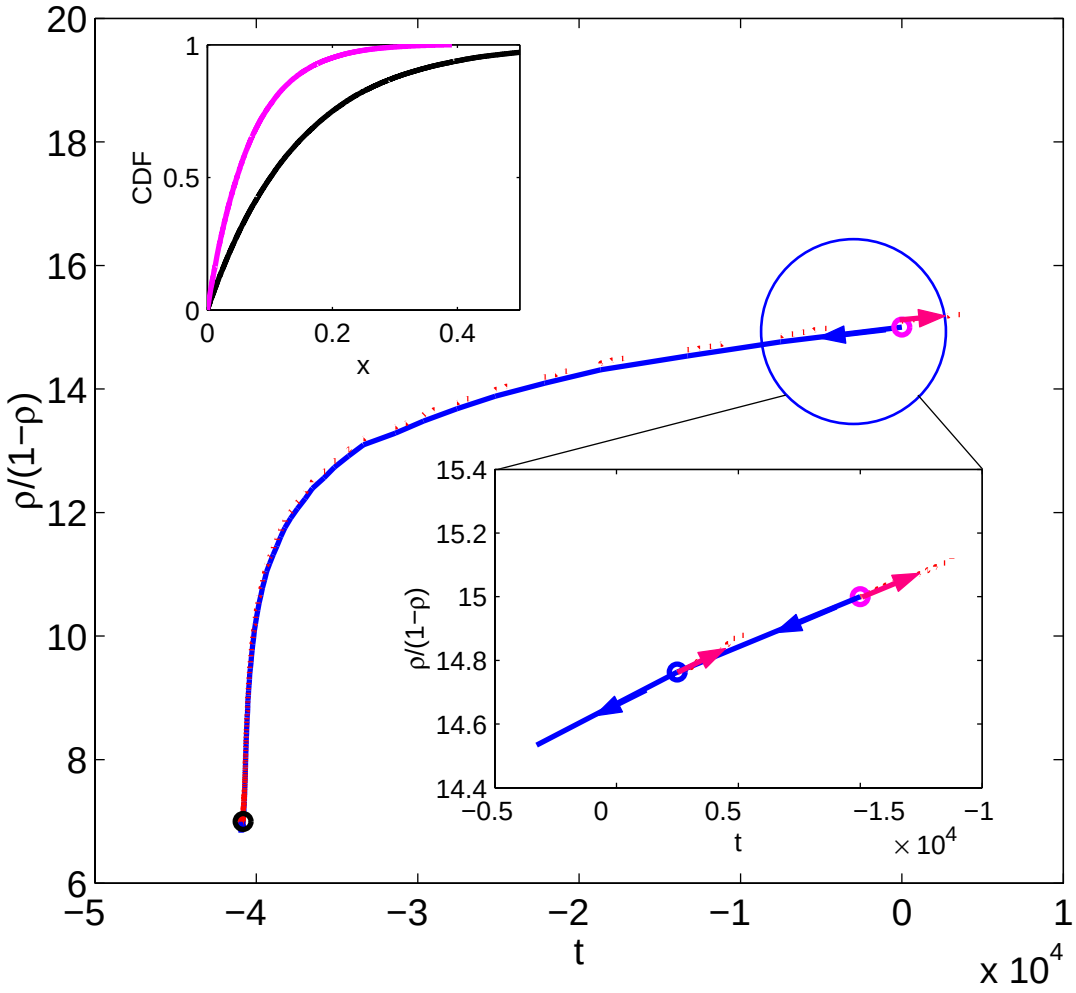
Also



$D\mathbf{\Phi} \cdot \boldsymbol{\varepsilon}$


CDF
x
n
R
n
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
10^{0}
10^{-1}
10^{-2}
10^{-3}
0
2
4
6

Coarse, reverse, projective, in renormalized frame… integration



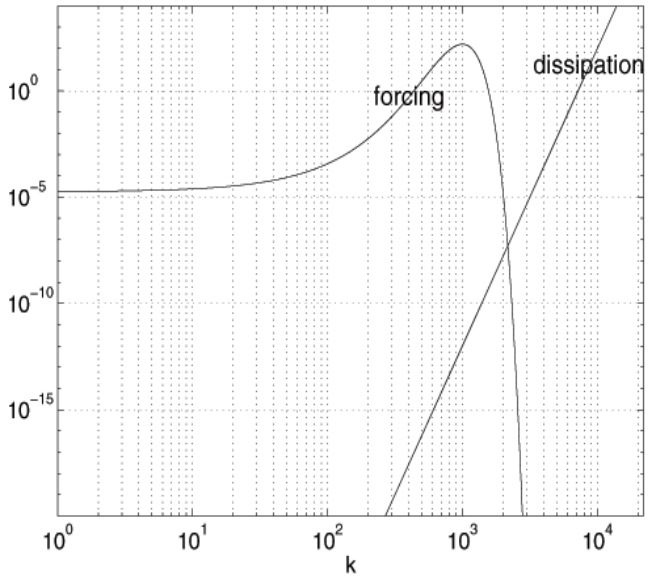
Randomly forced Burgers equation in 1-D

$$u_t + uu_x = (-1)^{n+1} \nu_{hyp} \nabla^{2n} u + f(x,t)$$

$$n = 7, \nu_{hyp} = 10^{-54}$$

- High Reynolds number regime modeled by a hyperviscosity term acting essentially at the smallest scales
- White-in-time forcing acting at scales much smaller than the size of the system

$$f(k,\omega) f(k',\omega') = D_0 e^{-\frac{(k-k_f)^2}{\sigma_f^2}} \delta(k-k')\delta(\omega-\omega')$$



Wavenumber, k

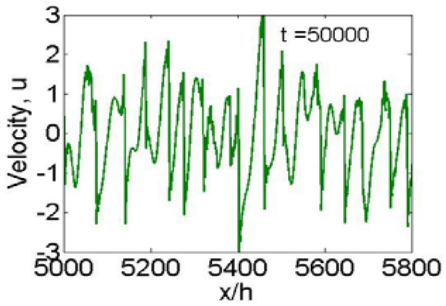
Observables are:

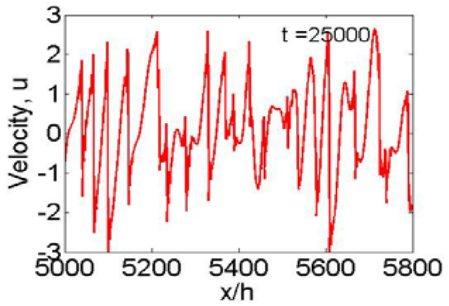
$$E(k,t) = \langle u(k,t) u^*(k,t) \rangle$$

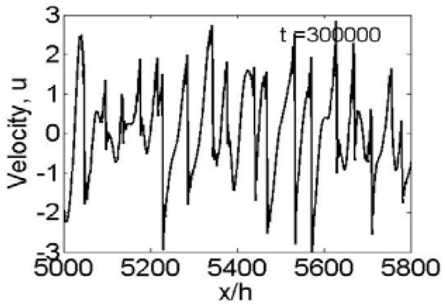
$$S_3(r,t) = \langle (u(x,t) - u(x+r,t))^3 \rangle$$

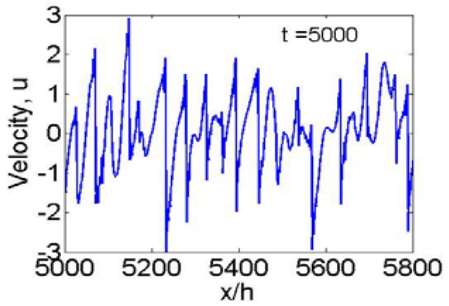
Why *E(k)*?

- Velocity field *u(x)* is stochastic and consists of tiny shocks
- The fields at different times "look" the same



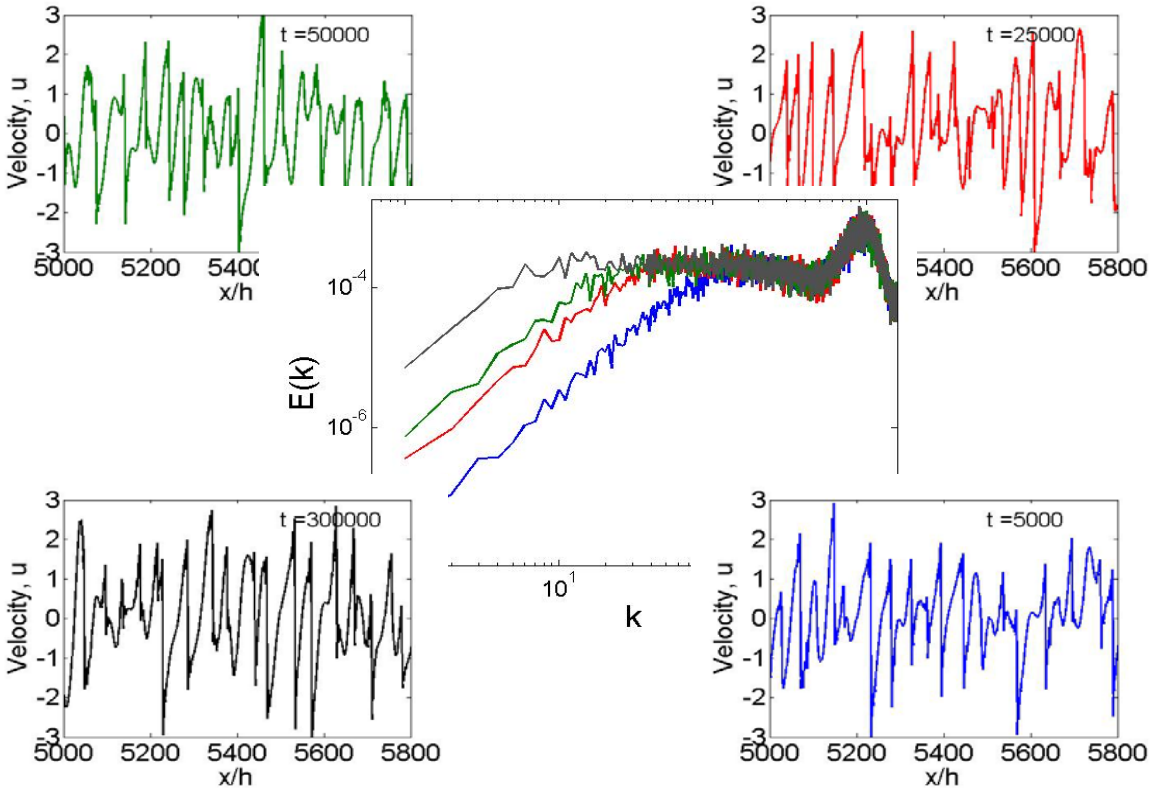






Why *E(k)*?

- Velocity field *u(x)* is stochastic and consists of tiny shocks
- The fields at different times "look" the same
- Energy spectrum enables us to distinguish between the fields
- "Coarse Evolution" of *E(k)* appears deterministic



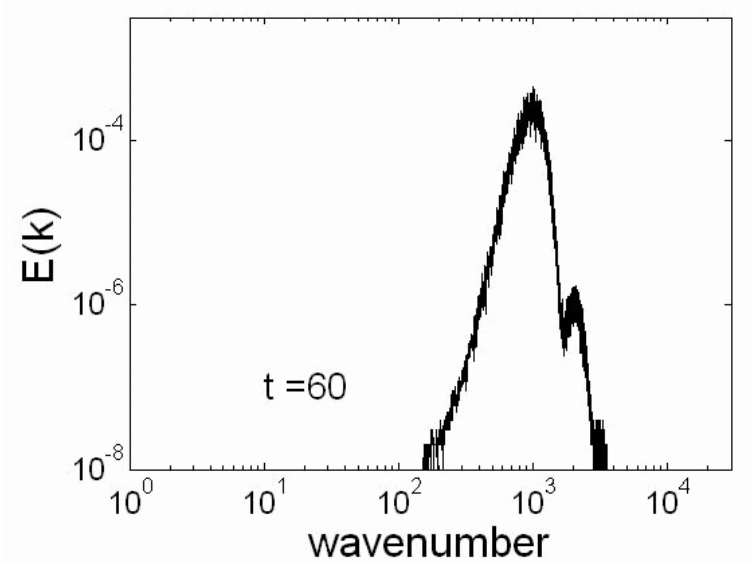
Behavior of *E(k)*

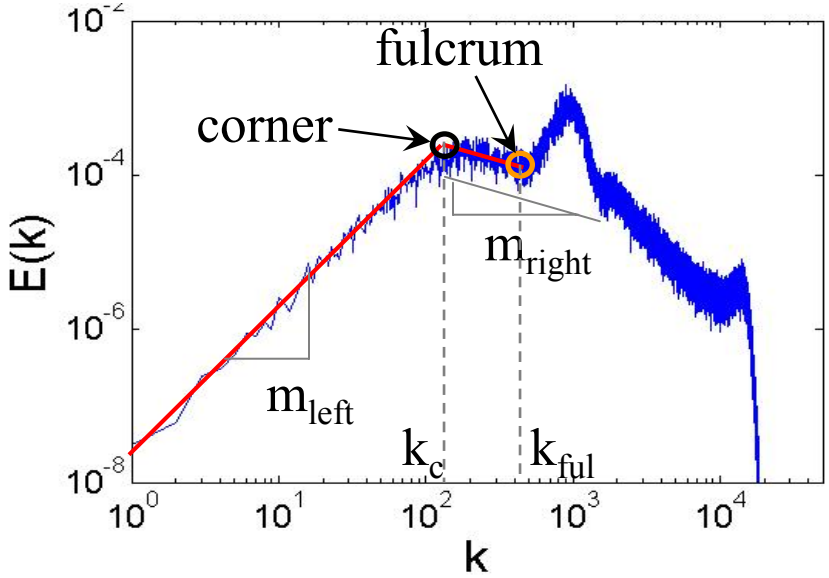
• Fast evolution for the large wavenumbers to stationarity - THEN slow –and slower- evolution for small wavenumbers

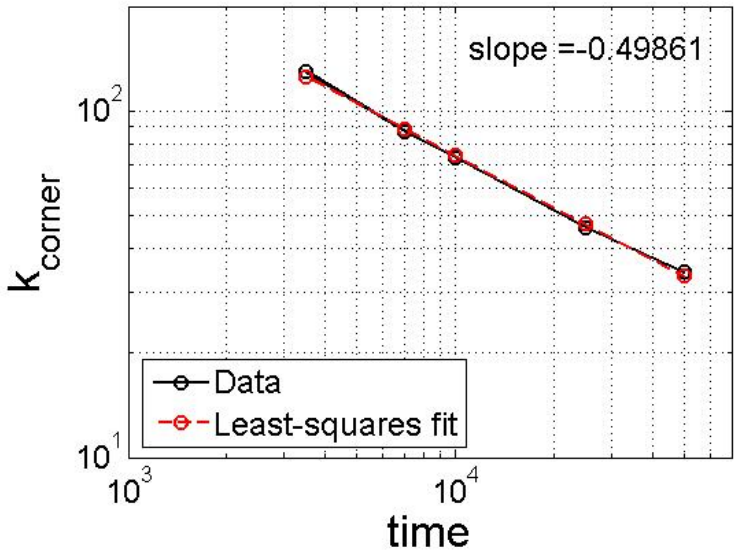
• Steady state for small wavenumbers:

E(k)=const

• Transient evolution in the small wavenumber region can be described by two straight lines

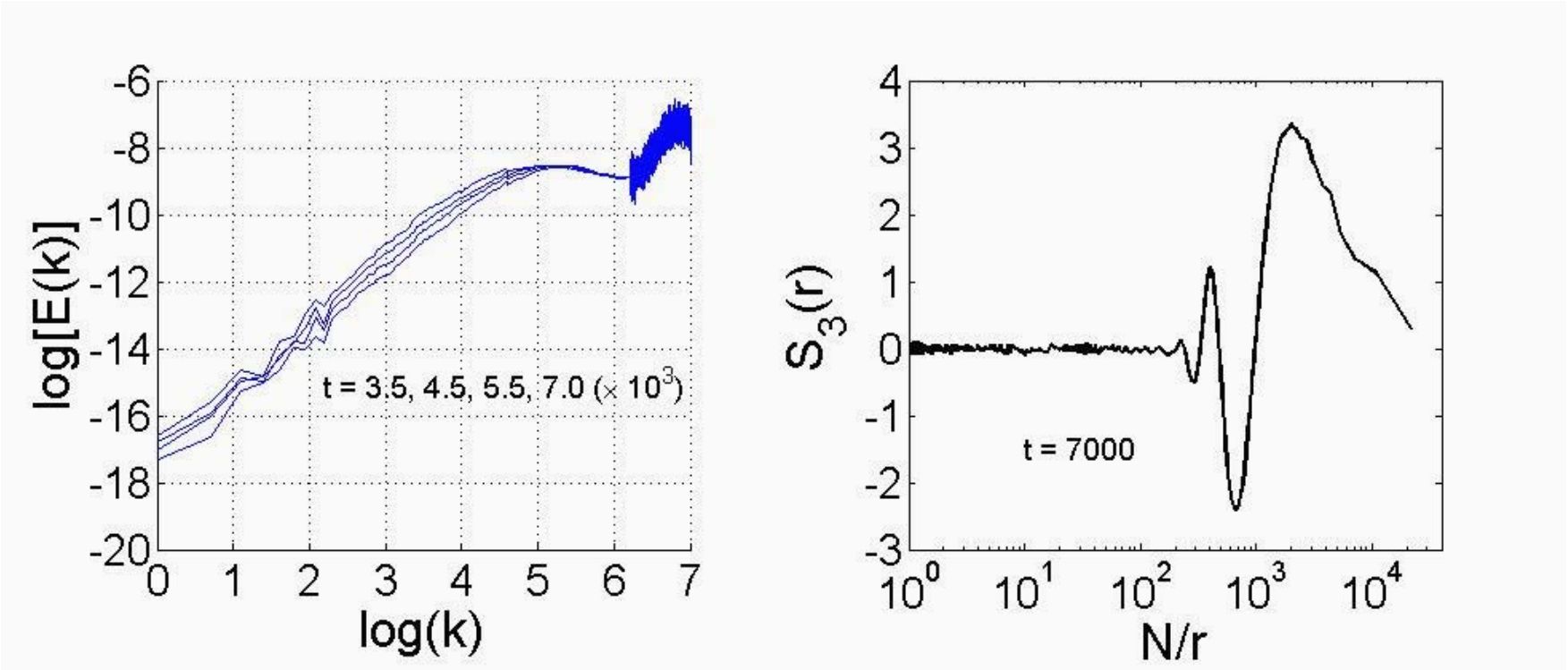






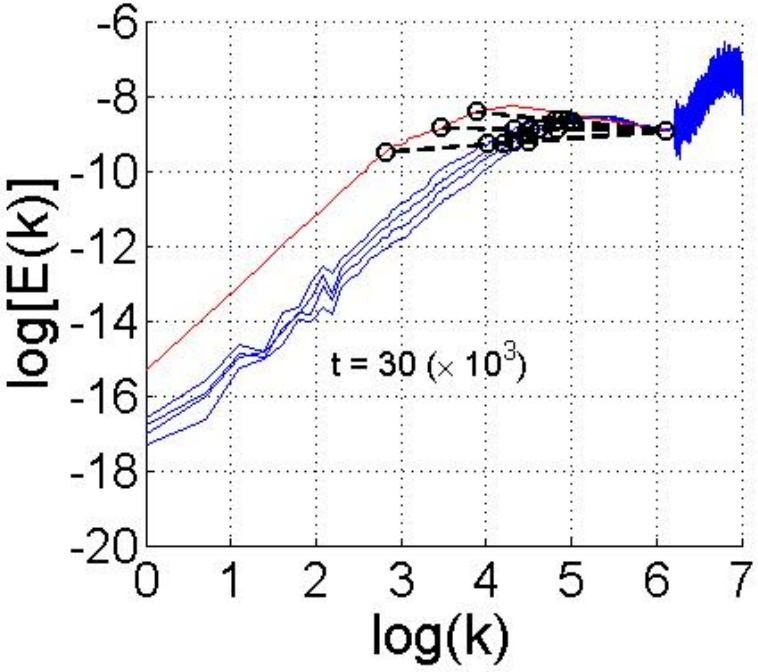
Projective integration

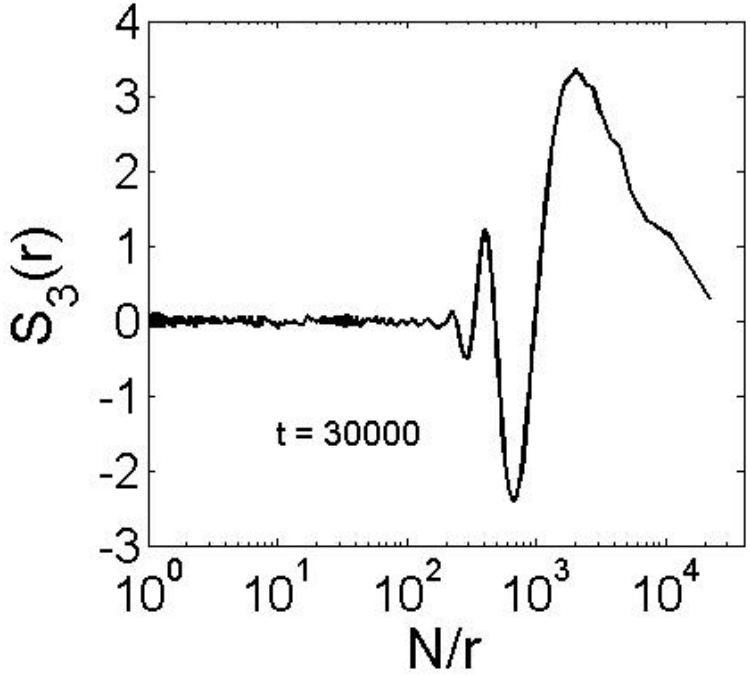
1. Run the simulation for short time and get the averaged stretching factor



Projective integration

1. Run the simulation for short time and get the averaged stretching factor
2. Project *E(k)* using the predicted value of the stretching factor (LIFT)

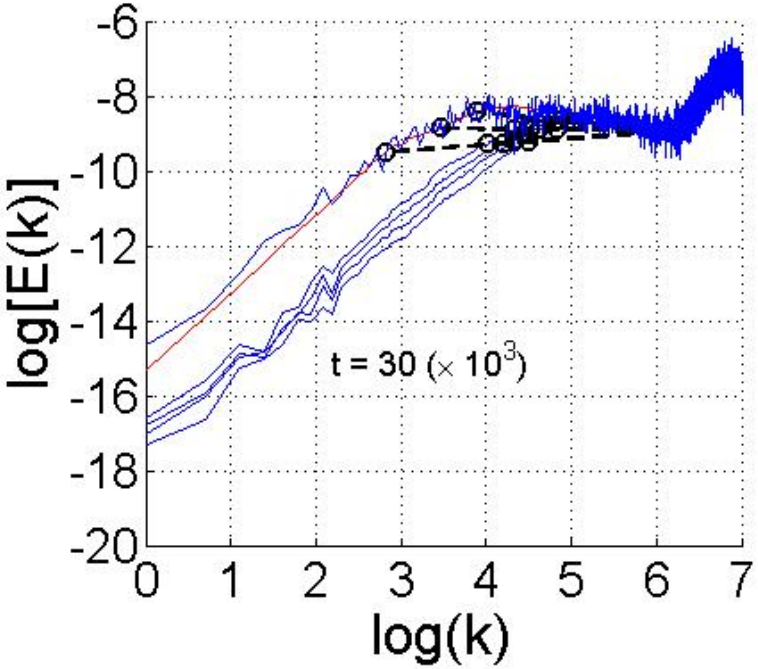


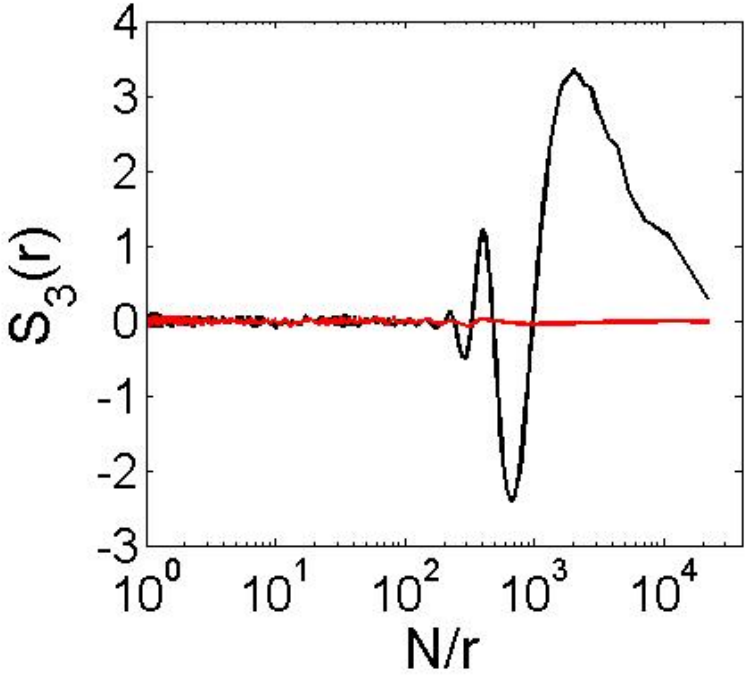


Projective integration

1. Run the simulation for short time and get the averaged stretching factor
2. Project *E(k)* using the predicted value of the stretching factor (LIFT)
3. Randomize the phases to generate new initial conditions

   $S_3(r)$ goes to zero

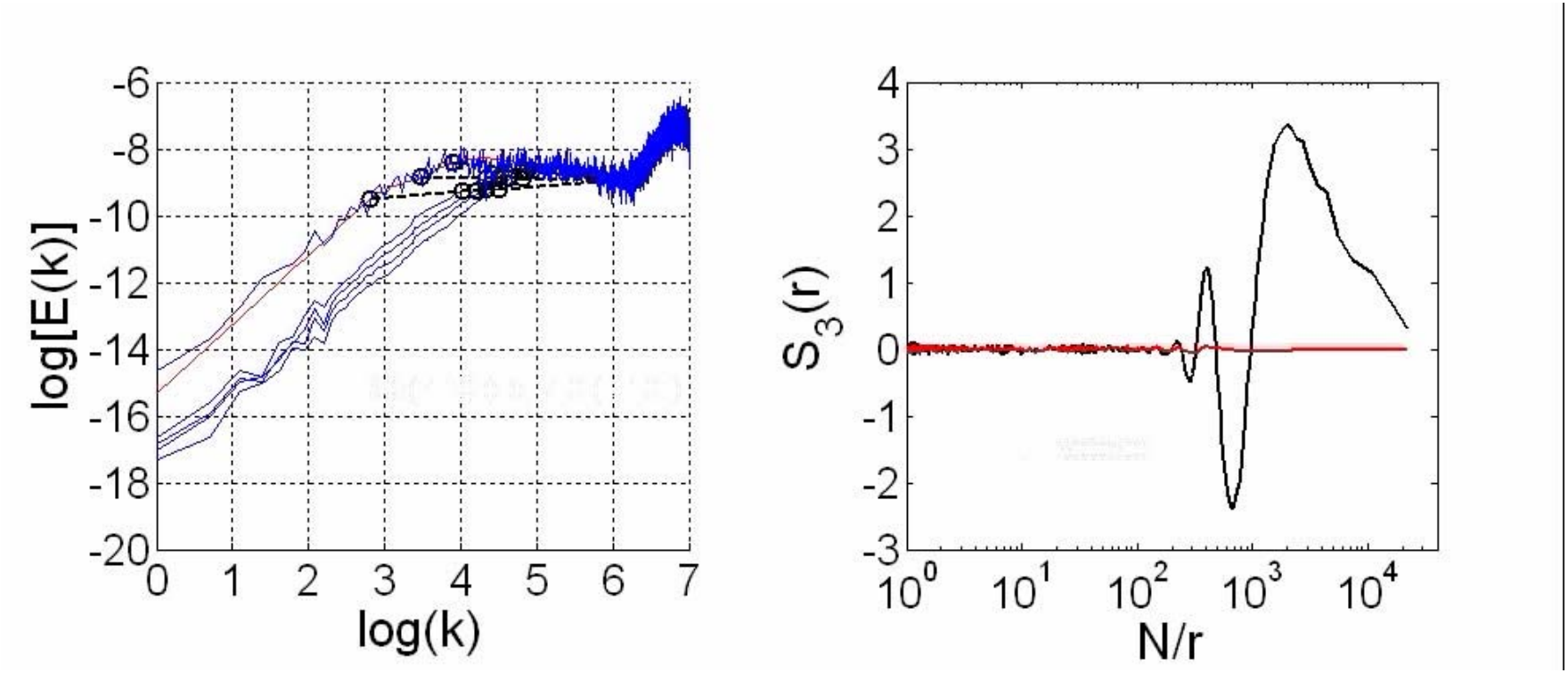




Coarse Projective integration

1. Run the simulation for short time and get the averaged stretching factor
2. Project *E(k)* using the predicted value of the stretching factor (LIFT)
3. Randomize the phases to generate new initial conditions

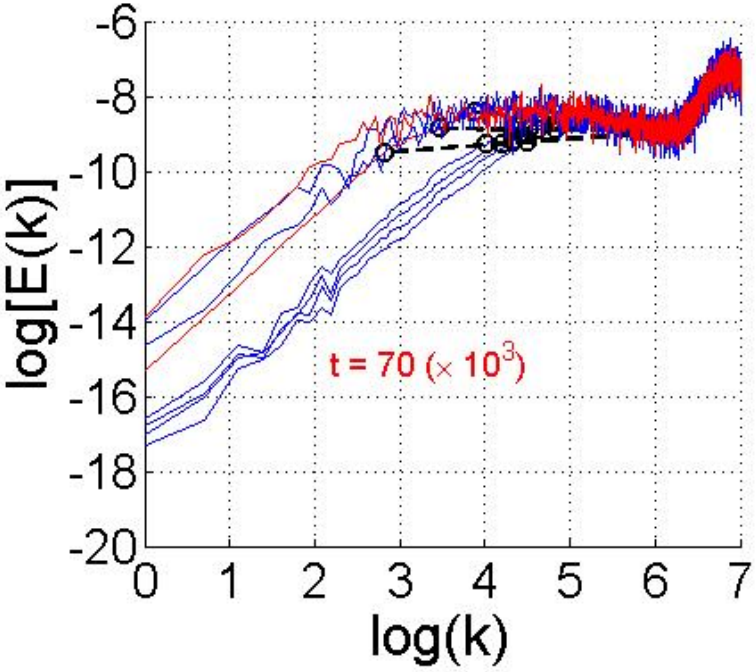
   $S_3(r)$ goes to zero
4. Continue the simulation with new ic's
5. $S_3(r)$ gets enslaved to the steady state value in less than 100 time steps

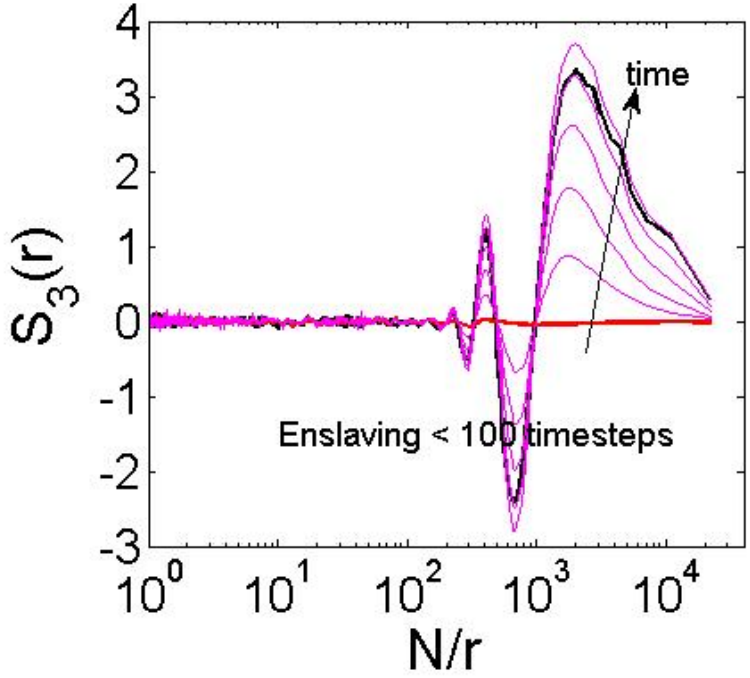


Projective integration

1. Run the simulation for short time and get the averaged stretching factor
2. Project *E(k)* using the predicted value of the stretching factor (LIFT)
3. Randomize the phases to generate new initial conditions

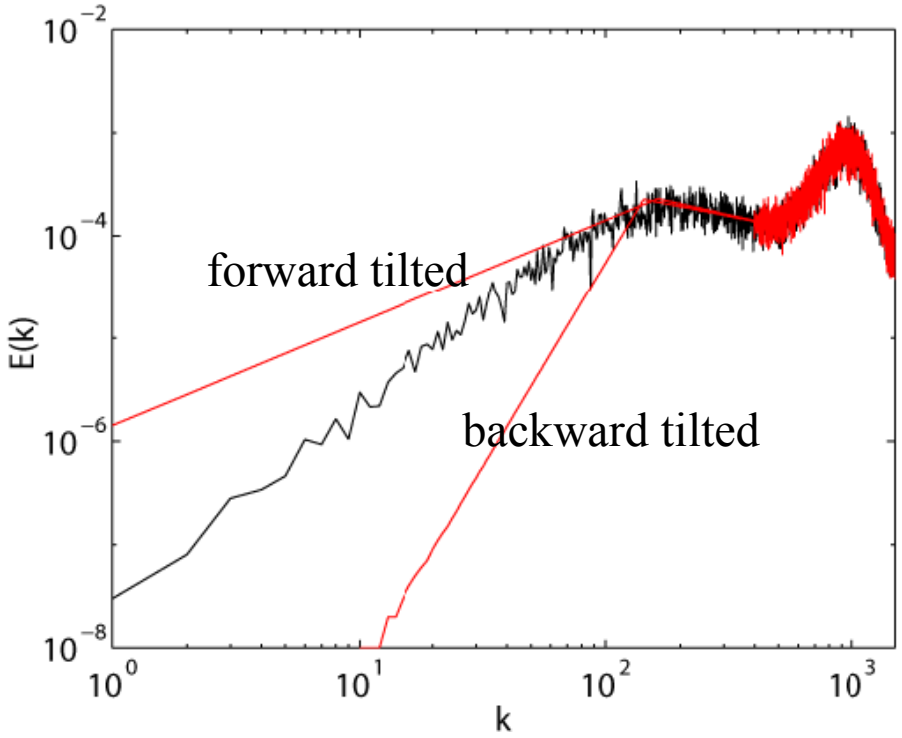
   $S_3(r)$ goes to zero
4. Continue the simulation with new ic's
5. $S_3(r)$ gets enslaved to the steady state value in less than 100 time steps
6. *E(k)* thus obtained evolves with the original simulation





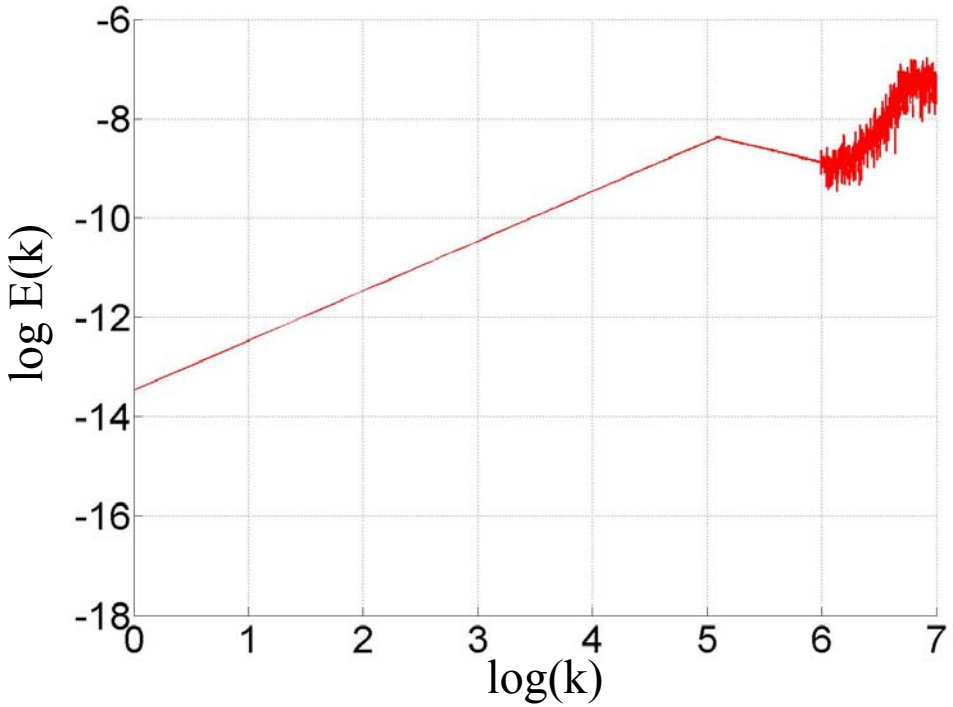
"Non self-similar" initial condition

- Projective integration designed for initial conditions in the "self-similar" regime
- Other initial conditions *renormalized* using *run and restrict* algorithm
- *Forward-tilted* and *backward-tilted* initial conditions considered



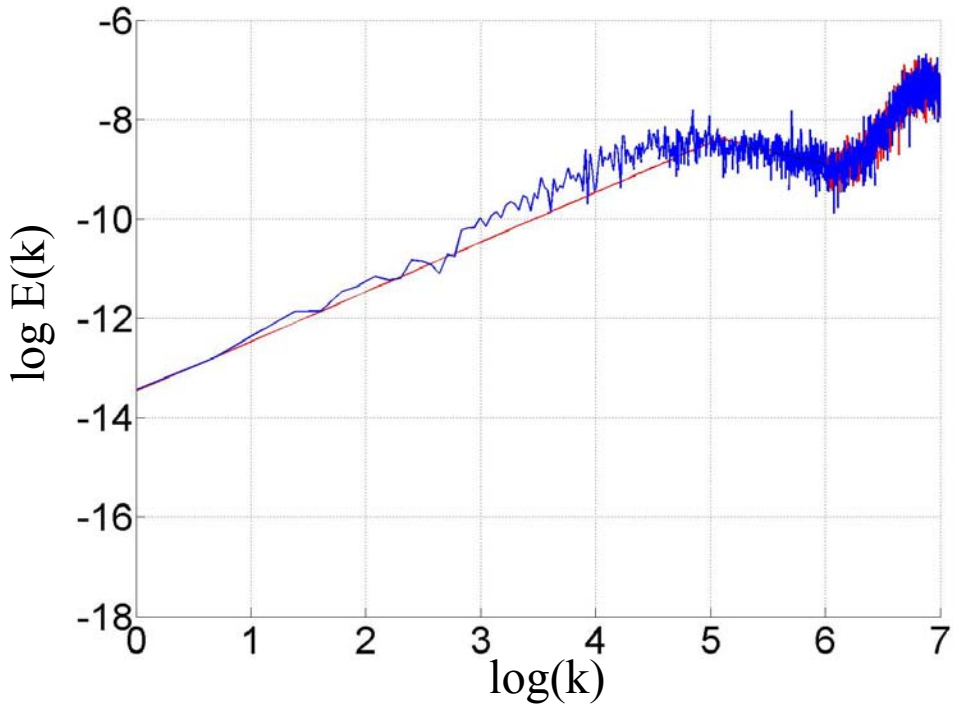
Renormalization algorithm

- A simplified initial condition



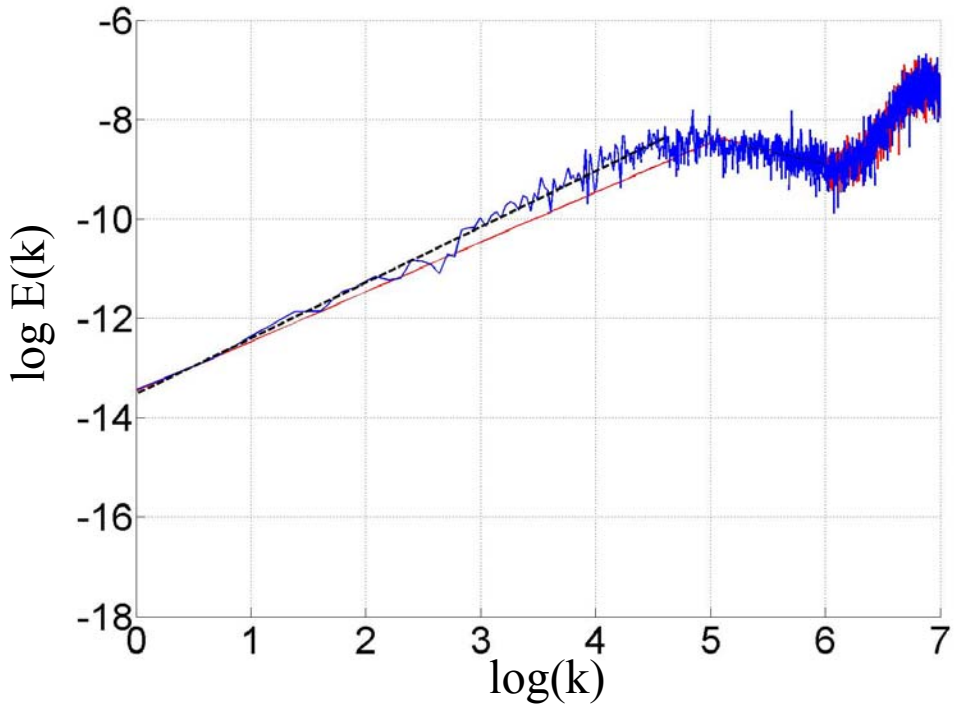
Renormalization algorithm

- A simplified initial condition
- Run for short time (5000 time steps)



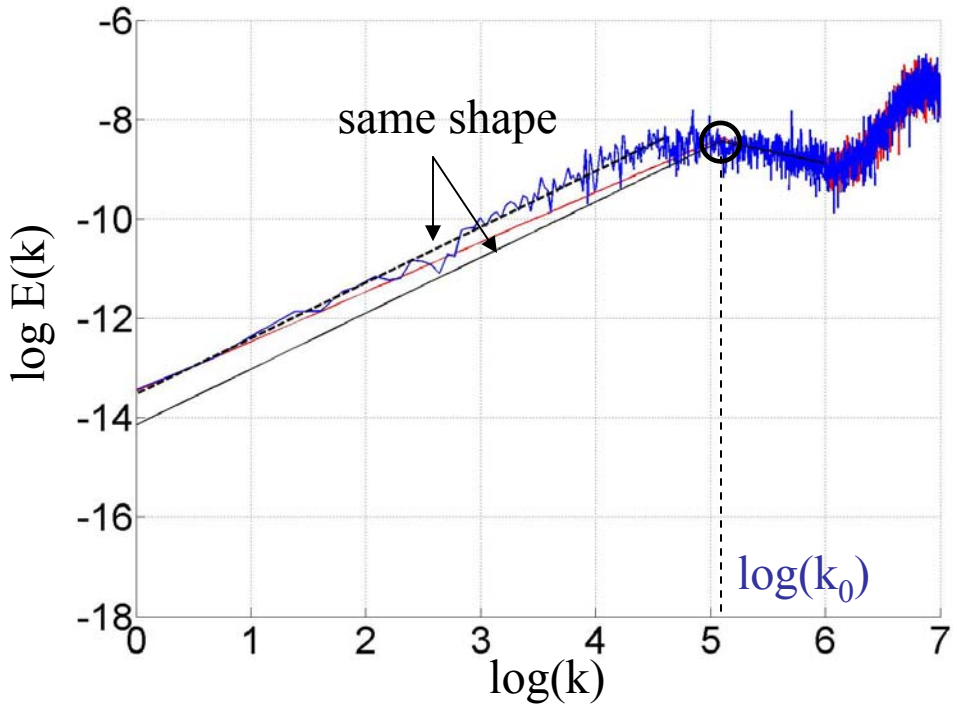
Renormalization algorithm

- A simplified initial condition
- Run for short time (5000 time steps)
- Approximate the spectrum by straight lines



Renormalization algorithm

- A simplified initial condition
- Run for short time (5000 time steps)
- Approximate the spectrum by straight lines
- Pull back the shape keeping the initial ordinate fixed

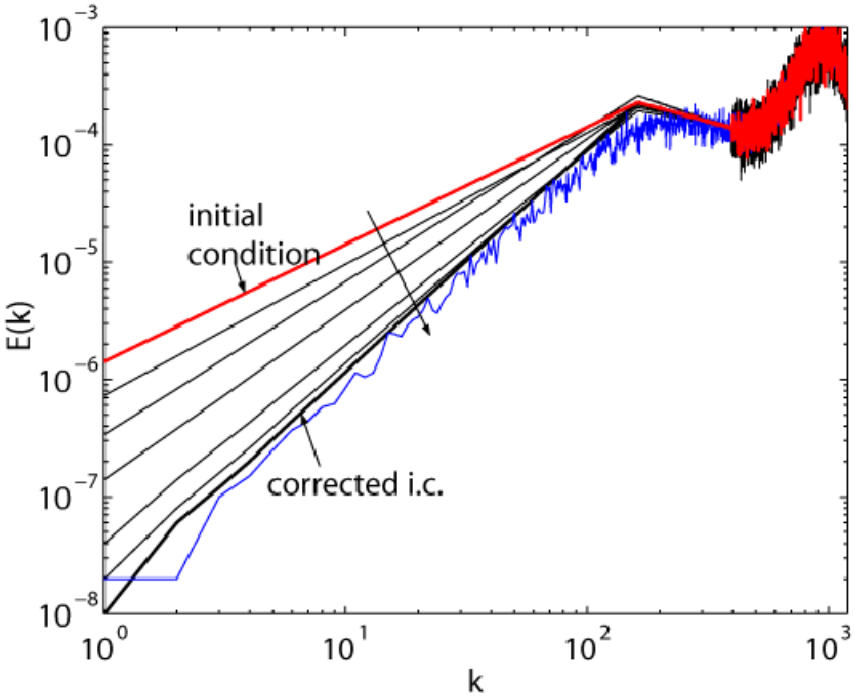


Renormalization algorithm

- Sequence of iterations leading to the right shape
- Comparison with a representative spectrum

Forward tilted i.c.

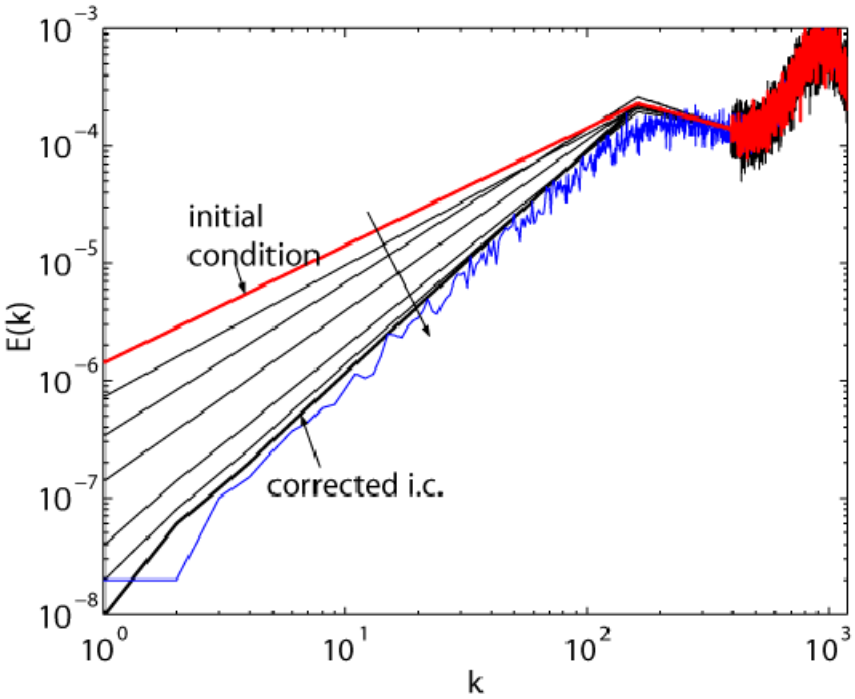
- convergence in 6-8 iterations



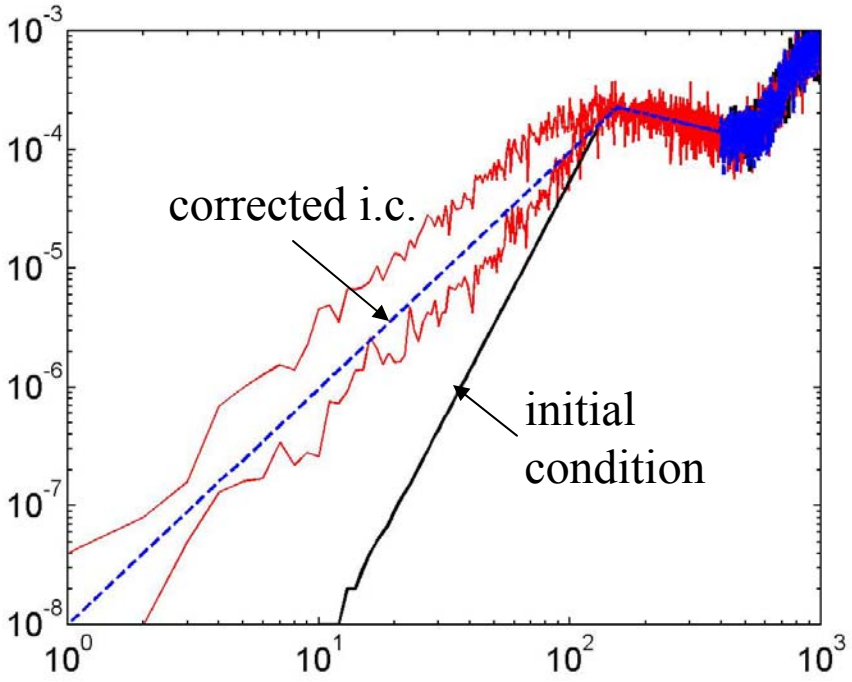
Renormalization algorithm

- Sequence of iterations leading to the right shape
- Comparison with a representative spectrum

Forward tilted i.c.
- convergence in 6-8 iterations



Backward tilted i.c.
- convergence in 1 iteration



A r-fixed point computation

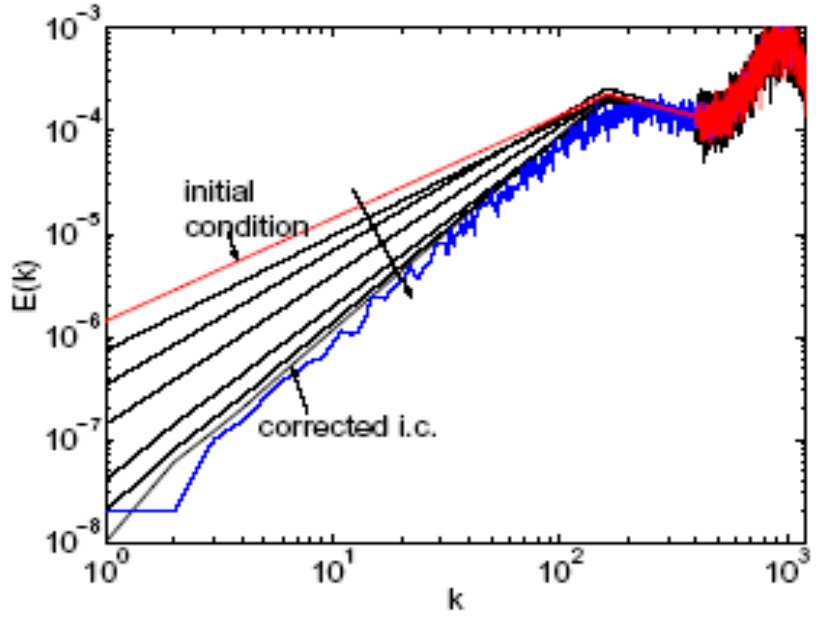
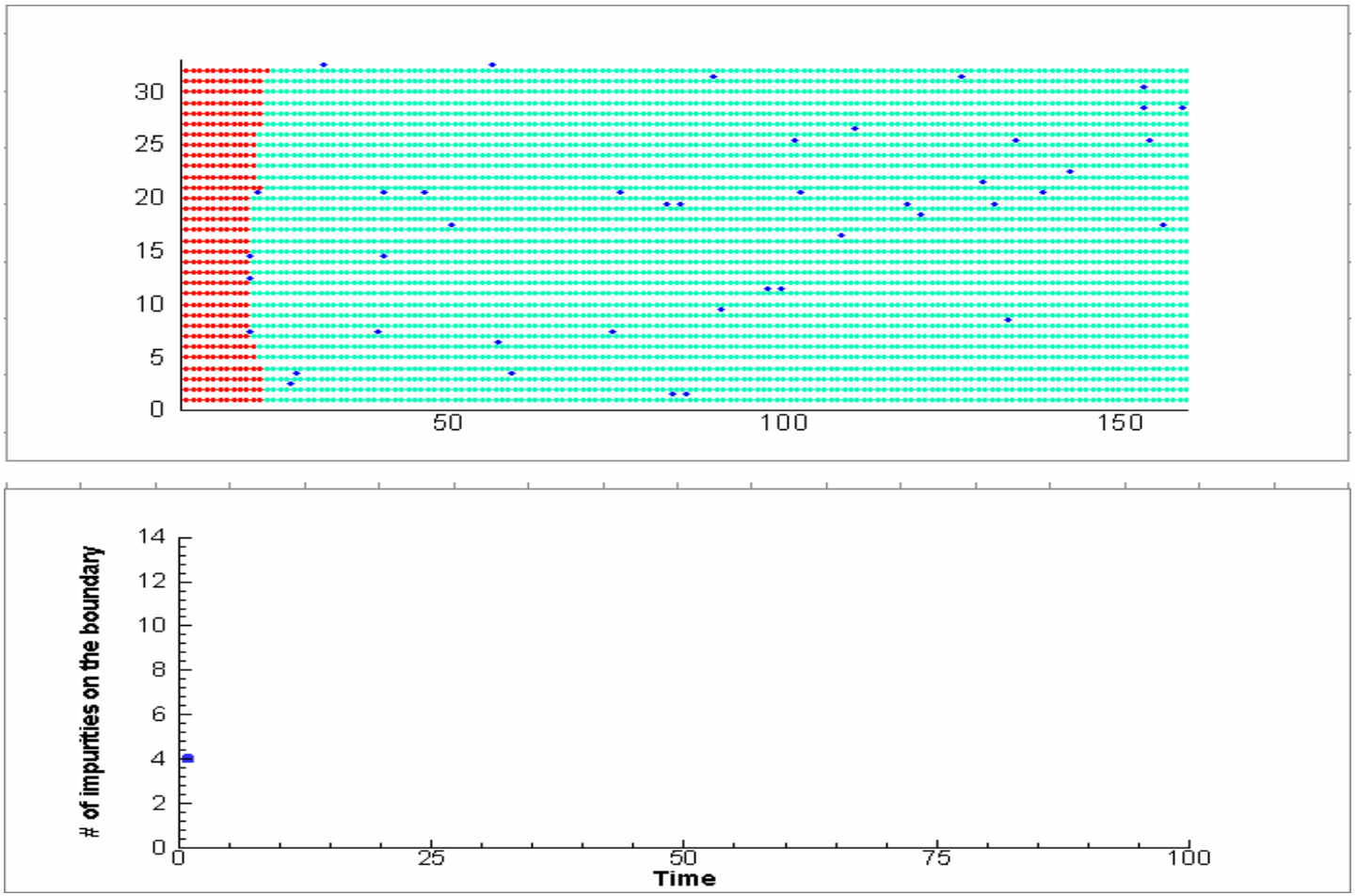


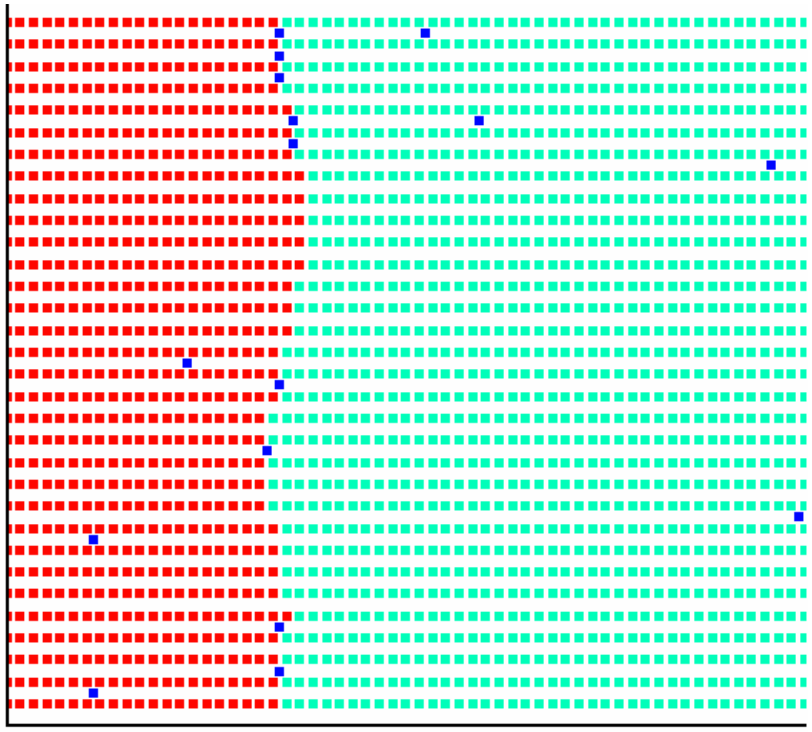
Figure 19: Sequence of iterations or corrections for the "tilted-forward" initial condition. Final comparison with a representative of the self-similar transients (shown in blue).

Rare Events at the Onset of Motion:

a magnetization front in a field of mobile impurities



Model for domain walls and impurities



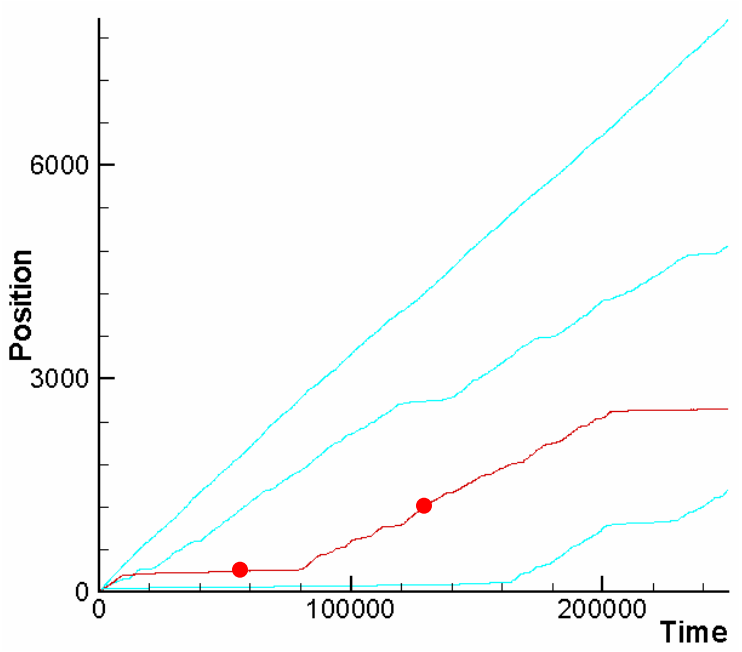
- Simple lattice model – tractable numerically but physically motivated (Mendelev and Srolovitz, 2001)
- Write the model Hamiltonian as

$$H = -J\sum_{<ij>} s_i s_j - h\sum_i s_i + E_0 \sum_i c_i \left| \sum_j s_j \right|$$

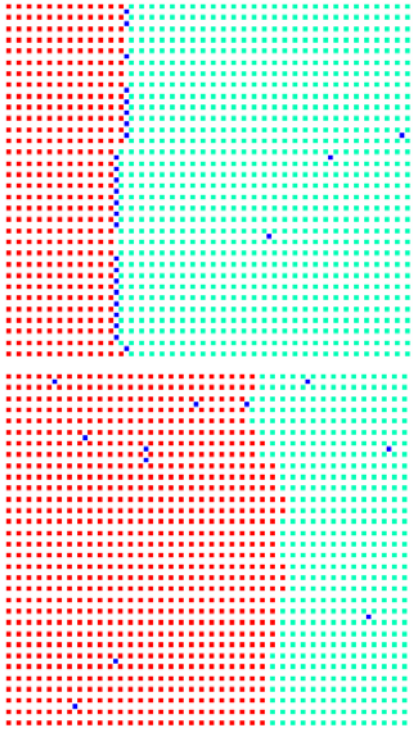
- No double occupancy of the interstitial impurity sites

- Important parameters: domain wall length W, external drive h, impurity-domain wall interaction energy E_0, impurity density c_{imp}, and impurity diffusivity D
- Simulation method: kinetic Monte Carlo

Domain wall dynamics

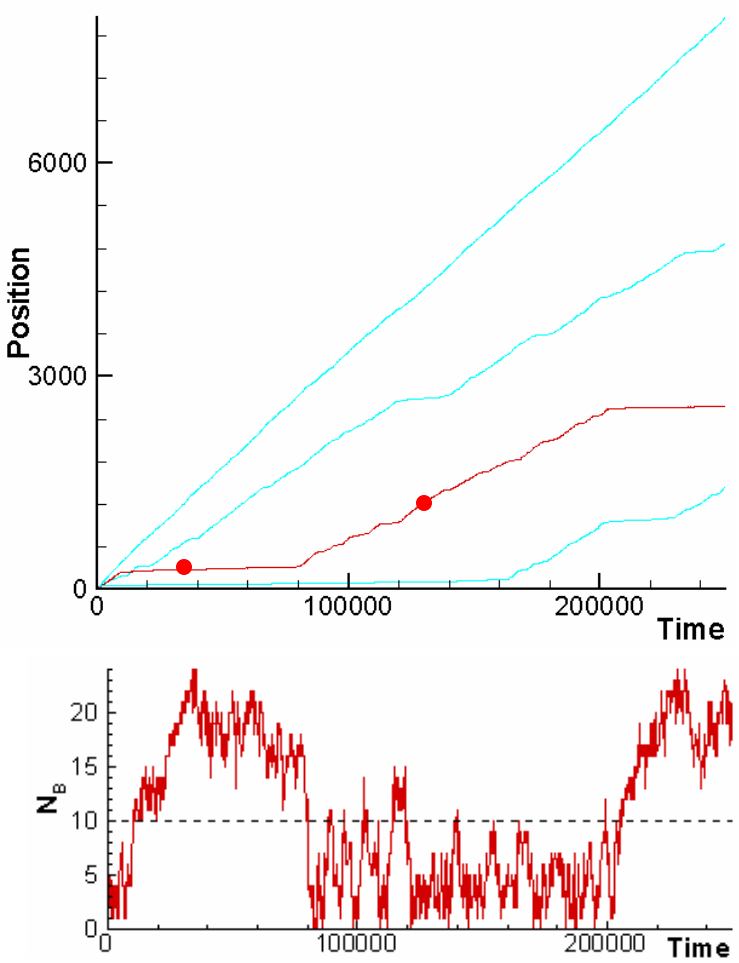


Increasing $|E_0|$



- For fixed drive and diffusivity, small interaction energy $|E_0|$ leads to **smooth** propagation
- Increasing $|E_0|$ leads to **"jerky"** motion with long periods of "**pinning**" and "**running**"

“Jerky” motion of the domain wall



- In the **jerky** regime, **pinned state** is characterized by a large number of impurities along the domain wall N(t)
- **Running state** is characterized by a small number of impurities along the domain wall
- More quantitatively, the two states correspond to two different average impurity numbers along the domain wall
- This suggests a two-state picture of the domain wall/impurity dynamics!

Coarse Fokker-Planck analysis

• **Quantitative picture**: assume that N(t) is a Markovian (i.e., no memory effects) and Gaussian process ↔ can derive an equation for the time evolution of the probability distribution P(N,t) (also known as the Fokker-Planck equation!):

$$\frac{\partial P(N,t)}{\partial t} = \left[-\frac{\partial V(N)}{\partial N} + \frac{\partial^2 D(N)}{\partial N^2} \right] P(N,t)$$

• The so-called drift (V) and diffusion (D) terms can be extracted from numerical data by employing the following (formal) definitions:

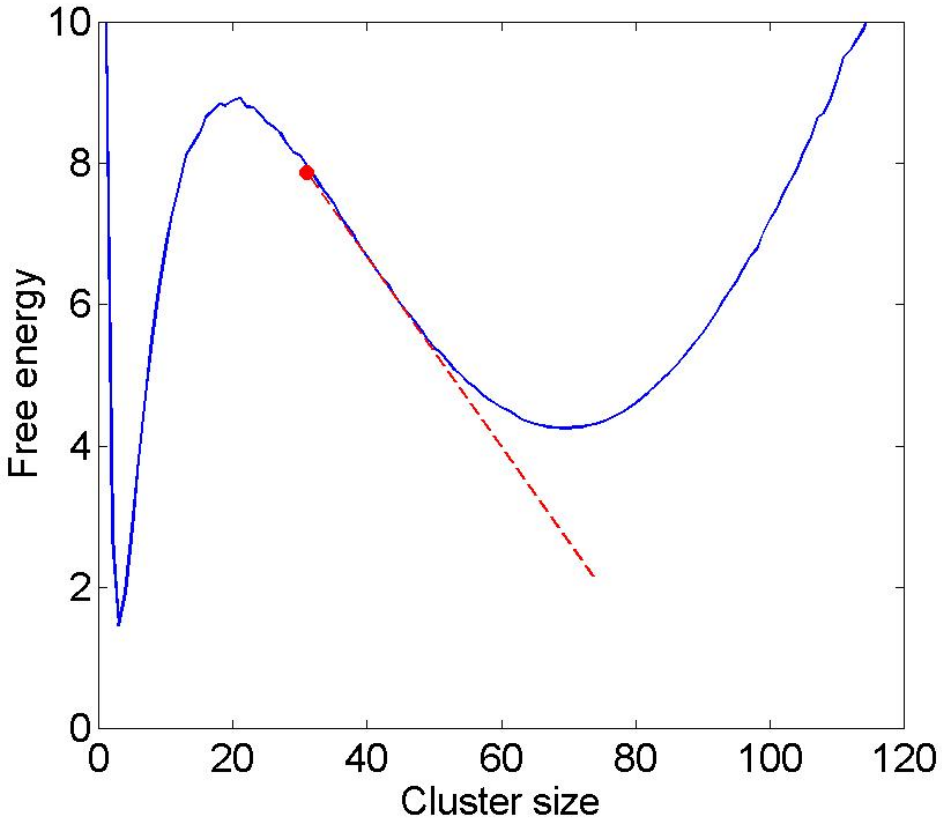
$$V(N) \equiv \lim_{\Delta t \to 0} \left\langle \Delta N \right\rangle / \Delta t \qquad 2D(N) \equiv \lim_{\Delta t \to 0} \left\langle \left[\Delta N \right]^2 \right\rangle / \Delta t$$

• Important: V(N) and D(N) are required for <u>all possible</u> values of N!

Reconstruction of Free Energy Surface

$$\gamma \dot{N} + G'(N) = F(t)$$

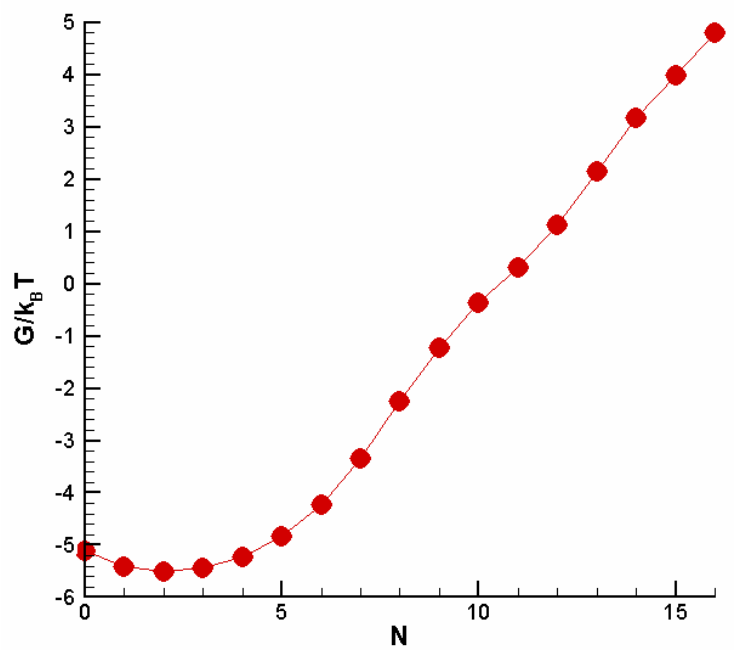
Obtain G(N) and γ(N) from short-scale nonequilibrium simulations

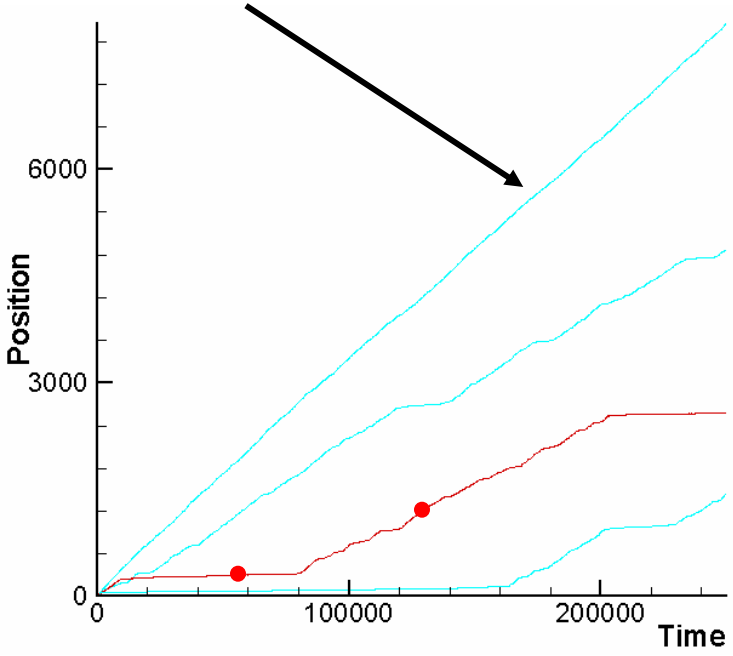


$$\frac{d\langle N \rangle}{dt} = -\frac{1}{\gamma} G'(N)$$

$$\frac{d\,\mathrm{var}(N)}{dt} = \frac{2k_B T}{\gamma}$$

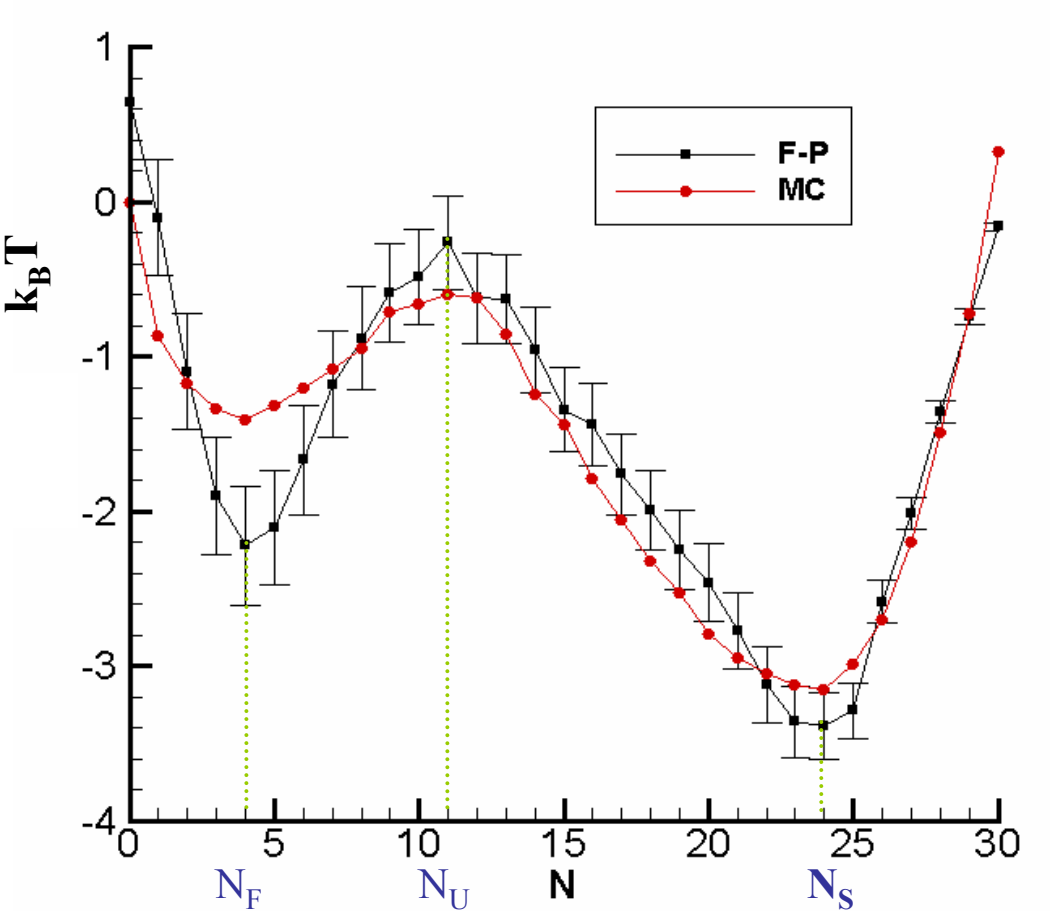
Effective free-energy – weak interaction





- **G has only one minimum** – no matter where N starts at, it very quickly relaxes towards this **global minimum**
- This implies that there is only **one "coarse steady-state"**, and thus a **unique speed (i.e., no hysteresis)**!

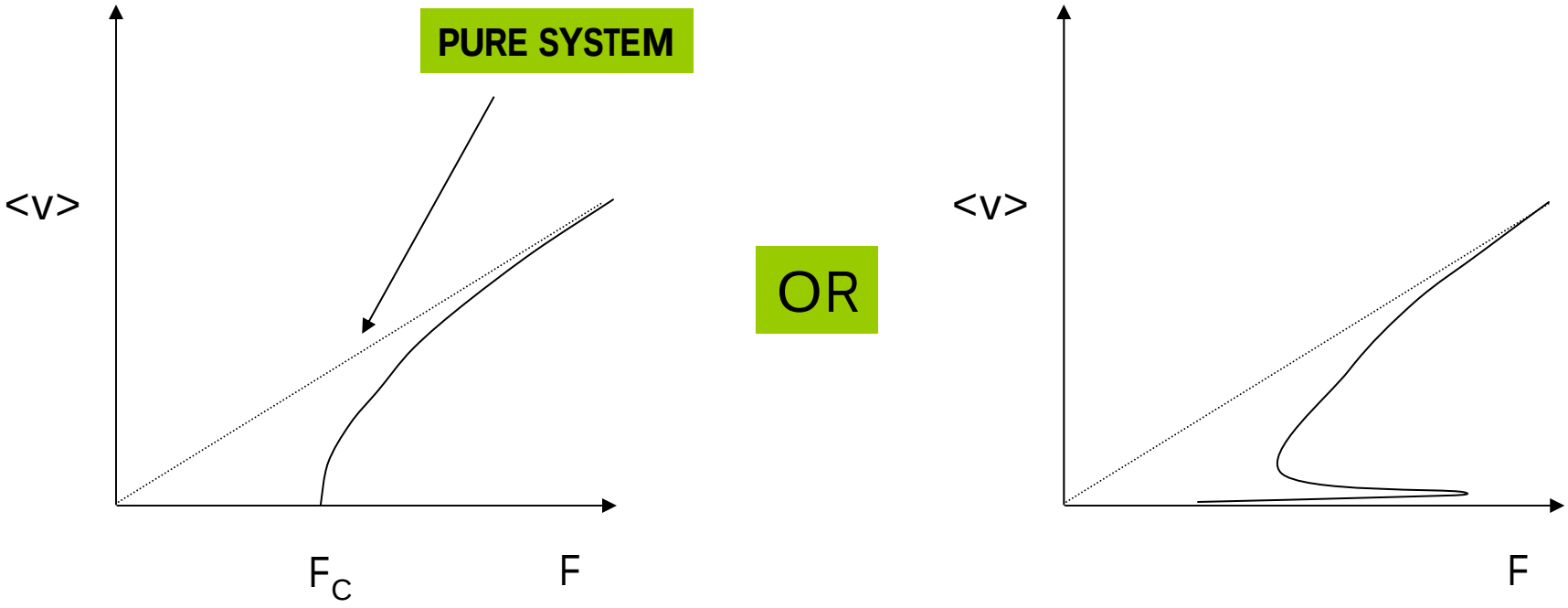
Effective potential – strong interaction



- Familiar double well structure with **local minima** at N=4 and N=24, and **(unstable) saddle point** around N=11
- System initially around one of the local minima remains there untill a sufficiently large fluctuation comes along!!

Motivation

In essence, we are asking whether the response of the system is one of the following:

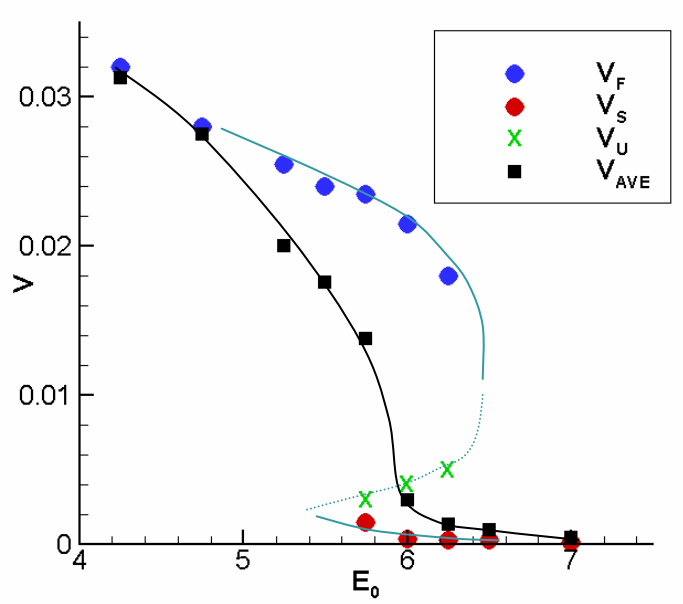


We will argue that the answer is, in some sense, “both”!!

Speed vs. heat of segregation

- Computed speed V vs. E_0 graph contains **multiple solutions** over a wide range of interaction energy – **observation of hysteresis possible**!!!

- The analysis & data are consistent with the following "common sense" thoughts:

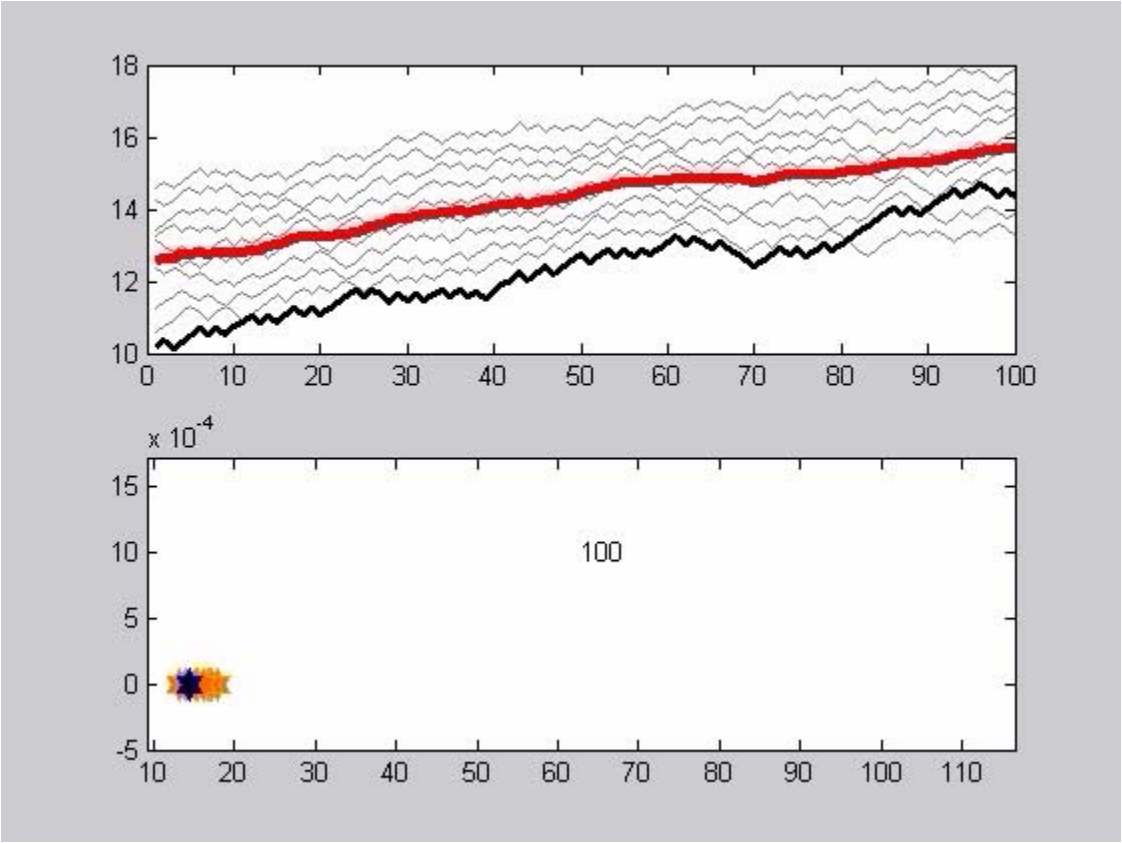


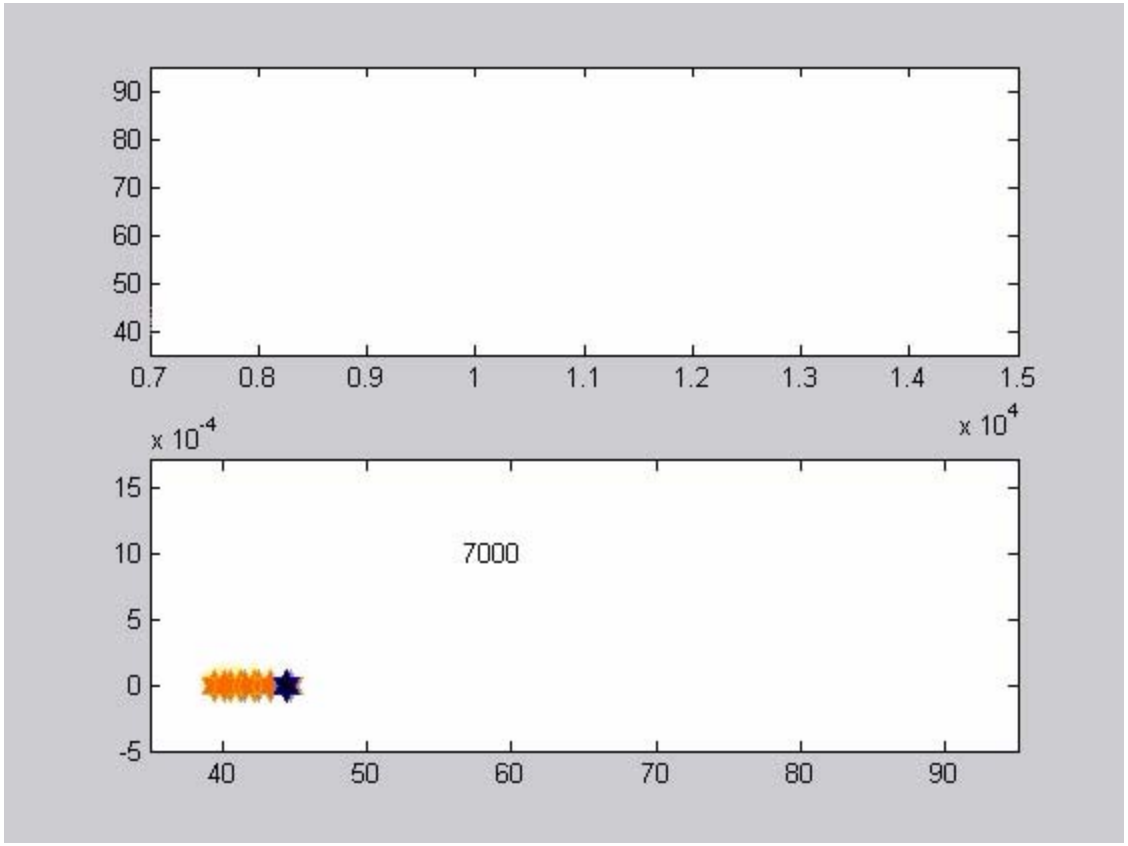
(1) A domain wall with just a **few impurities** is (more or less) **free to propagate**; need a **critical number of impurities** to slow down propagation sufficiently and collect even more impurities!

(2) If the domain wall is **saturated with impurities**, it will **remain saturated** for some time until it loses the impurities.

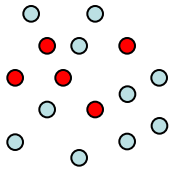
and now for something completely different:

Little stars ! (well…. think fishes)




90
80
70
60
50
40
0.7
0.8
0.9
1
1.1
1.2
1.3
1.4
1.5
x 10^4
x 10^-4
15
10
5
0
-5
7000
40
50
60
70
80
90

Fish Schooling Models



Initial State

Position, Direction, Speed

$$c_i(t), v_i(t), s_i(t)$$

- INFORMED
- UNINFORMED

Compute Desired Direction $d_i(t+\Delta t)$

Zone of Deflection $R_{ij}<\alpha$

$$d_i(t+\Delta t) = -\sum_{j\neq i} \frac{c_j(t)-c_i(t)}{\left|c_j(t)-c_i(t)\right|}$$

Zone of Attraction $R_{ij}<\rho$

$$d_i(t+\Delta t) = \sum_{j\neq i} \frac{c_j(t)-c_i(t)}{\left|c_j(t)-c_i(t)\right|} + \sum_{j=1} \frac{v_j(t)}{\left|v_j(t)\right|}$$

Normalize $\rightarrow \quad \hat{d}_i(t+\Delta t)$

Update Direction for Informed Individuals ONLY $d_i'(t+\Delta t)$

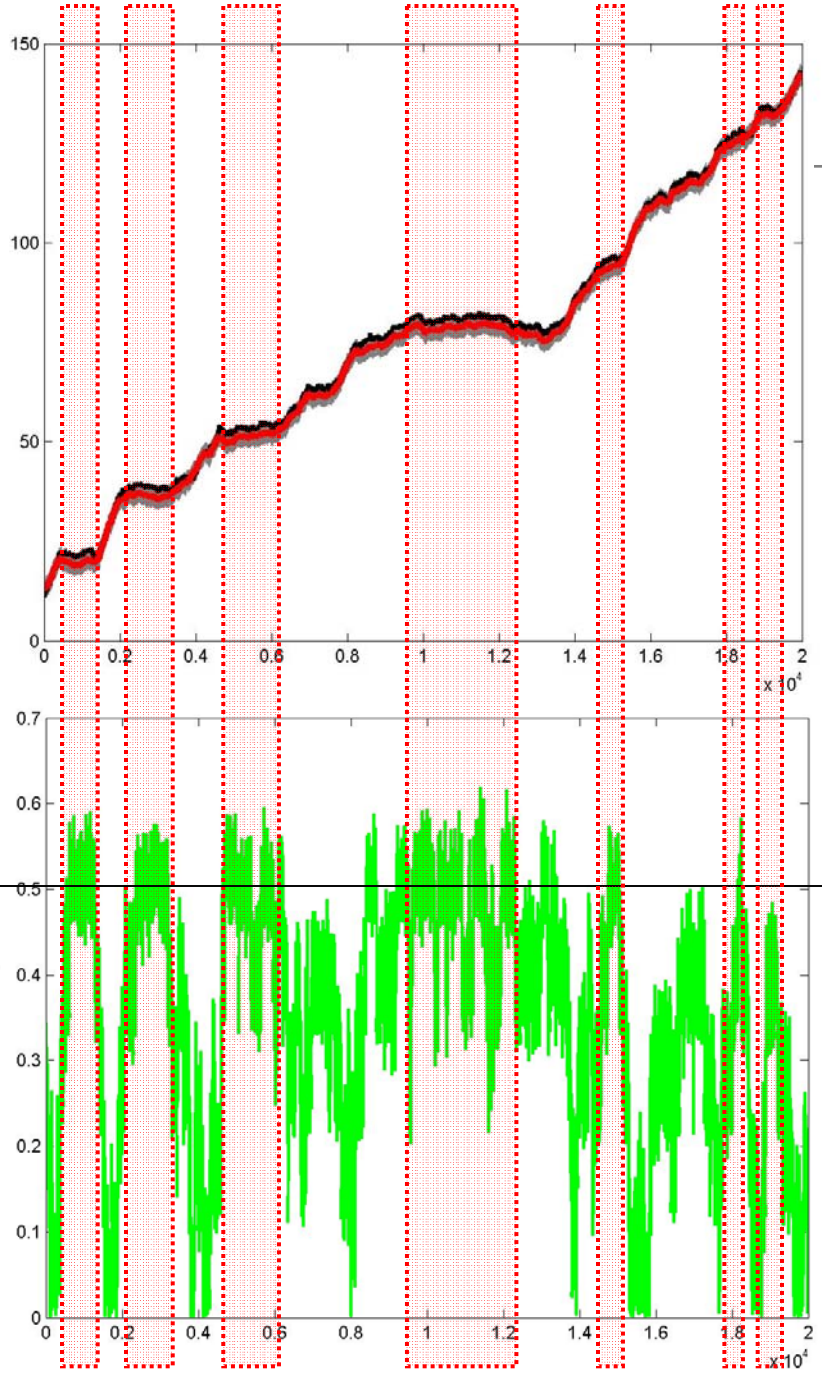
$$d_i'(t+\Delta t) = \frac{\hat{d}_i(t+\Delta t)+\omega g_i}{\left|\hat{d}_i(t+\Delta t)+\omega g_i\right|}$$

Update Positions

$$c_i(t+\Delta t) = c_i(t) + v_i(t+\Delta t) s_i \Delta t$$

Couzin, Krause, Franks & Levin (2005) *Nature* (433) 513

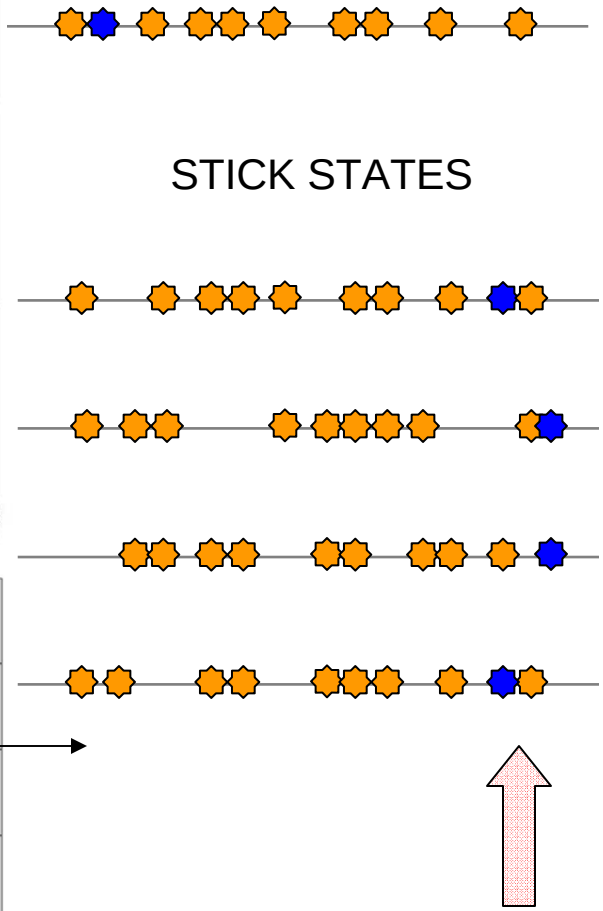
STUCK



~ typically around rxn coordinate value of about 0.5

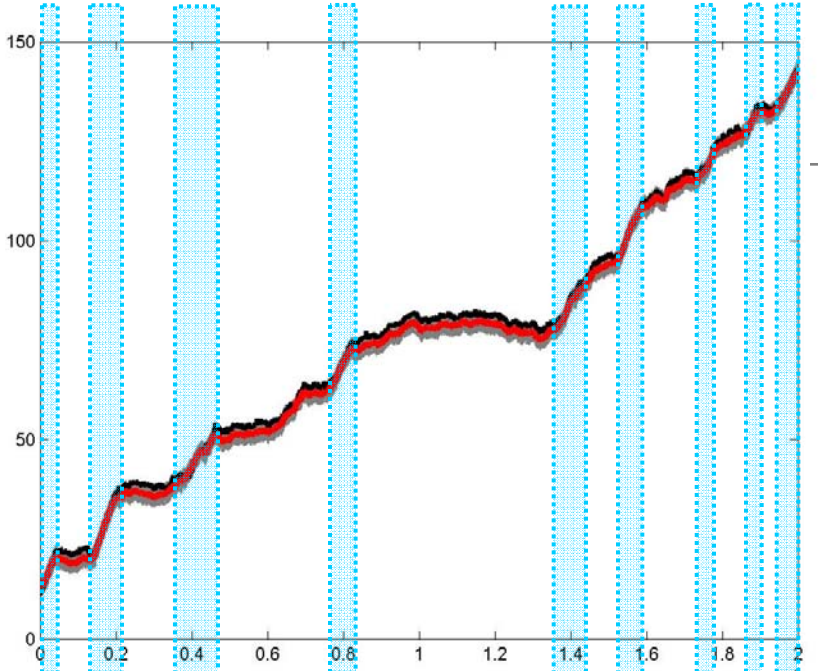
INFORMED DIRN

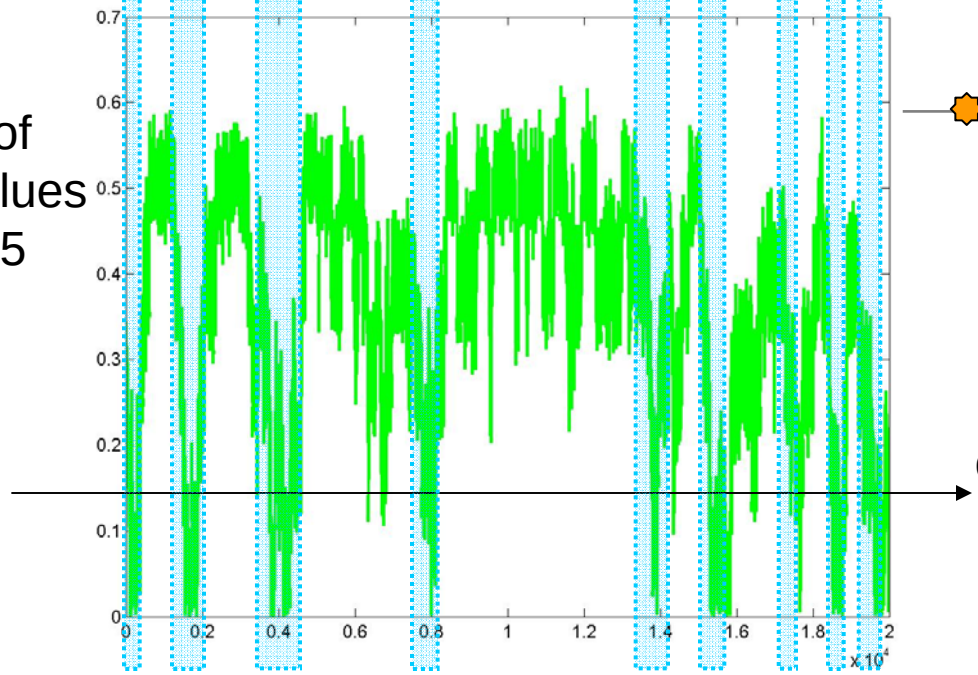
STICK STATES



INFORMED individual close to front of group away from centroid

SLIP

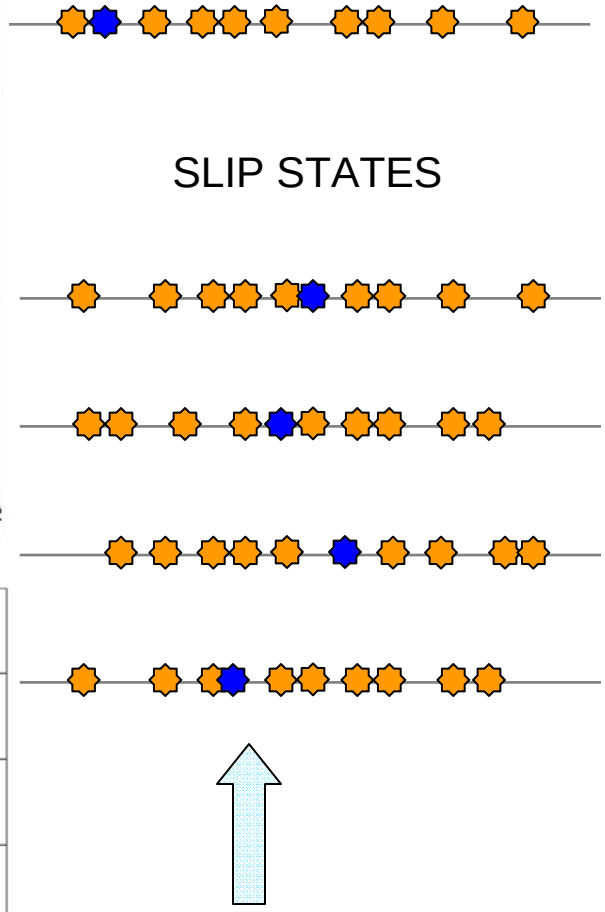




~ wider range of rxn coordinate values for slip 0→0.35

INFORMED DIRN

SLIP STATES



INFORMED individual close to group centroid

Effective Fokker-Planck Equation

FPE: $$\frac{\partial P(r,t)}{\partial t} = \left[-\frac{\partial}{\partial r} v(r) + \frac{\partial^2}{\partial r^2} D(r) \right] P(r,t)$$

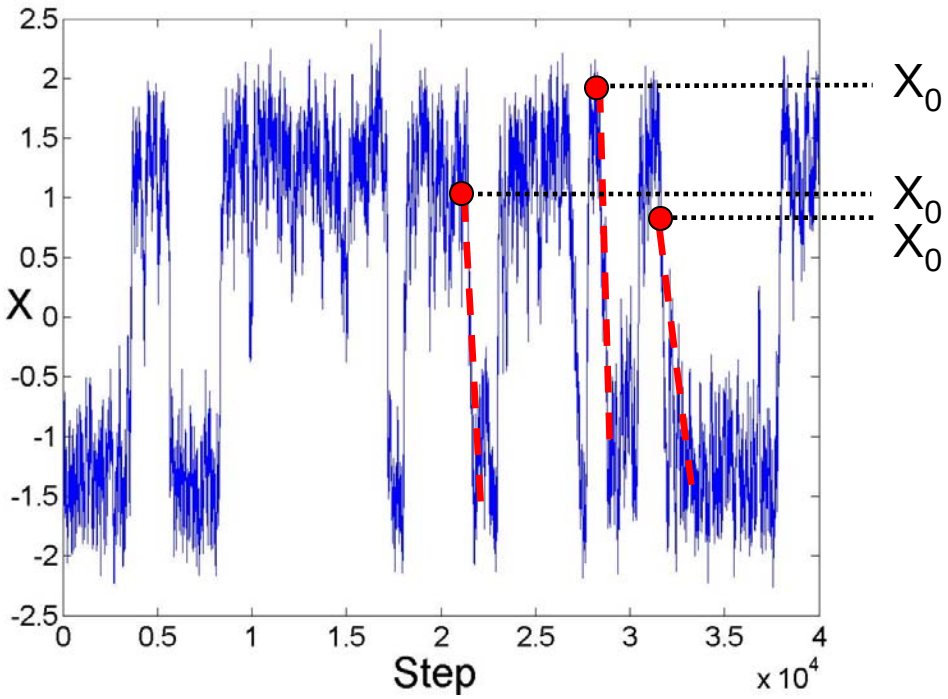
$$v(r) = \frac{\partial \bar{r}(t, r_0)}{\partial t}; \; D(r) = \frac{1}{2} \frac{\partial \sigma^2 (r(t, r_0))}{\partial t}$$

Drift Coefficient

Diffusion Coefficient

Potential: $$\frac{\Phi(r)}{k_B T} = -\int_0^R \frac{v(r')}{D(r')} dr' + \ln D(r) + \text{Const}$$

Coarse Free Energy Calculation



Estimate Drift and Diffusion coefficients numerically from simulation "bursts"

$$v(X) = \frac{\partial \overline{X}(t, X_0)}{\partial t}$$

$$D(X) = \frac{1}{2} \frac{\partial \sigma^2 (X(t, X_0))}{\partial t}$$

Improved estimates using Maximum Likelihood Estimation (MLE)

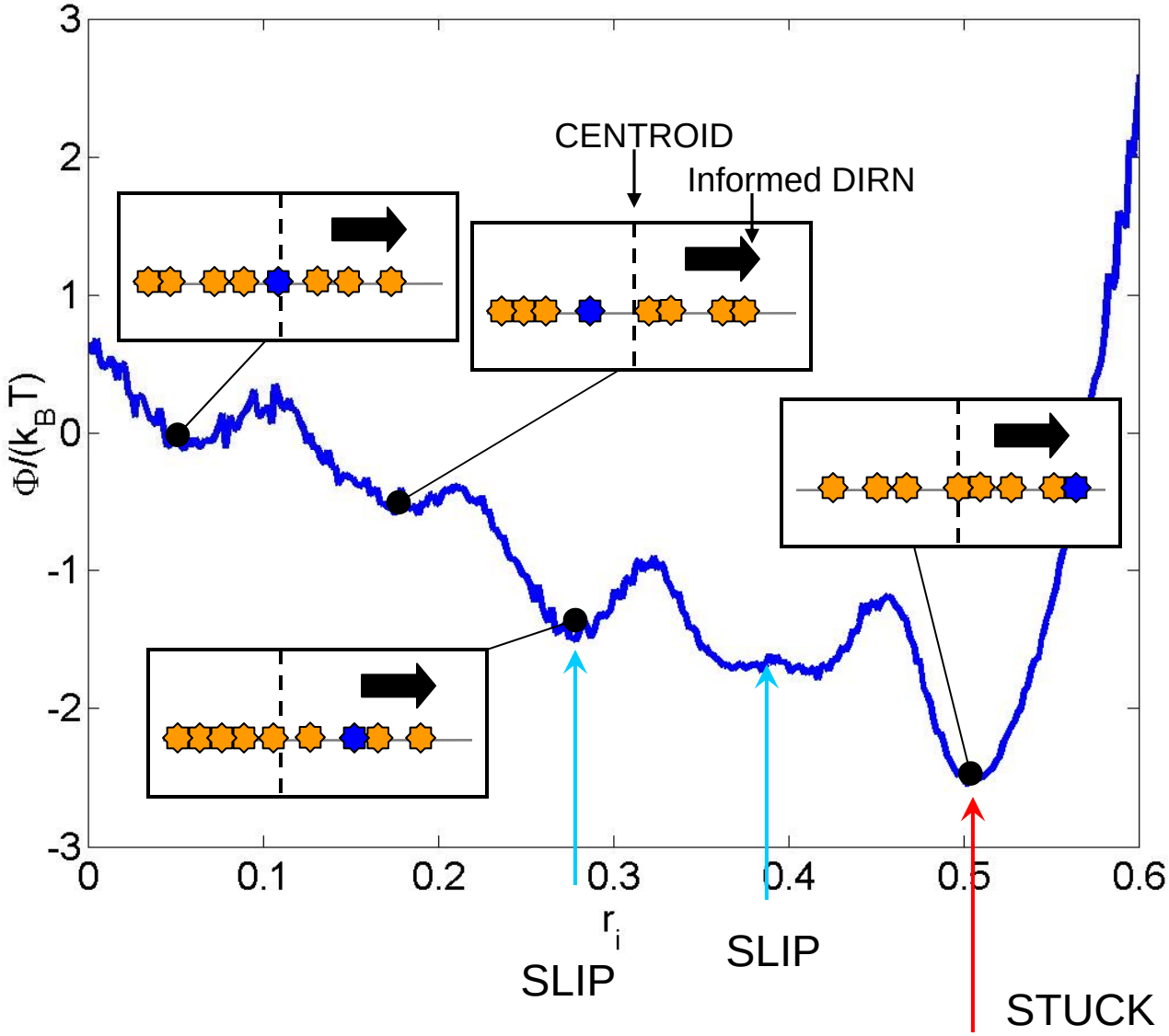
Y. Aït-Sahalia.
Maximum Likelihood Estimation of Discretely Sampled Diffusions: A Close-Form Approximation Approach.
***Econometrica* 70 (2002).**

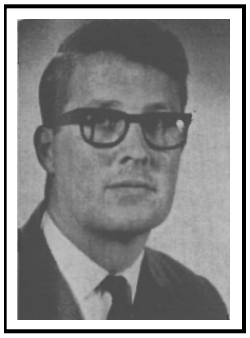
$$\frac{\Phi(X)}{k_B T} = -\int_0^R \frac{v(X')}{D(X')} dX' + \ln D(X) + \mathrm{K}$$

Kopelevich, Panagiotopoulos & Kevrekidis
J Chem Phys 122 (2005)

Hummer & Kevrekidis
J Chem Phys 118 (2003)

Energy Landscape – Fish Swarming Problem





ChE lecture

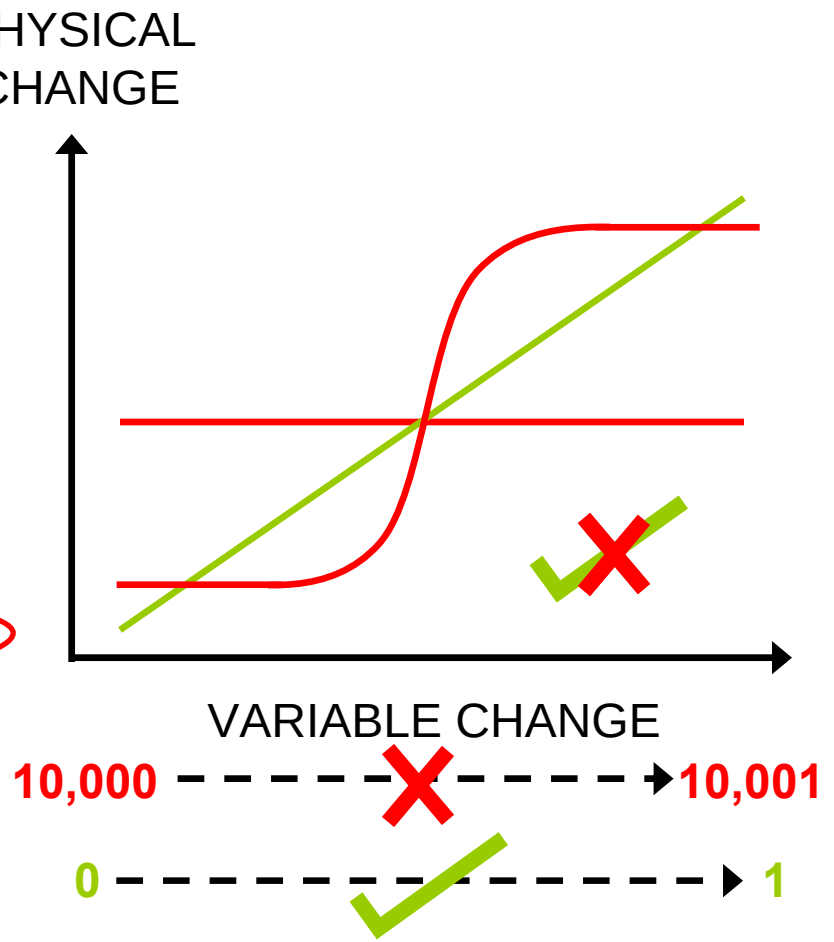
HOW TO GET THE MOST OUT OF AN EQUATION WITHOUT REALLY TRYING

RUTHERFORD ARIS
University of Minnesota
Minneapolis, Minnesota 55455

MAXIMS FOR MATHEMATICAL MODELLING

1. Cast the problem in as elegant a form as possible.
2. Choose a sympathetic notation, but don't become too attached to it.
3. Make the variables dimensionless, since this is the only way in which their magnitudes take on general significance, but do not lose sight of the quantities which may have to be varied later on in the problem nor forget the physical origin of each part.
4. Use a priori bounds of physical or mathematical origin to keep all variables of the same order of magnitude, letting the dimensionless parameters show the relative size of the several terms.
5. Think geometrically. See when you can reduce the number of variables (even at the expense of first treating an over-simplified problem), but keep in mind the needs of the general case.

10. Rearrange the problem. Don't get fixed ideas on what are the knowns and what the unknowns. Be prepared to work with implicit solutions.
11. Neglect small terms, but distinguish between regular and singular perturbations.
12. Use partial insights and despise them not. (E.g. Descarters' rule of signs).
13. These maxims will self-destruct. Make your own!

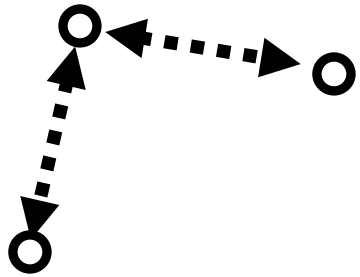


Smooth Change in Problem
↔
Smooth Change in Variables

Using computer to select variable

Rationale:

Lake Carnegie, Princeton, NJ



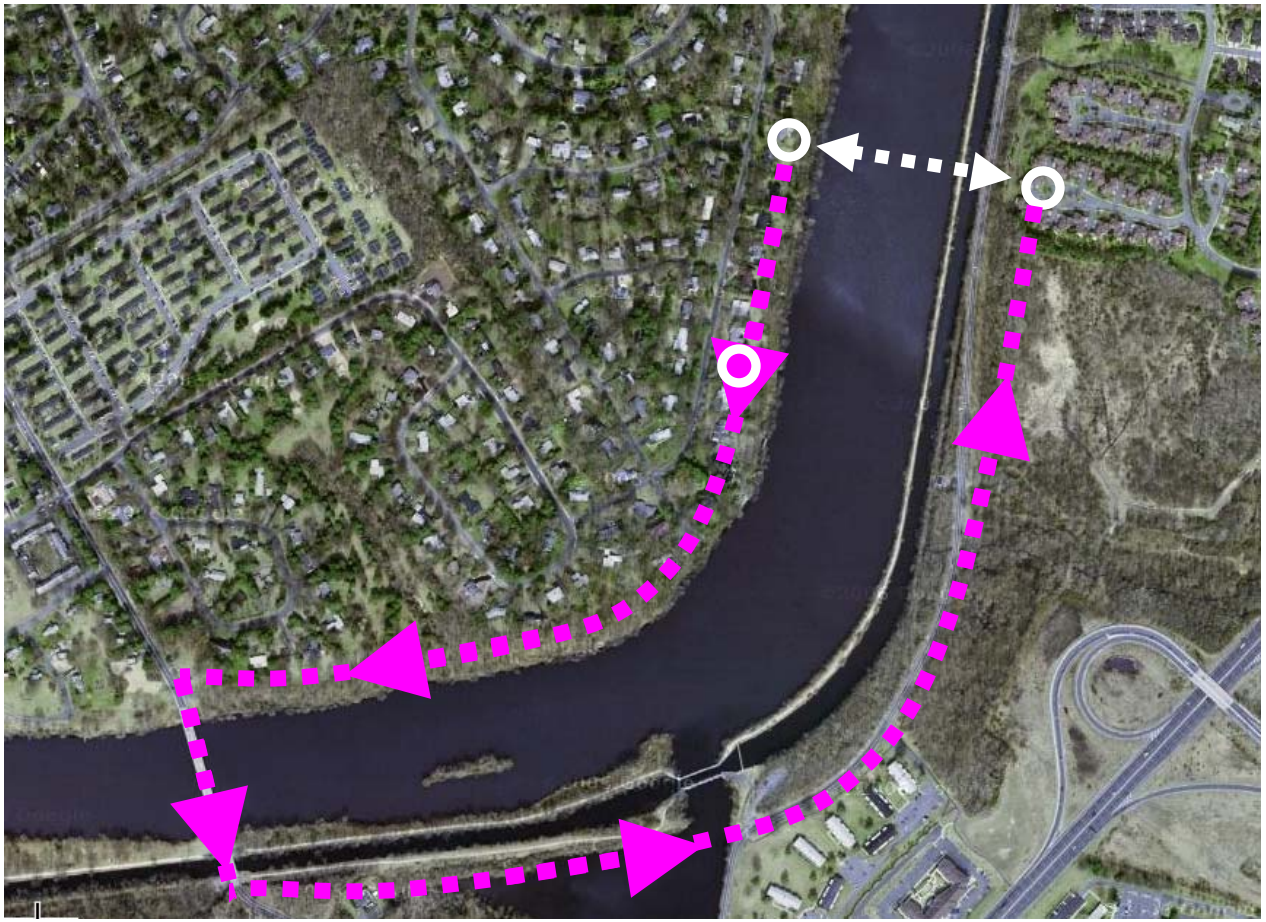
Straight Line Distance

between locations

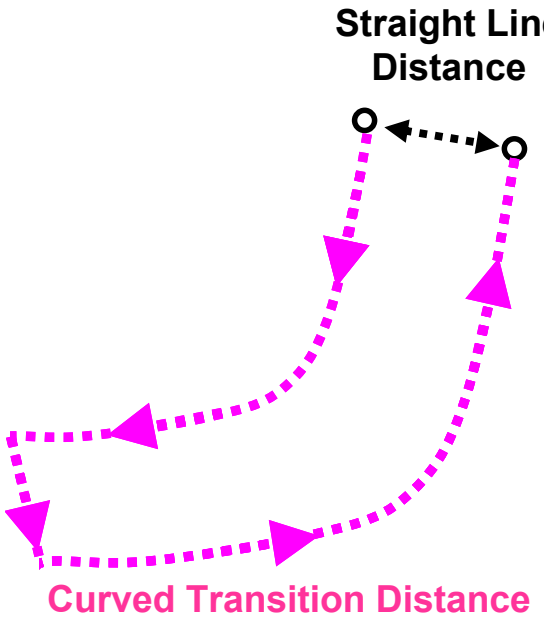
NOT representative of actual transition difficulty\distance

Using computer to select variable

Rationale:



Lake Carnegie, Princeton, NJ

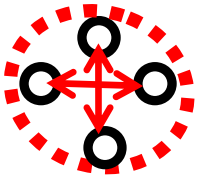


Actual transition difficulty represented by curved path

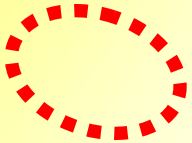
Using the computer to select good variables

Rationale:

Lake Carnegie, Princeton, NJ



Straight Line Distance IS representative of actual transition difficulty\distance



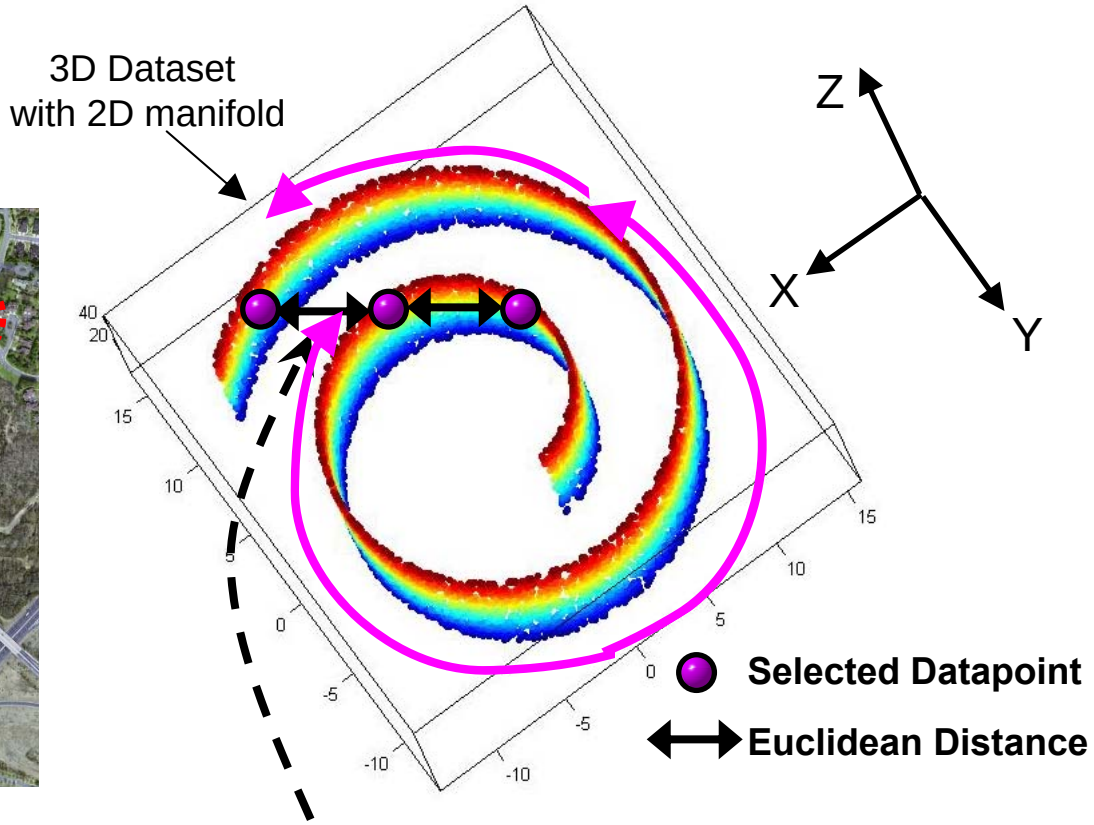
in small LOCAL patches

Patch size related to problem “geography”

Using computer to select variable

Rationale:

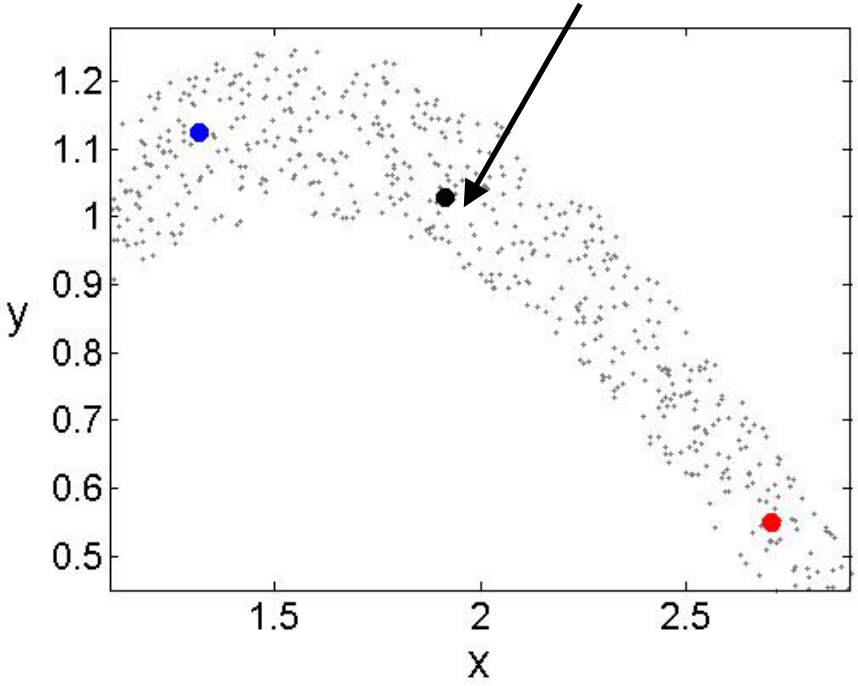
Lake Carnegie, Princeton, NJ

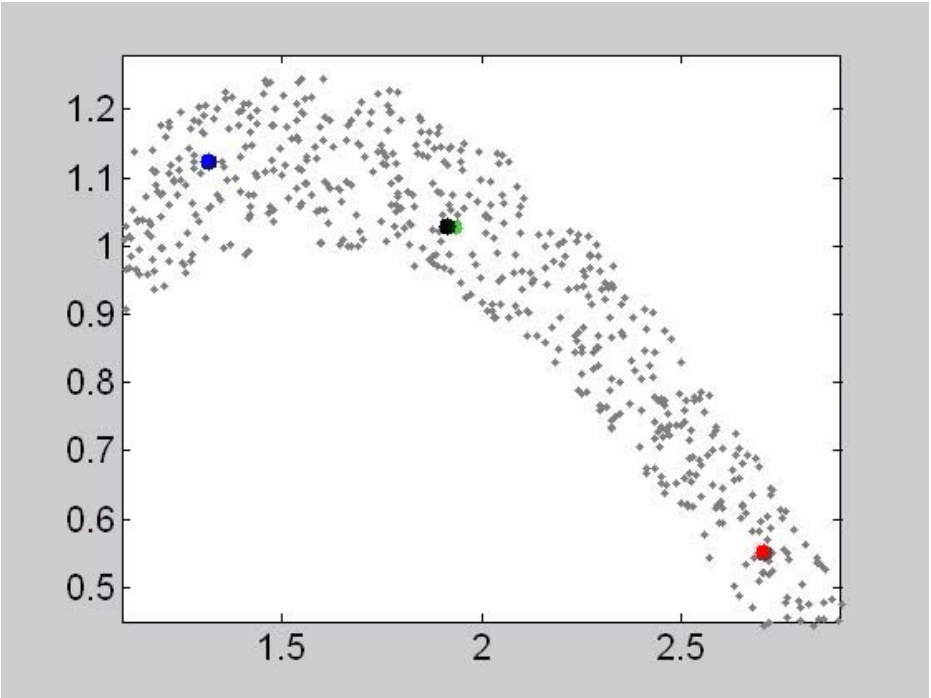


Euclidean distance in input space
may be weak indicator
of INTRINSIC similarity of datapoints

Geodesic distance is good for this dataset

Multiple random walks ● through simulation data initialized at ●

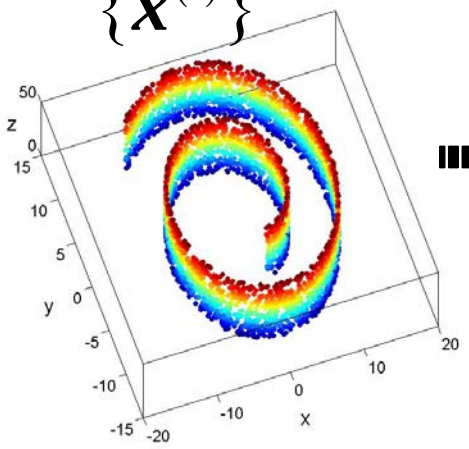




Unequal separation (Euclidean distance) between IC (●) and limits of random walk (●,●)

N datapoints

$\{\boldsymbol{x}^{(i)}\}$



Compute *N*×*N* "neighborhood" matrix *K*

$$K_{i,j} = \exp\left[-\left[\frac{\left\|\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(j)}\right\|}{\sigma}\right]^2\right]$$

Parameter σ
Local neighborhood size

Compute diagonal normalization matrix *D*

$$D_{i,i} = \sum_{j=1}^{N} K_{i,j}$$

Compute Markovian matrix *M*

$$M = D^{-1}K$$

Require: Eigenvalues λ and Eigenvectors **Φ** of *M*

Top $\longrightarrow M\Phi_1 = \lambda_1\Phi_1$

2nd $\longrightarrow M\Phi_2 = \lambda_2\Phi_2$

$\vdots \qquad\qquad \vdots$

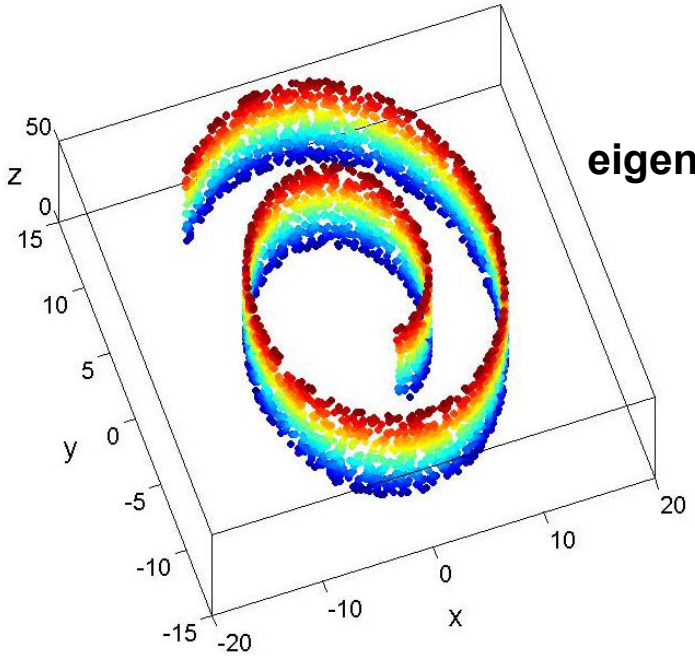
*N*th $\longrightarrow M\Phi_N = \lambda_N\Phi_N$

$$\lambda_1 = 1 > \lambda_2 > \lambda_3 \cdots > \lambda_N$$

A few Eigenvalues\Eigenvectors provide meaningful information on dataset geometry

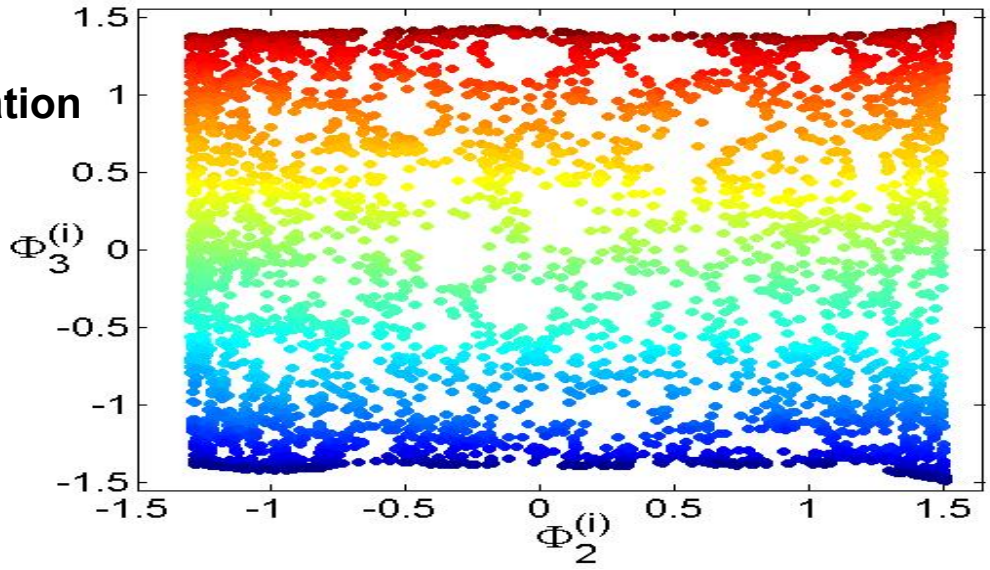
Diffusion Maps

Dataset in x, y, z



eigencomputation

Dataset Diffusion Map



N datapoints

$$\boldsymbol{x}^{(i)} = \left(x_i, y_i, z_i \right), \; i = 1, N$$

N datapoints

$$\boldsymbol{\Phi}^{(i)} = \left(\Phi_2^{(i)}, \Phi_3^{(i)} \right), \; i = 1, N$$

R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps.
***PNAS* 102 (2005).**

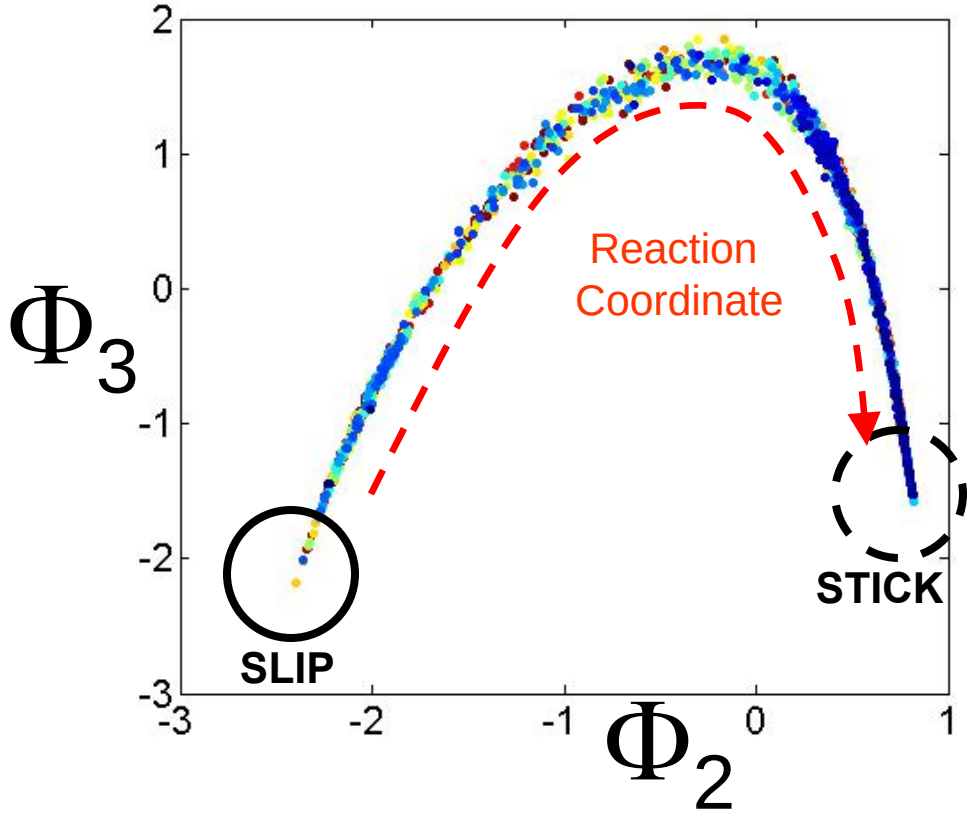
B. Nadler, S. Lafon, R. Coifman, and I. G. Kevrekidis, Diffusion maps, spectral clustering and reaction coordinates of dynamical systems.
***Appl. Comput. Harmon. Anal.* 21 (2006).**

Diffusion Map (Φ_2, Φ_3)

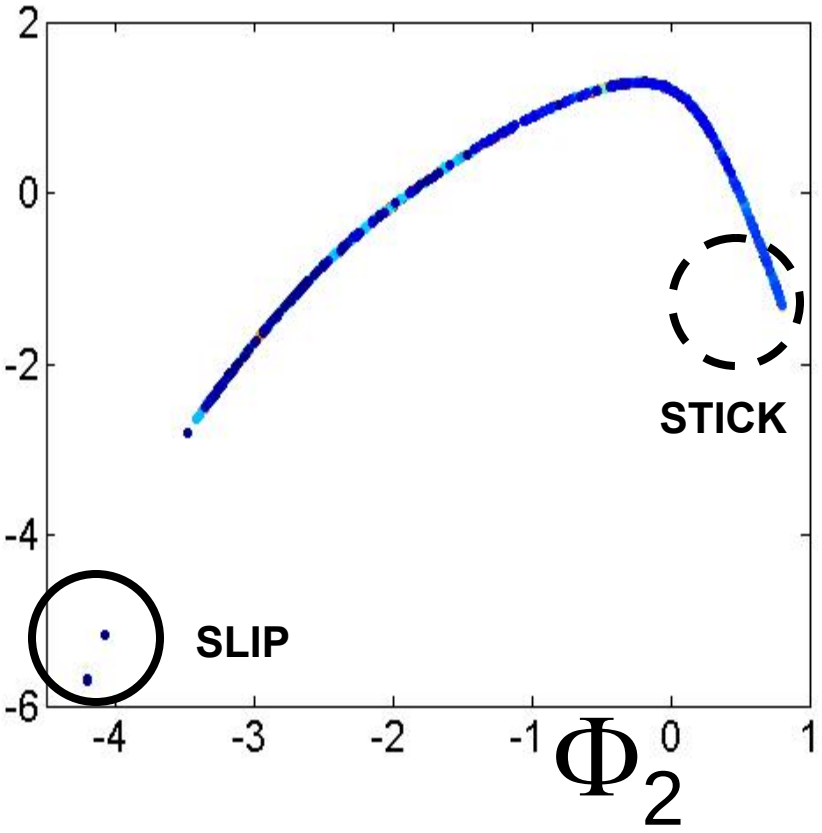
Report absolute distance
of all uninformed individuals
to informed individual to DMAP routine

Report (signed) distance
of all uninformed individuals
to informed individual to DMAP routine

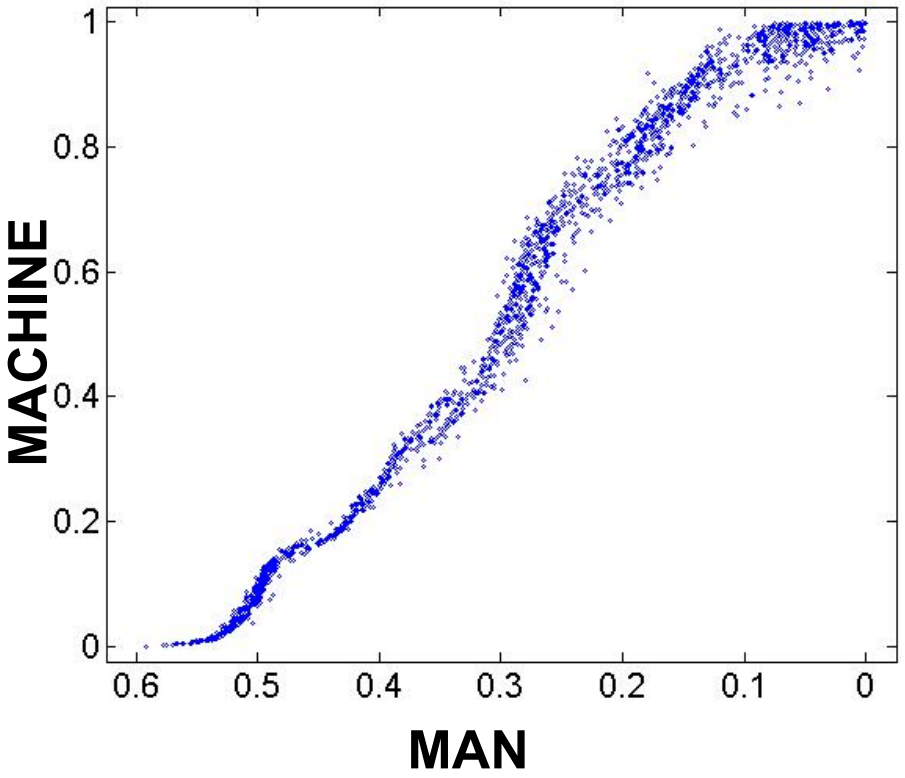
ABSOLUTE Coordinates



SIGNED Coordinates



ABSOLUTE Coordinates



SIGNED Coordinates