NAE/FOE Irvine– September 2004

Equation-free Modeling For Complex Systems

or

Enabling Microscopic Simulators to perform System Level Tasks

or

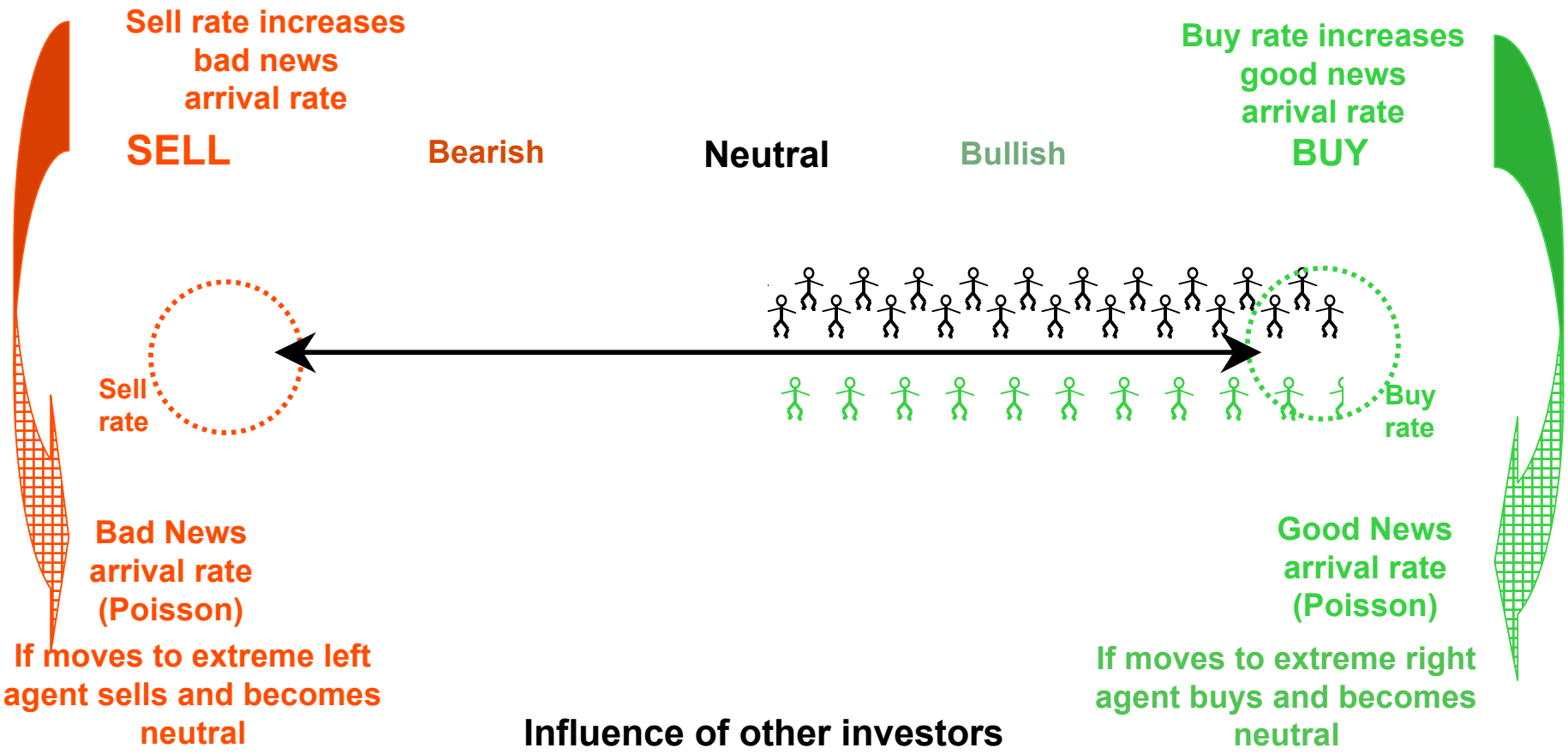
Solving Differential Equations Without the Equations

Or

A Systems Approach to Multiscale Simulations

I. G. Kevrekidis -C. W. Gear and several other good people-

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



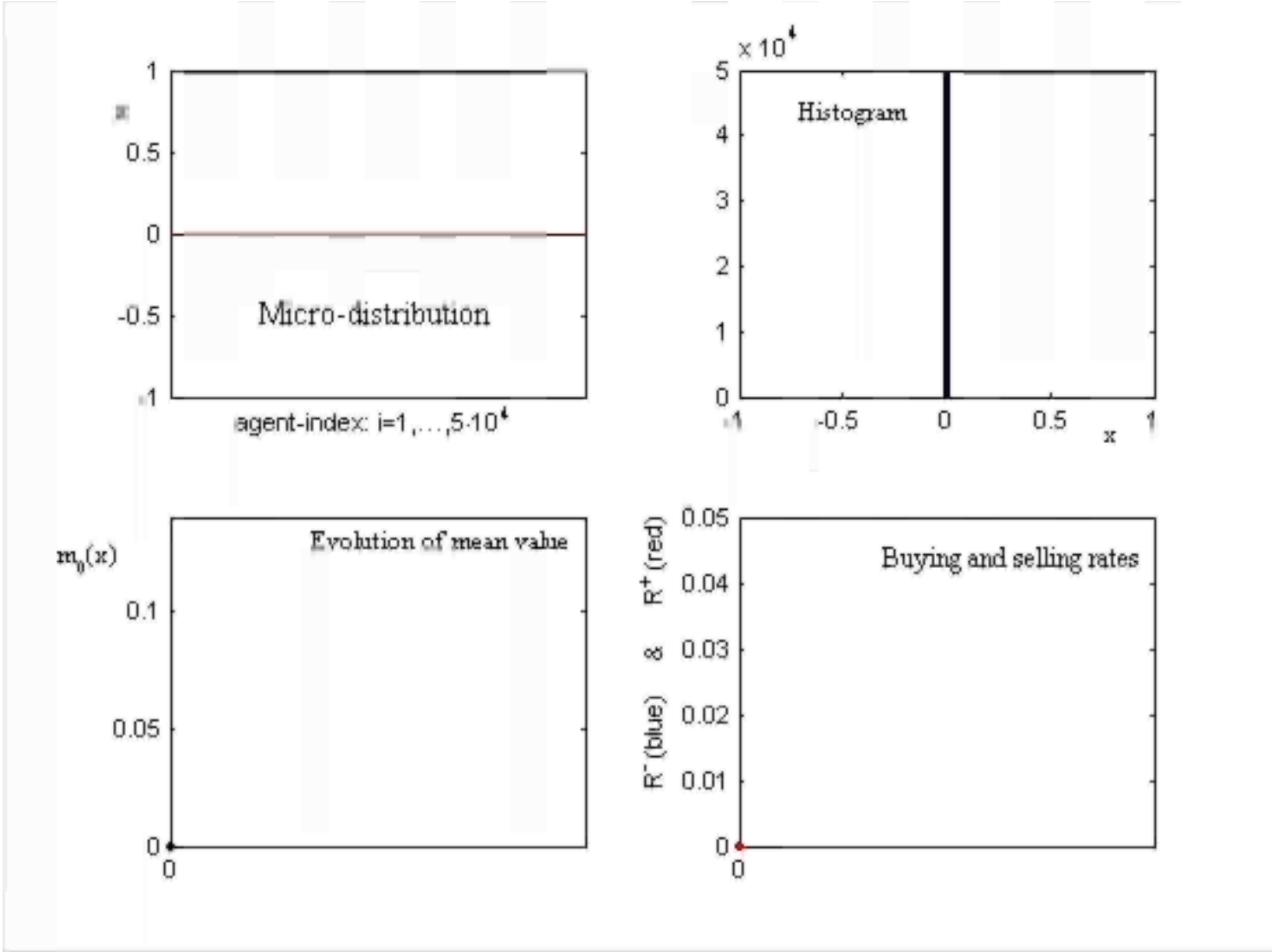
Influence of other investors

- or lemmings falling off a cliff

SIMULATION RESULTS

50000 agents, $\varepsilon^{+}=0.075$, $\varepsilon^{-}=-0.072$, $v_0^{+}=v_0^{-}=20$

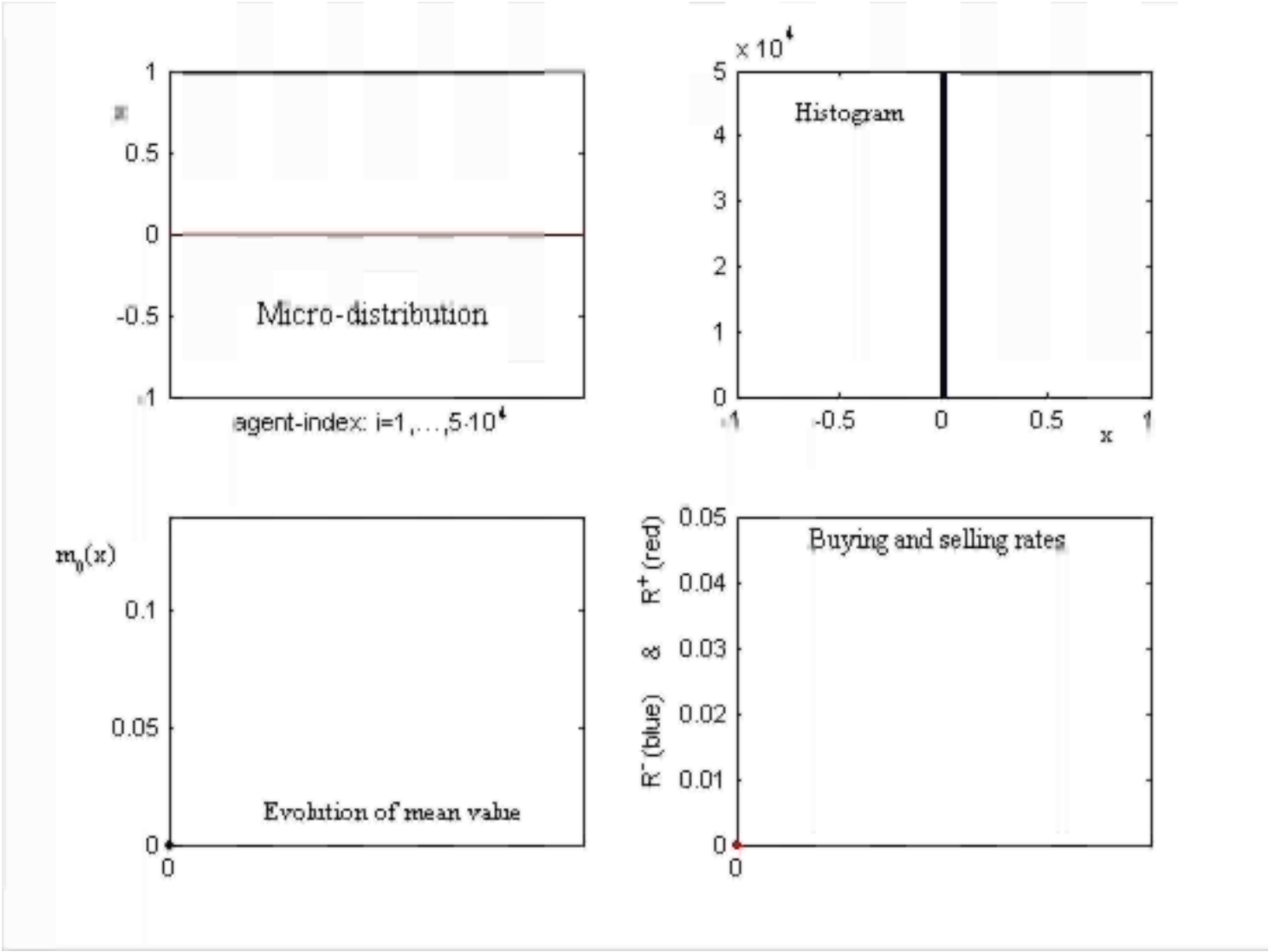
Open loop response, g = 42



SIMULATION RESULTS

50000 agents, $\varepsilon^{+}=0.075$, $\varepsilon^{-}=-0.072$, $v_0^{+} = v_0^{-}=20$

Open loop response, g = 46.5



Loose thoughts about multiscale /complex systems and their modeling

- Our example: interaction of many “units” between themselves and with their environment --- “emergent” behavior.
- “Physics” at a *fine* level, questions asked at a *coarse-grained* level

- This kind of complexity vs. The complexity of a commercial airliner

- Is laminar fluid flow “simple” ?
- Two billiard balls colliding elastically – “simple” or “complex” ?

Multiscale / Complex System Modeling

"Textbook" engineering modeling:

macroscopic behavior *through* macroscopic models

(e.g. conservation equations augmented by closures)

Alternative (and increasingly frequent) modeling situation:

- Models
  - at a **FINE** / ATOMISTIC / STOCHASTIC level
  - MD, KMC, BD, LB (also CPMD…)
- Desired Behavior
  - At a **COARSER** Macroscopic Level

What I will tell you:

Solve the equations WITHOUT writing them down.

Write "software wrappers" around "fine level" microscopic codes

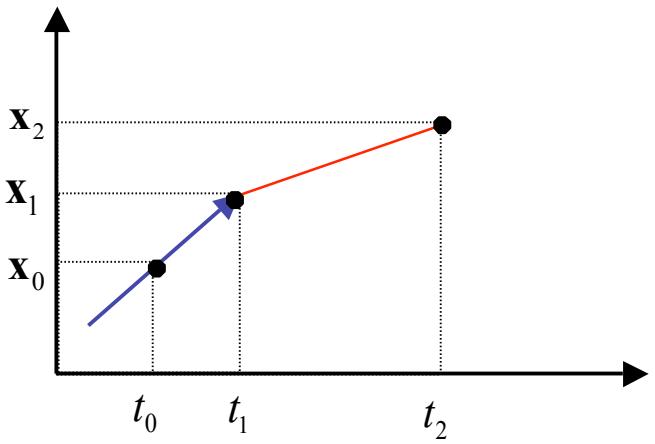
Top level: **all algorithms we know and love (e.g. AUTO)**
Bottom level: MD, kMC, LB, BD, heterogeneous/ discrete media, CPMD, hybrid
INTERFACE:

Trade Function Evaluation for "on demand" experimentaton and estimation

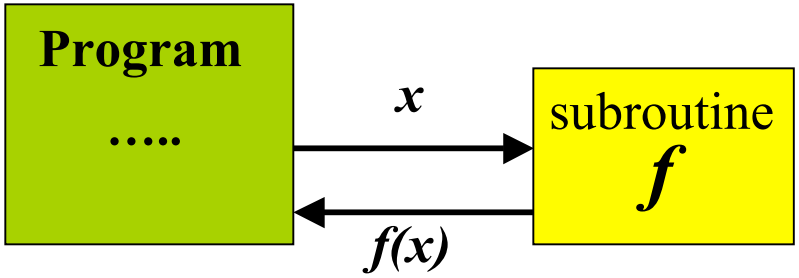
Think of the microscopic simulator AS AN EXPERIMENT That you can set up and run at will

"Equation Free" (motivated by "matrix free iterative linear algebra")
Algorithms (coarse integration, patch dynamics, coarse RPM…)
Tasks (stability/ bifurcation, control, optimization, dynamic renormalization)
Examples (LB, KMC, BD, MD), ***and some nebulous thoughts***

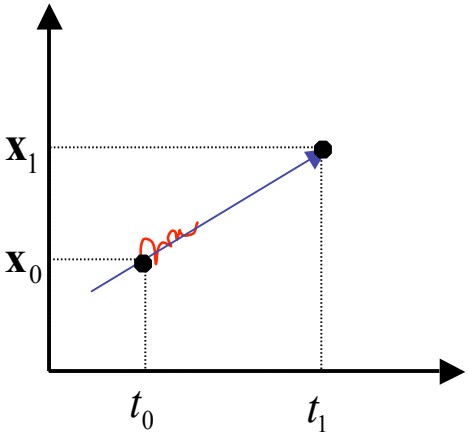
Equations $\frac{d\,\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x})$



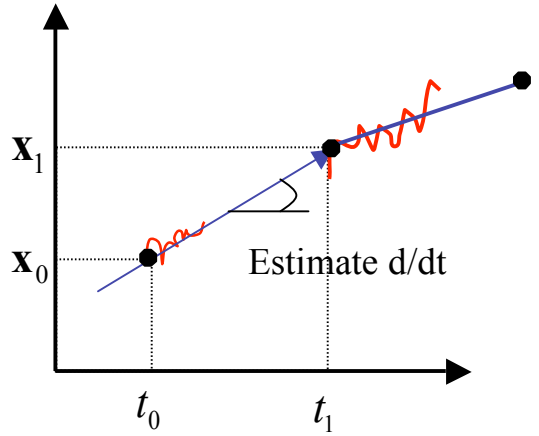
Determinism (in a practical way)



Experimental Experimentalist



Legacy Code
Computational Experimentalist

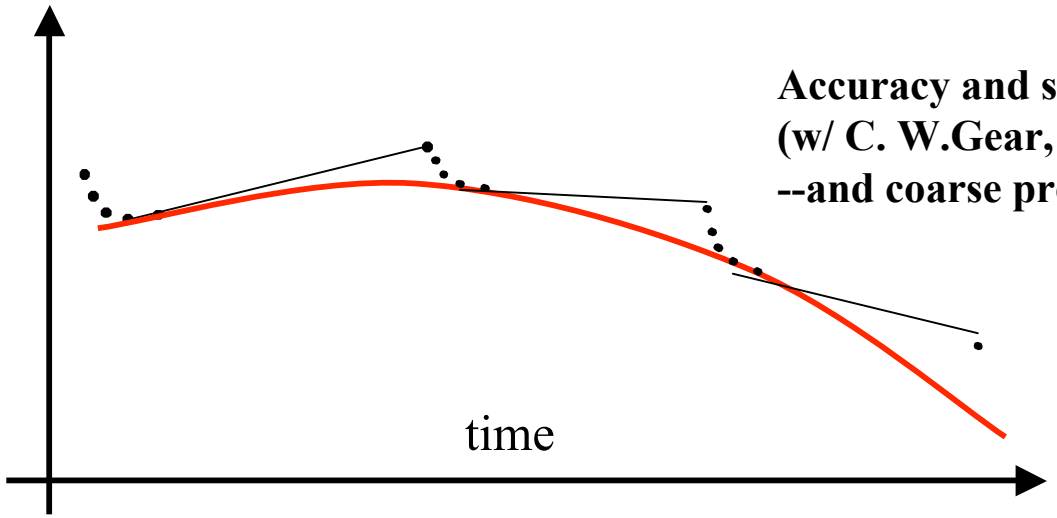


IN EFFECT, Do Forward Euler
not with the EQUATION, but with the (computational) EXPERIMENT

Look at the experiment & RESTART IT

Projective Integration - a sequence of outer integration steps

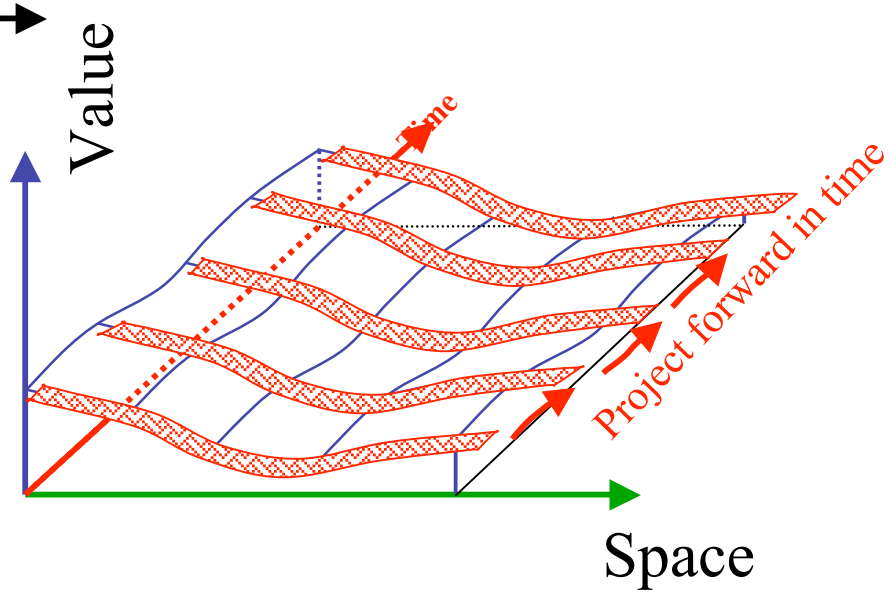
based on inner simulator + estimation (stochastic inference)



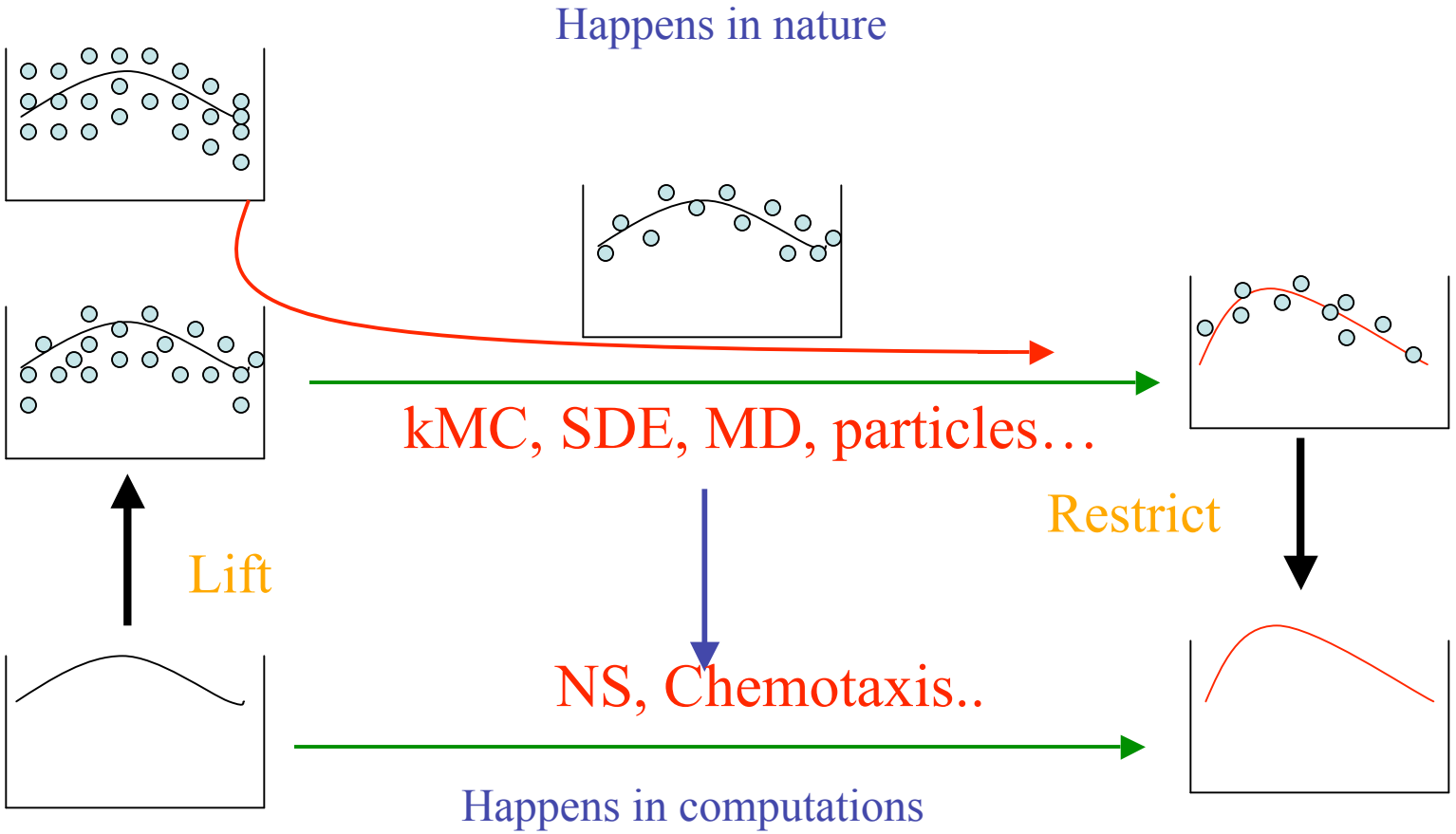
Accuracy and stability of these methods – NEC/TR 2001
(w/ C. W.Gear, *SIAM J.Sci.Comp.* 03, *J.Comp.Phys.* 03,
--and coarse projective integration (inner LB)
Comp.Chem.Eng. 2002

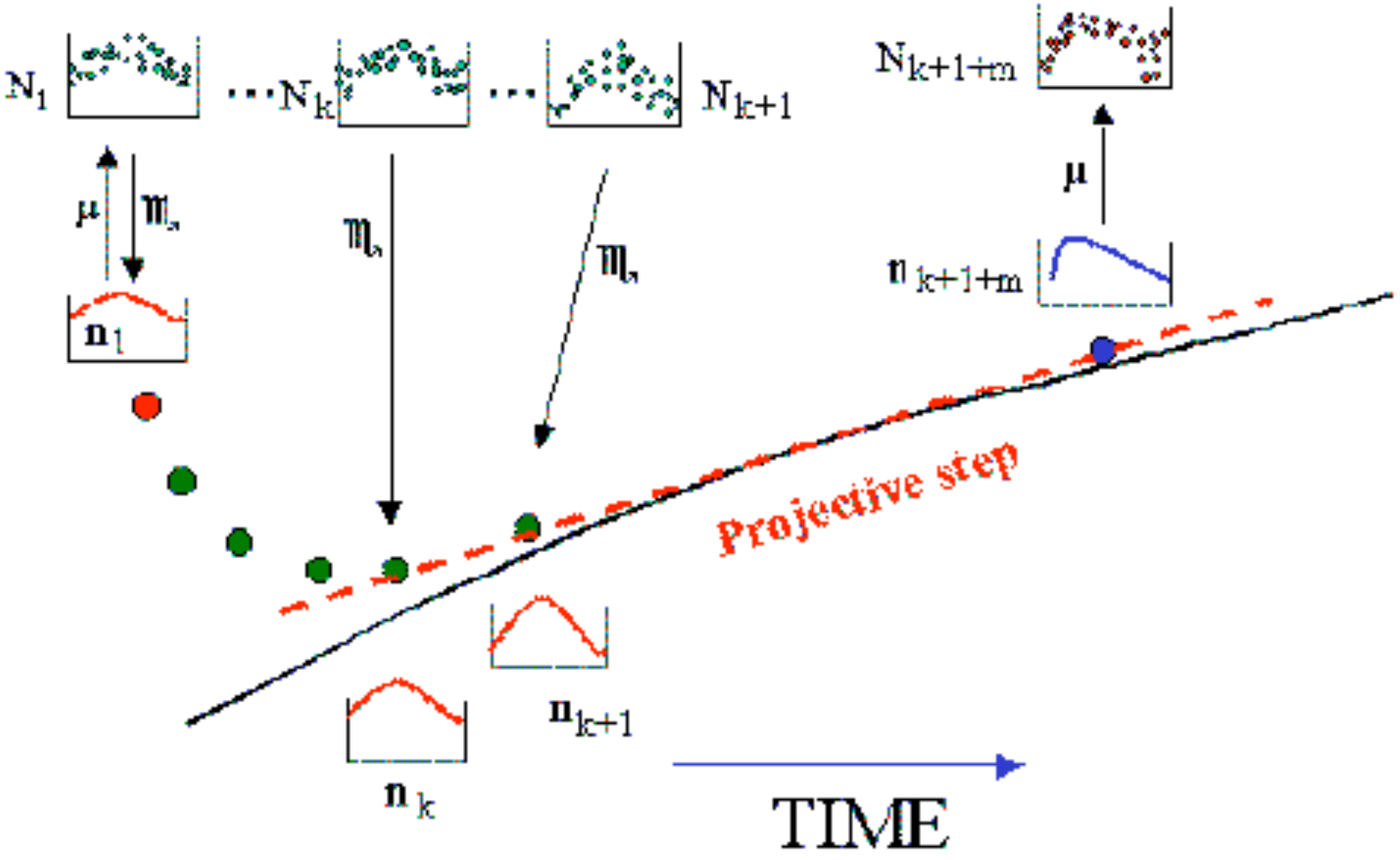
Projective methods in time:

-perform detailed simulation for short periods
or use existing/legacy codes
- and then *extrapolate* forward over large steps



Coarse Behavior




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

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.

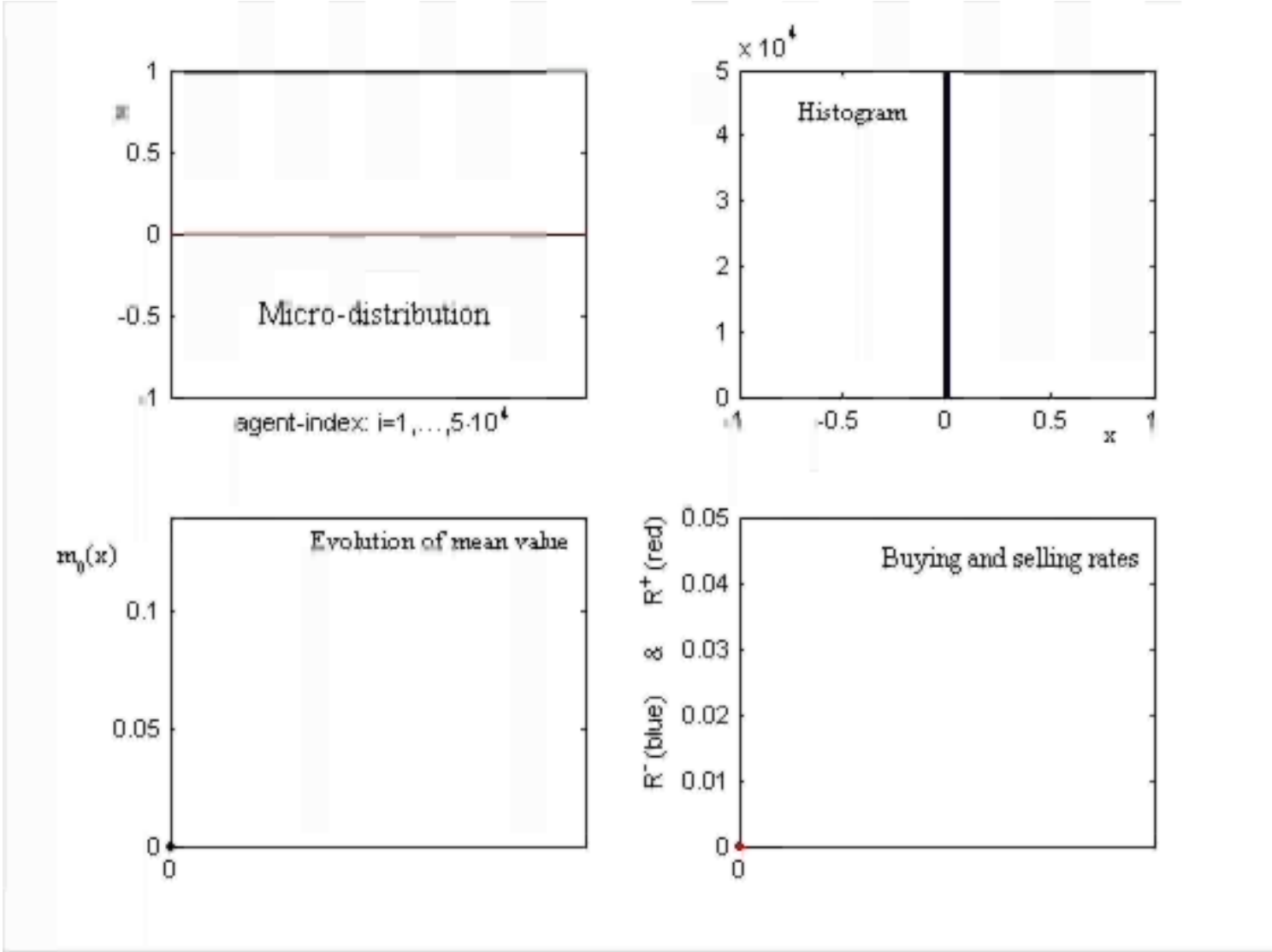
We do the step-by-step simulation in the high-dimensional description.

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

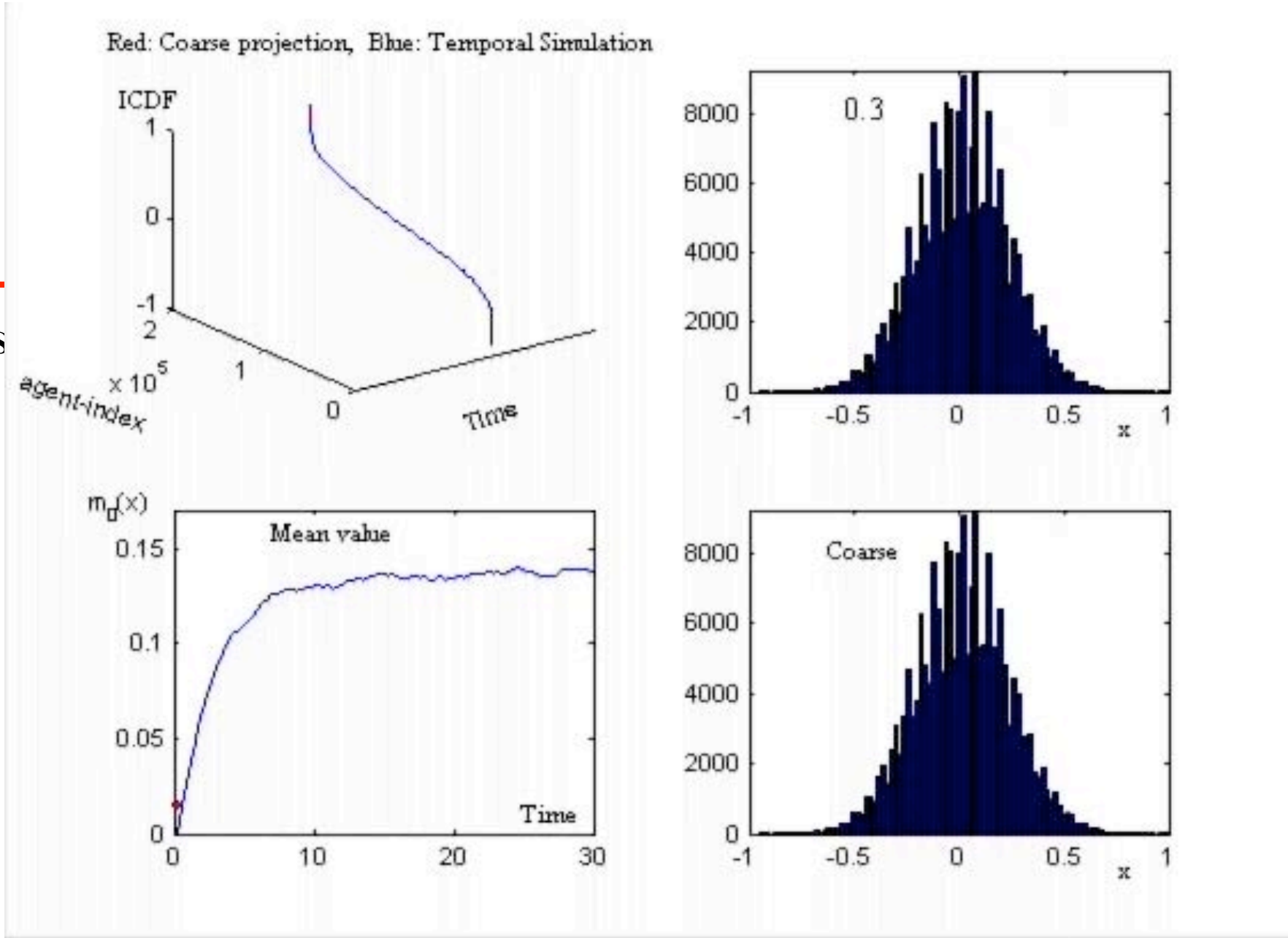
SIMULATION RESULTS

50000 agents, $\varepsilon^{+}=0.075$, $\varepsilon^{-}=-0.072$, $v_0^{+}=v_0^{-}=20$

Open loop response, g = 42



Coarse projective integration: Accelerating things

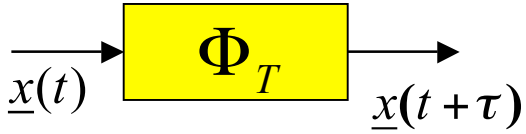


THE CONCEPT: What else can I do with an integration code ?

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

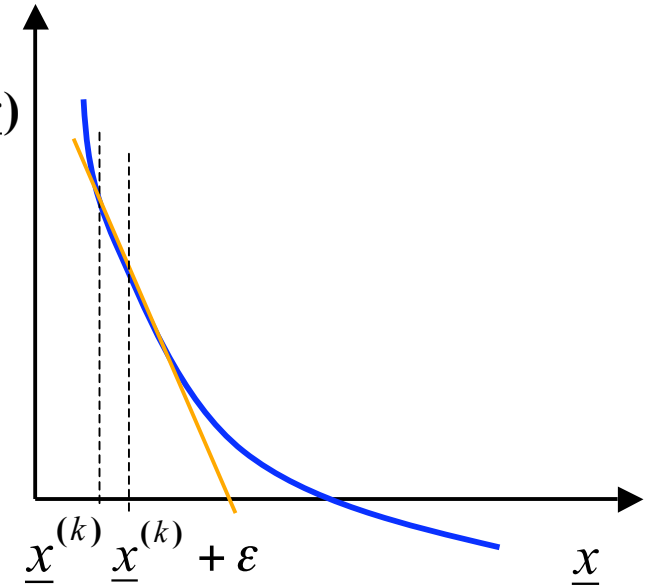
Write Simulation

Compile



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

Do Newton



Estimate

matrix-vector product

⇩

Matrix free iterative linear algebra

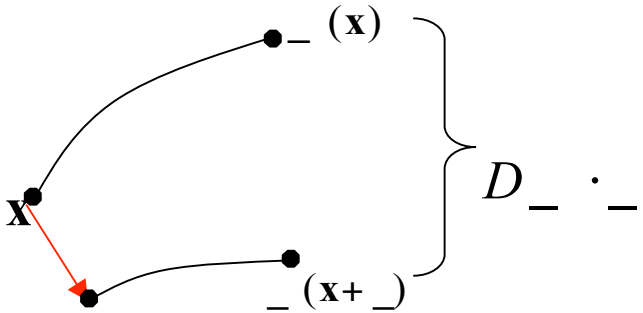
⇩

The World

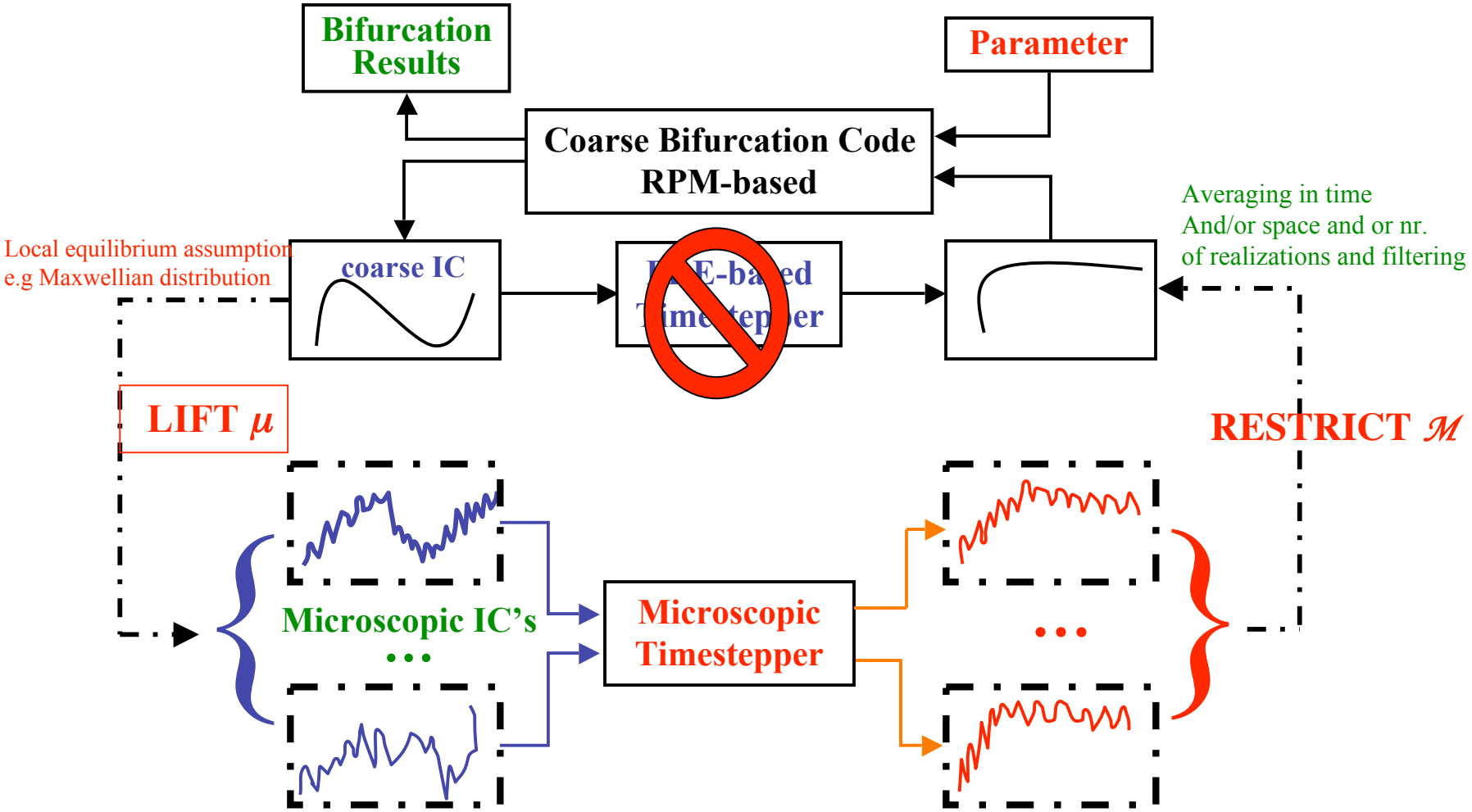
⟺

CG, GMRES Newton-Krylov

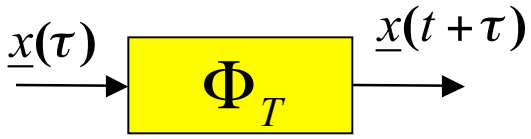
Also



"Coarse" Bifurcations



The Bifurcation Diagram

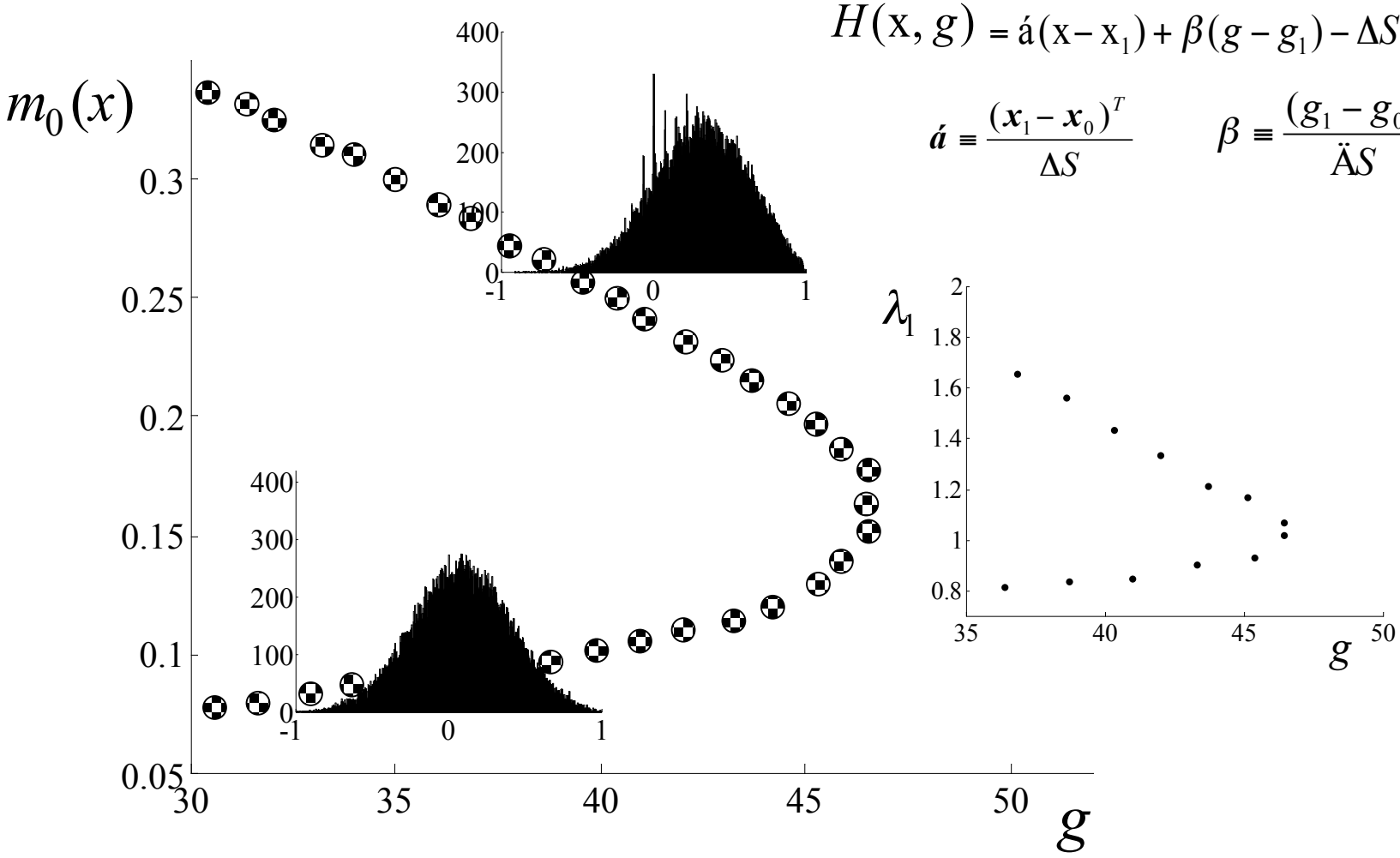


$$\underline{x} - \Phi(\underline{x}, g) = 0$$

Tracing the branch with arc-length continuation

$$H(\mathrm{x}, g) = \acute{a}(\mathrm{x} - \mathrm{x}_1) + \beta(g - g_1) - \Delta S = 0$$

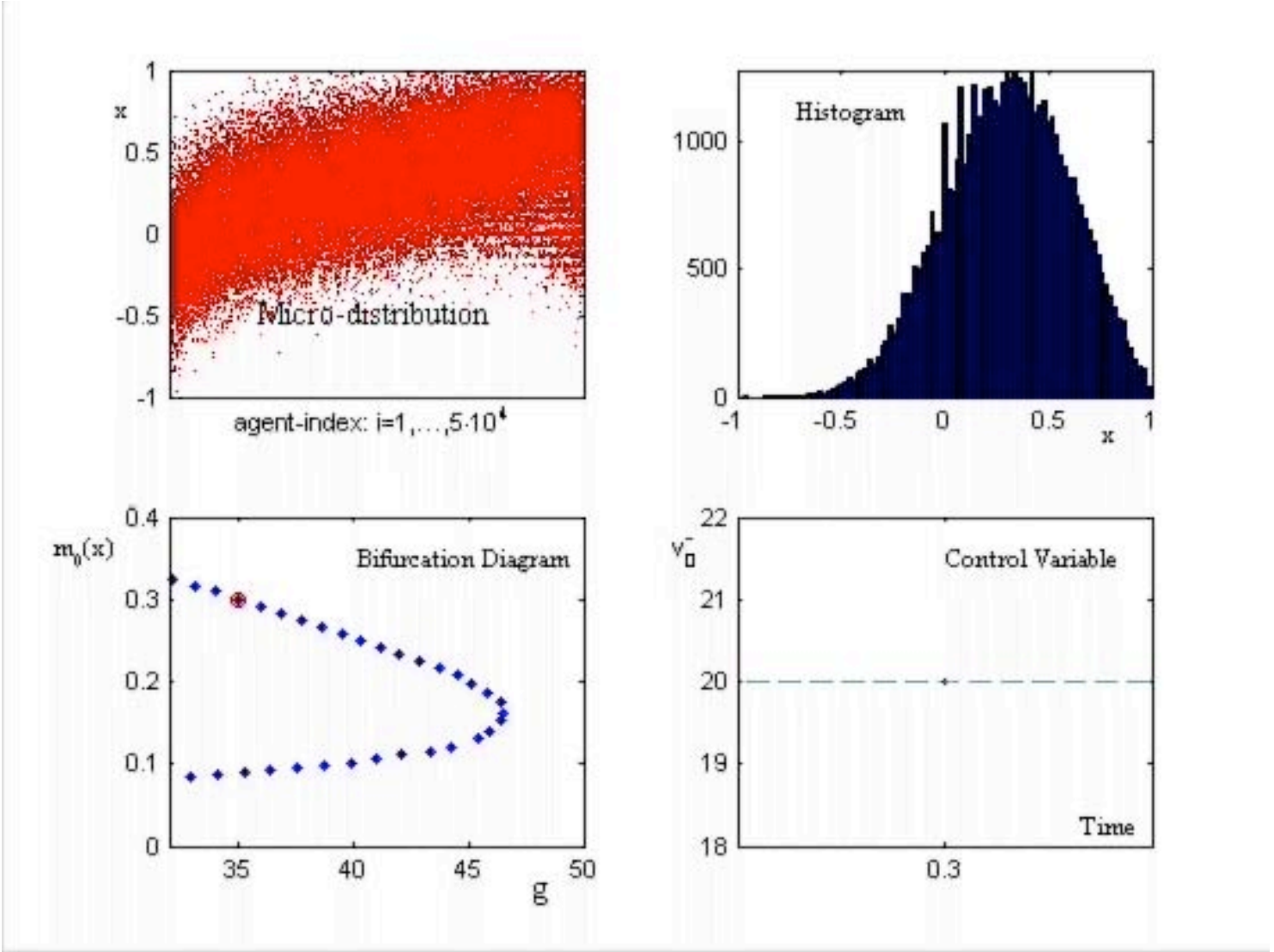
$$\acute{a} \equiv \frac{(\boldsymbol{x}_1 - \boldsymbol{x}_0)^T}{\Delta S} \qquad \beta \equiv \frac{(g_1 - g_0)}{\ddot{A}S}$$



The Bifurcation Diagram

50000 agents, g=35, ε^{+}=0.075, ε^{-}=-0.072, v_0^{+}=v_0^{-}=20

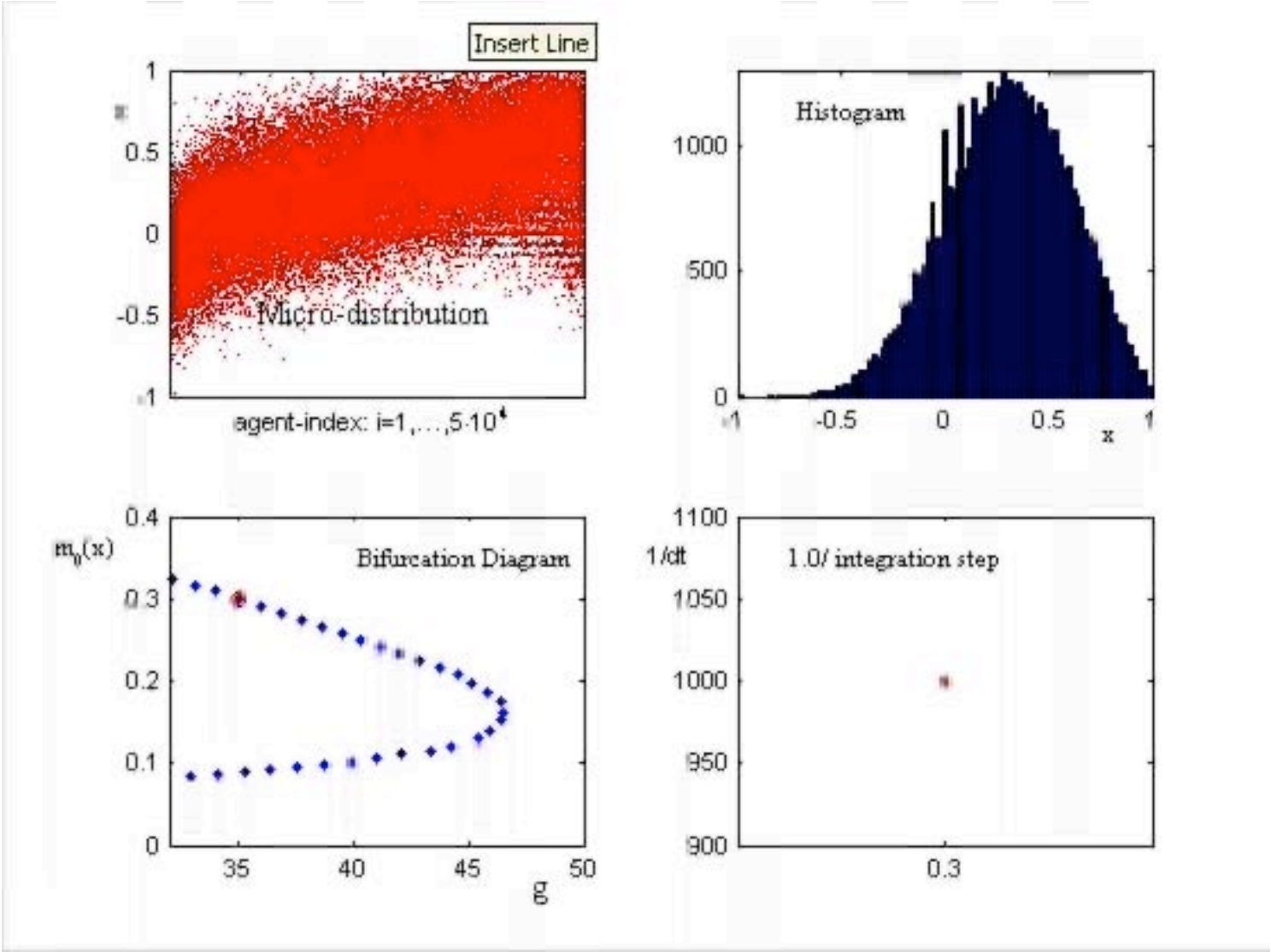
Open loop response. From unstable to stable markets



The Bifurcation Diagram

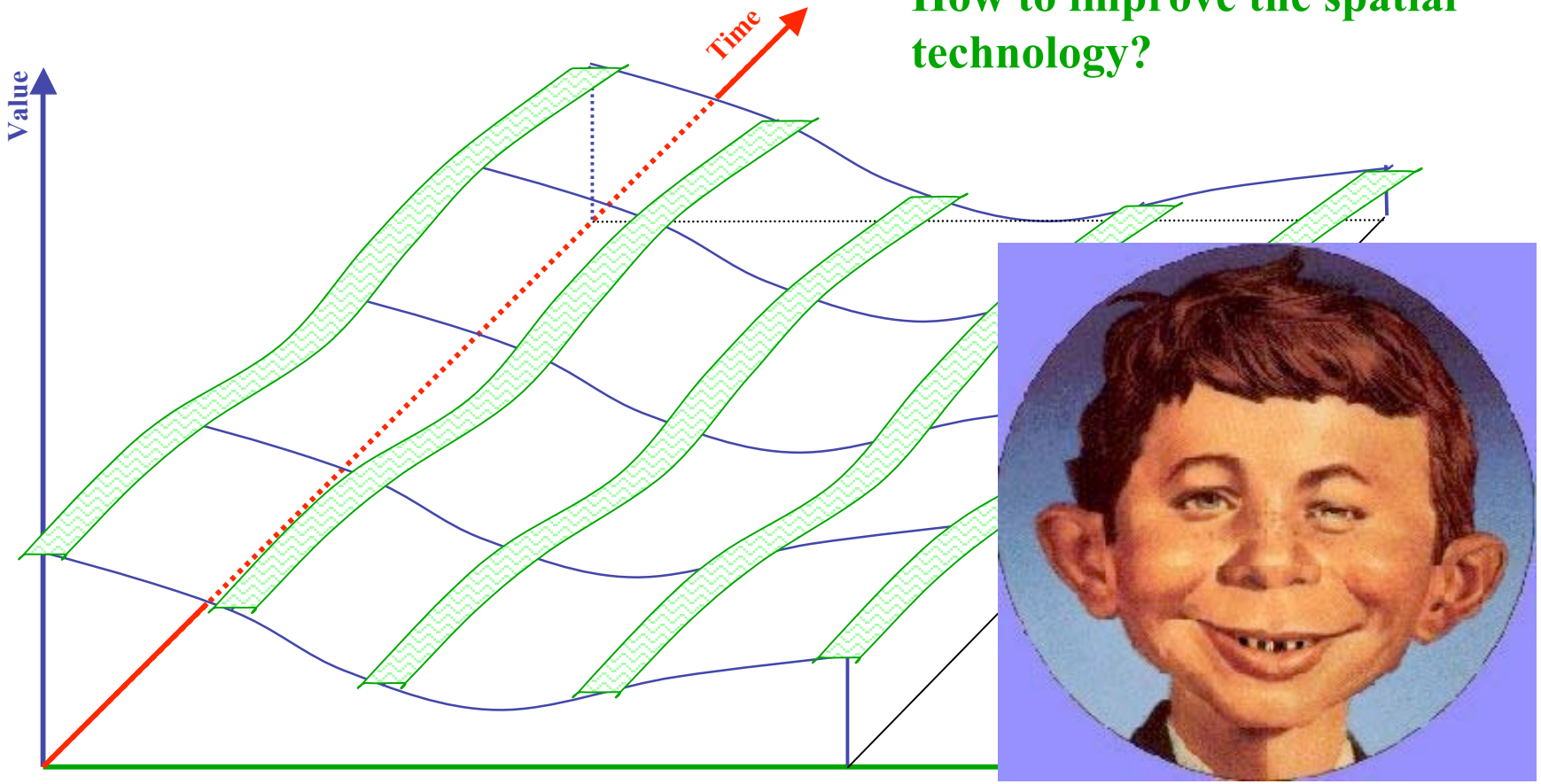
50000 agents, g=35, ε^{+}=0.075, ε^{-} = -0.072, $v_0^{+} = v_0^{-} = 20$

Open loop response. Blow up



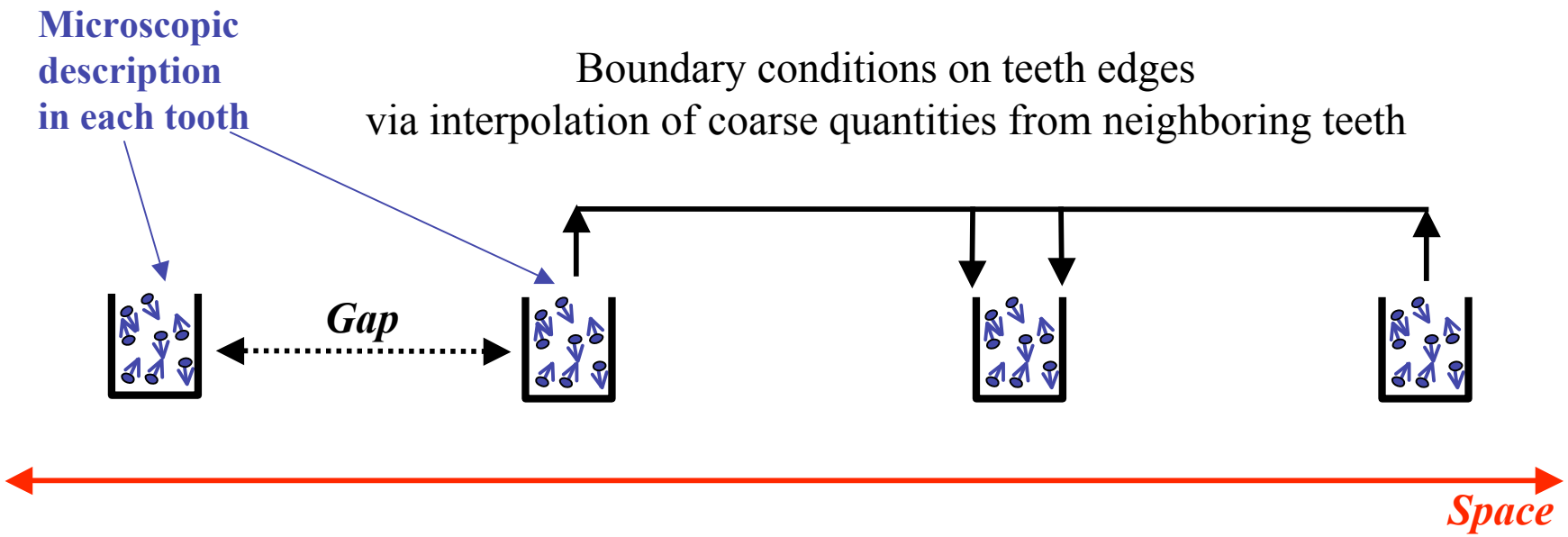
Multiscale Modeling Challenges:

How to improve the spatial technology?



Proposal: detailed modeling in small spatial boxes with interpolation between boxes - the "gap-tooth scheme"

Gap-Tooth Scheme

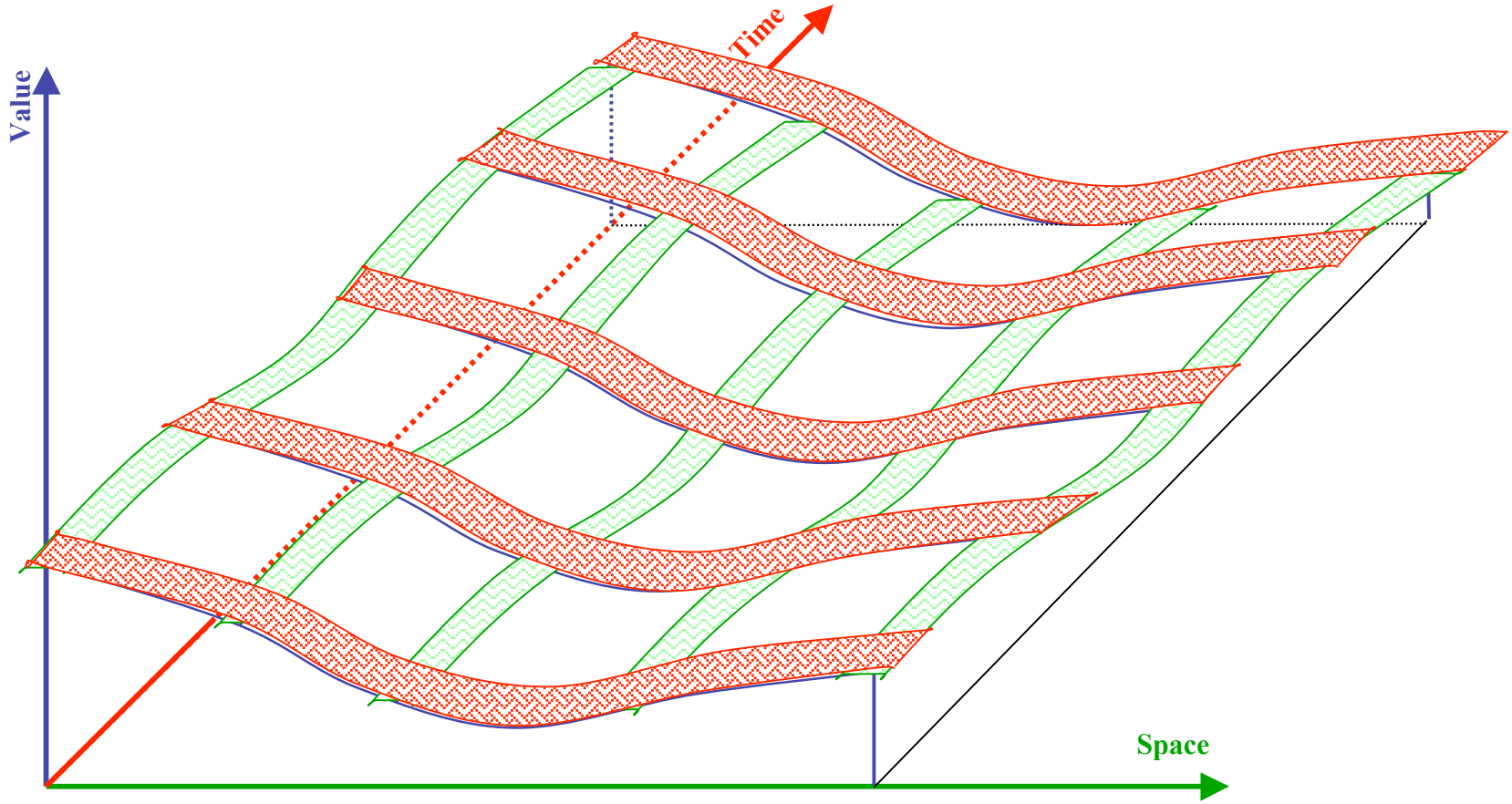


Ways to impose "coarsely inspired" boundary conditions

Motivated from Li & Yip, 1998: *Kevrekidis et al.,* **nlin.CD/0302055 at arXiv.org**

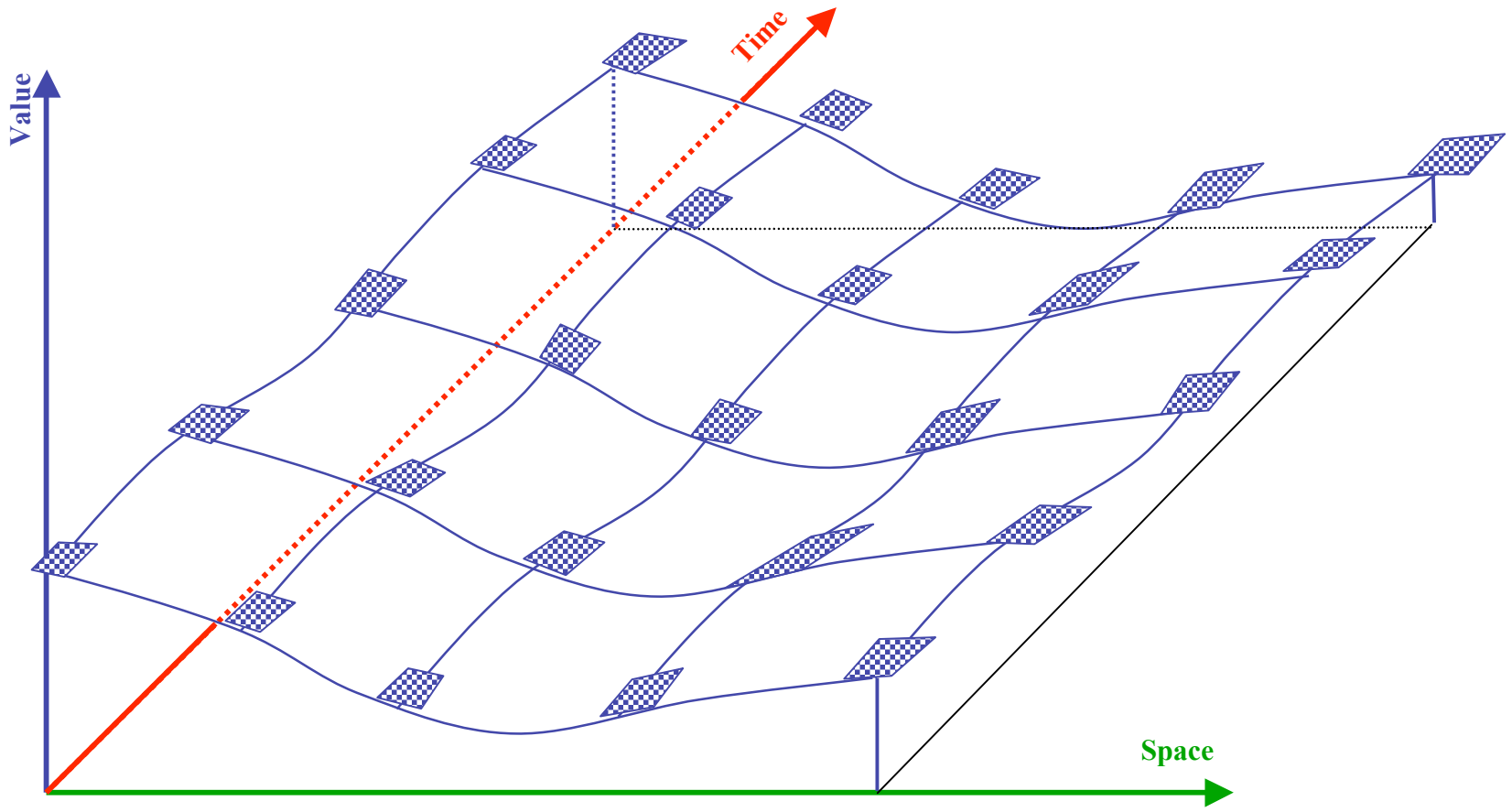
Gear, Li and Kevrekidis, **physics/0303010 at arXiv.org / PLA**

Multiscale Modeling Challenges:



Can we combine gap tooth with projective integration in time?

Multiscale Modeling Challenges:

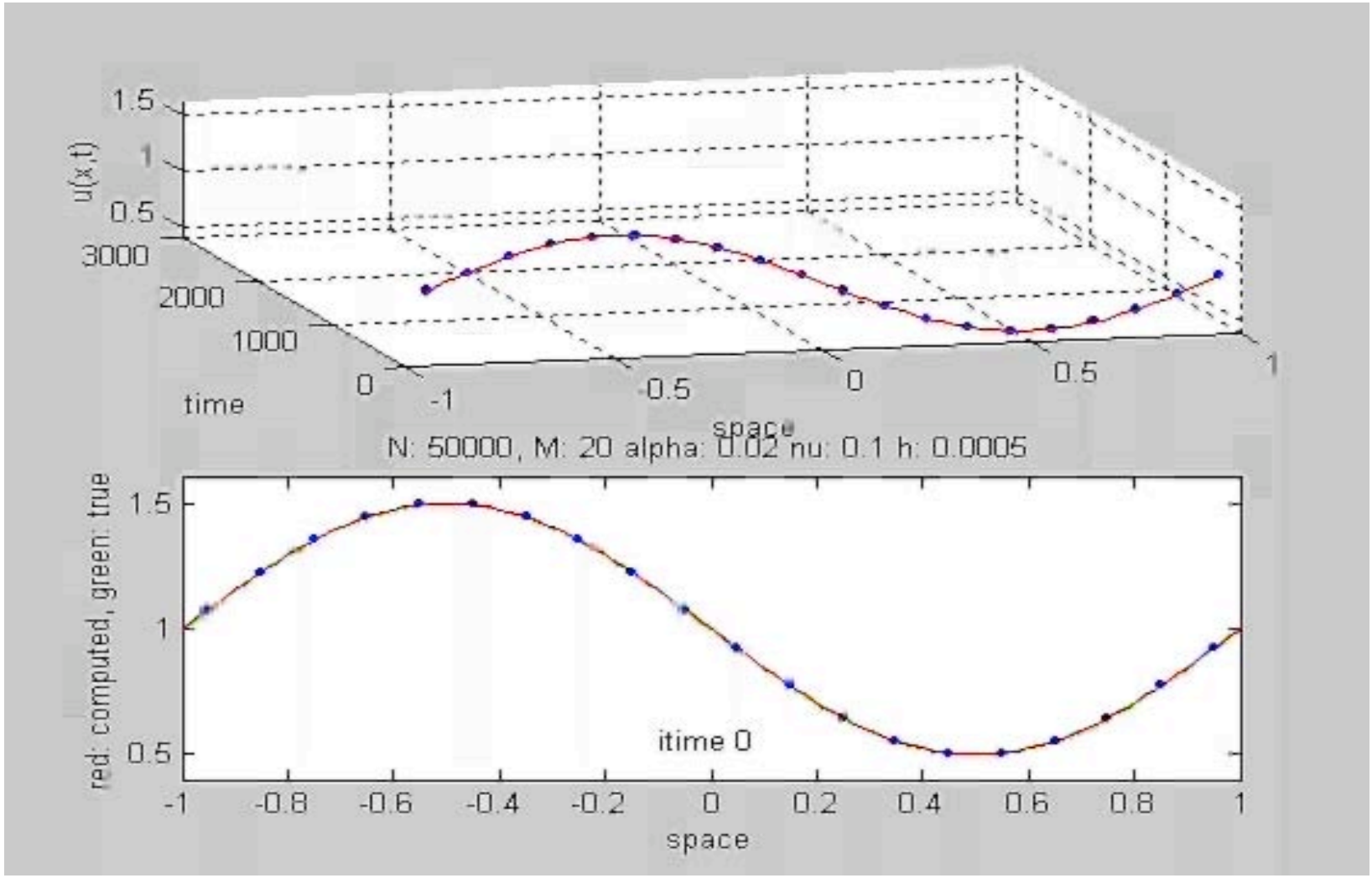


The "action" is going on at the intersection of the strips

- these are "microscopic" elements and, by interpolation and extrapolation, they are patched together over the full region

Viscous Burgers equation: kMC Realization

$$u_t + uu_x = vu_{xx}$$



STABILIZING UNSTABLE M*****S

Feedback controller design

We consider the problem of stabilizing an equilibrium $\boldsymbol{x}^*, p^*$ of a dynamical system of the form

$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), p(t))$ $\boldsymbol{f}: R^N \times R \rightarrow R^N$ where $\boldsymbol{f}$ and hence $\boldsymbol{x}^*$ is characterized of uncertainty

To do this the dynamic feedback control law is implemented: $p = p^* + \boldsymbol{K}(\boldsymbol{x} + \boldsymbol{D}\boldsymbol{w})$

Where $\boldsymbol{w}$ is a M-dimensional variable that satisfies $\dot{\boldsymbol{w}} = \boldsymbol{x} + \boldsymbol{D}\boldsymbol{w}$

Choose matrices $\boldsymbol{K}, \boldsymbol{D}$ such that the closed loop system is stable

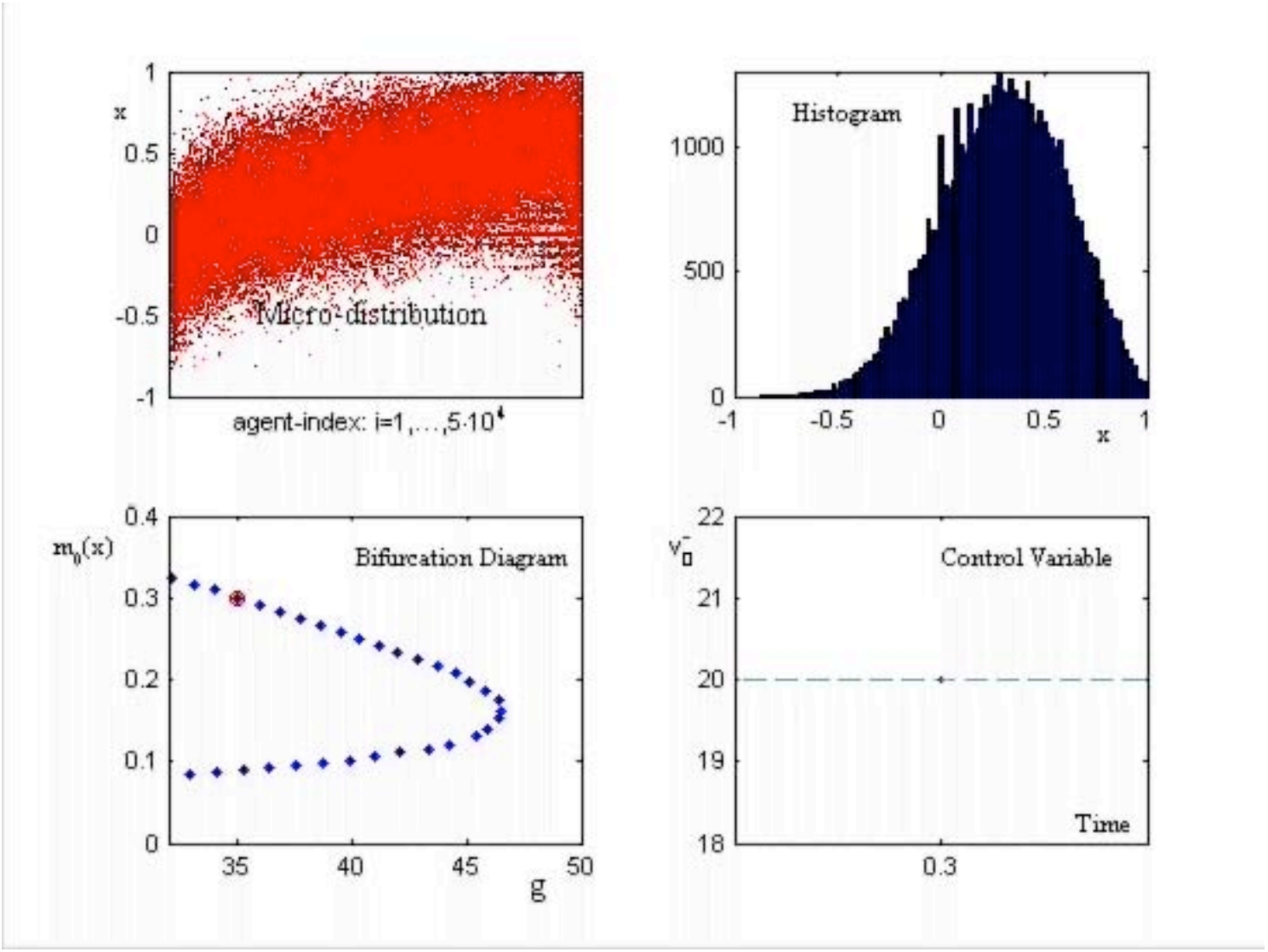
At steady state: $\dot{\boldsymbol{w}} = 0, \ \dot{\boldsymbol{x}} = 0 \rightarrow p = p^*$ and the system is stabilized in it's "unknown" steady state

$p = p^*, \ \boldsymbol{x} = \boldsymbol{x}^*$

In the case under study the control variable is the exogenous arrival frequency of "negative" information $\mathbf{v}_{ex}^-$ and the controlled variables the coefficients of the orthogonal polynomials used for the approximation of the ICDF

STABILIZING UNSTABLE MARKETS

Control variable: the exogenous arrival frequency of "negative" information v_{ex}^-

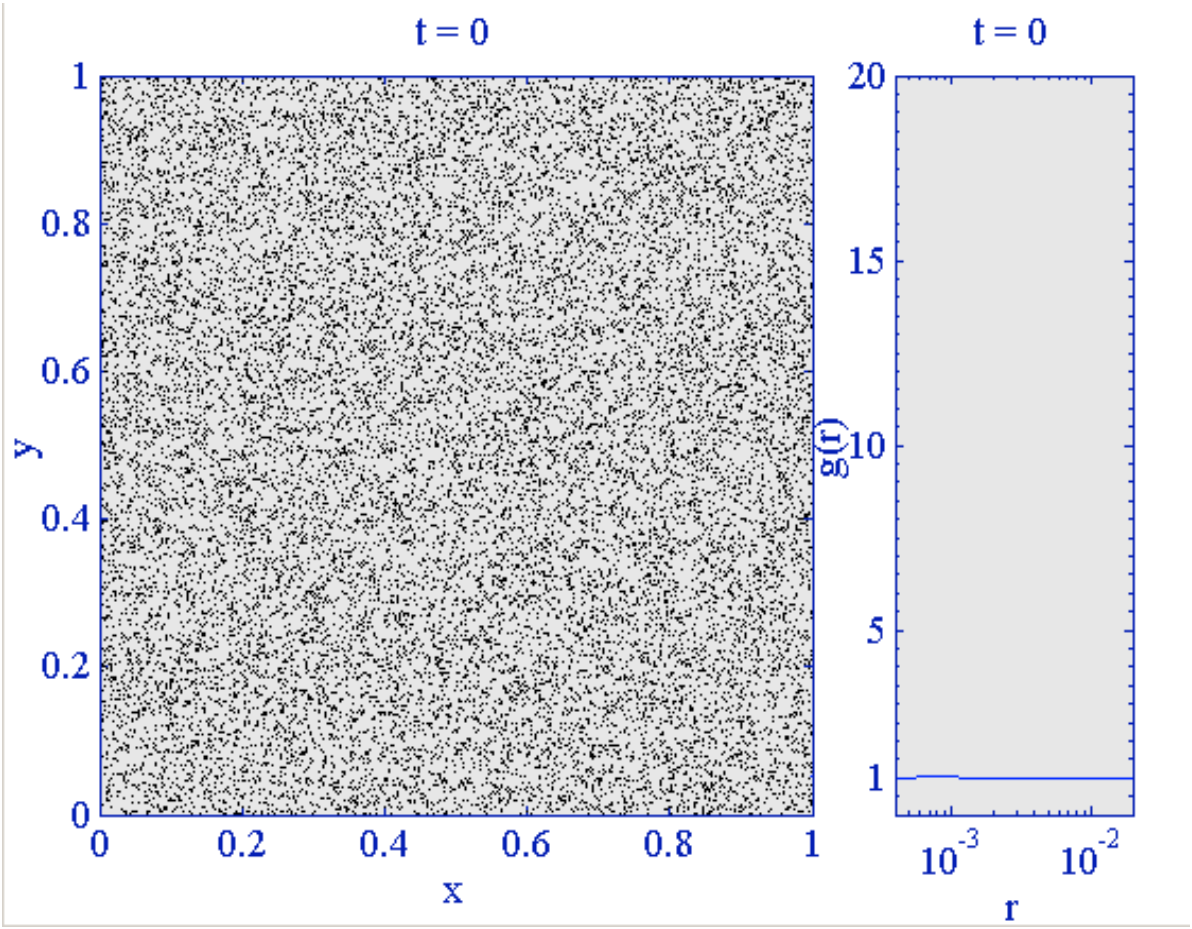


SIAM– July, 2004

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)

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

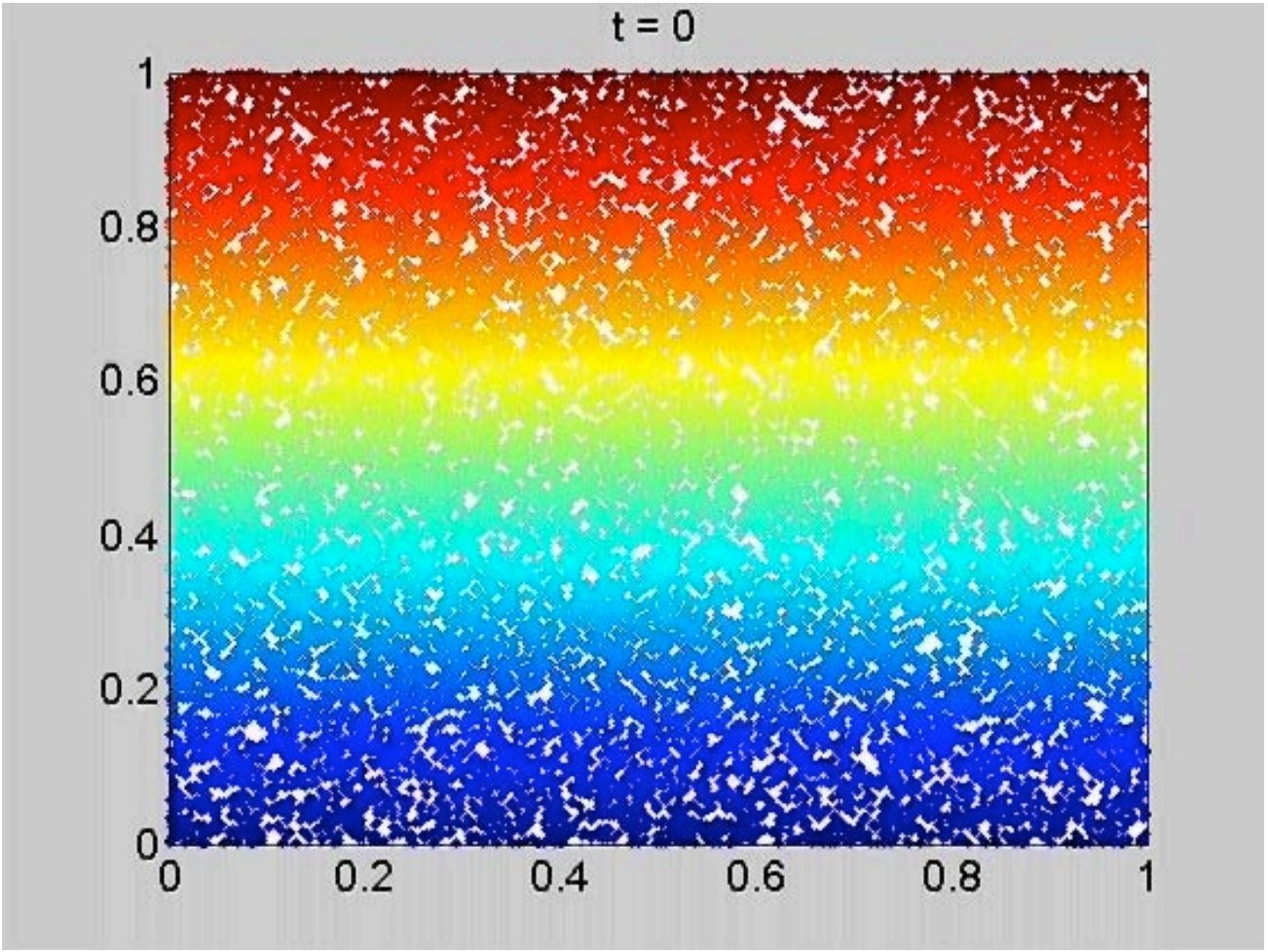
$$x_k(t+\tau) = x'_k(t) + U\tau/2 \cos[k y'_k(t) + \varphi(t)]$$

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

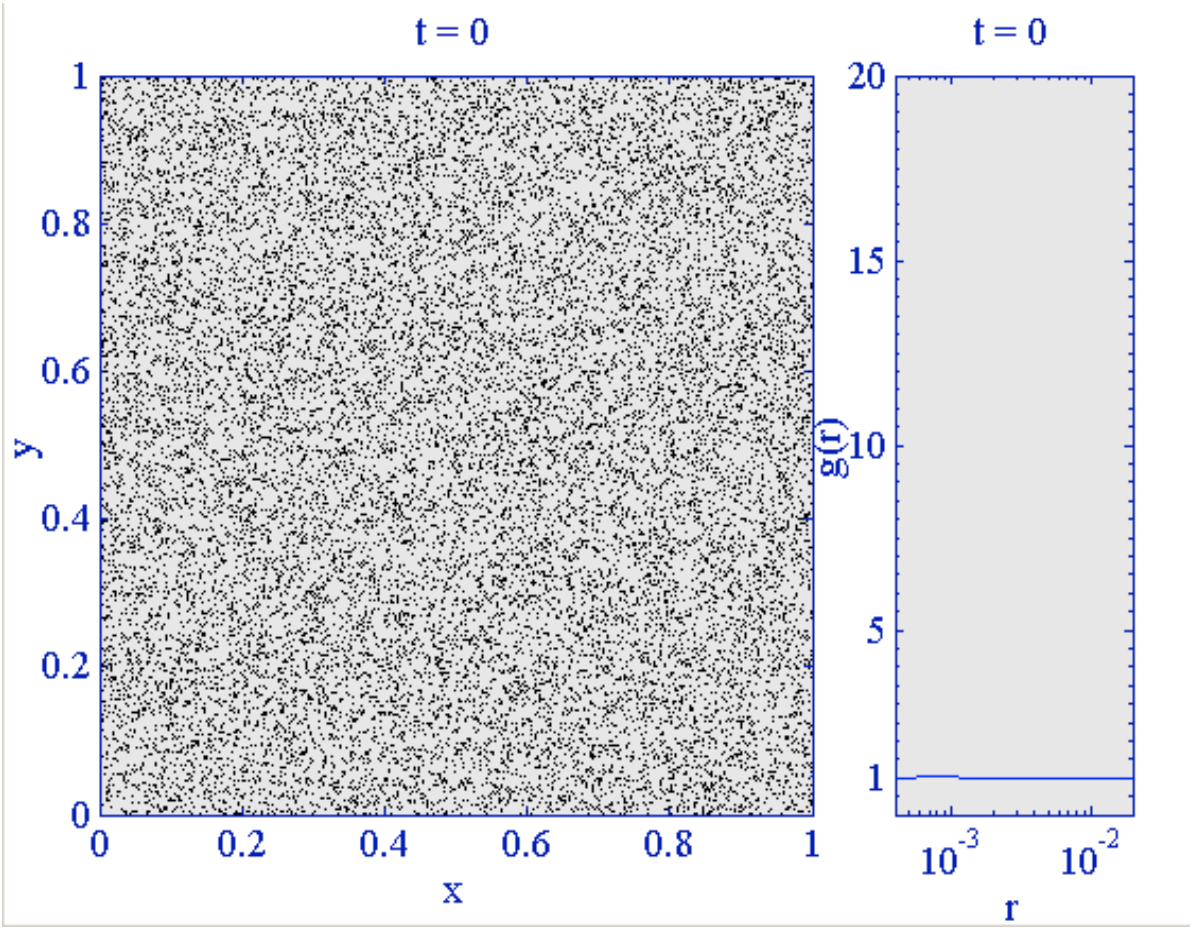
$$G_t = 2D\frac{1}{r}(rG_r)_r + 2(\lambda - \mu)G + \gamma\frac{1}{r}(r^3 G_r)_r + 2\lambda C\delta(\mathbf{r})$$

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

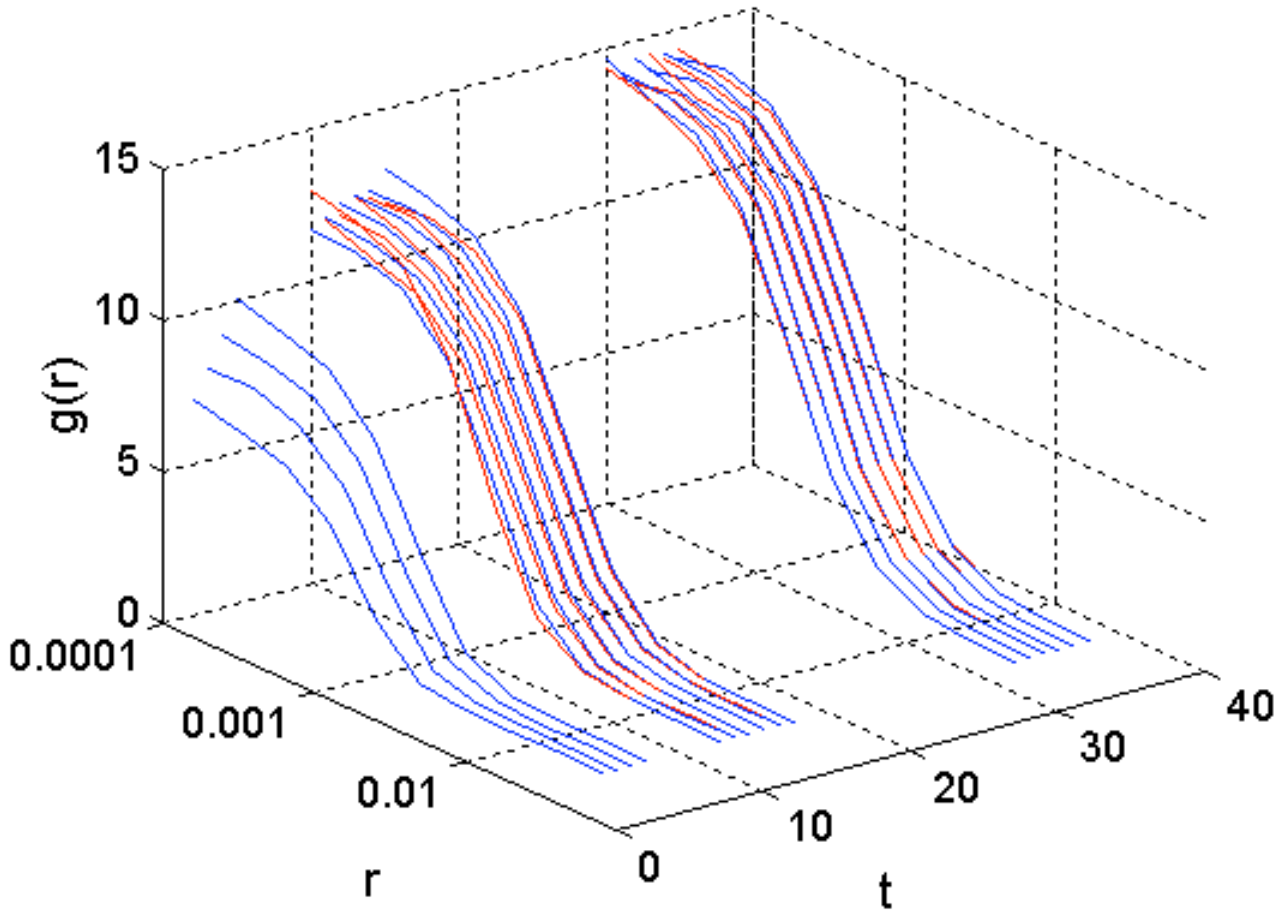
Stirring by a random field (color = y)



Dynamics of System with convection



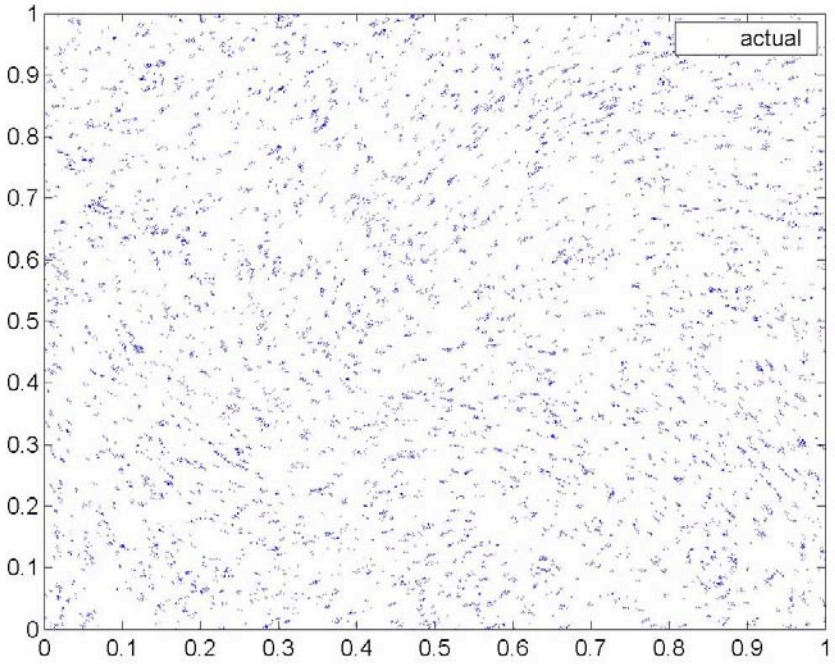
Projective Integration: From t=2,3,4,5 to 10

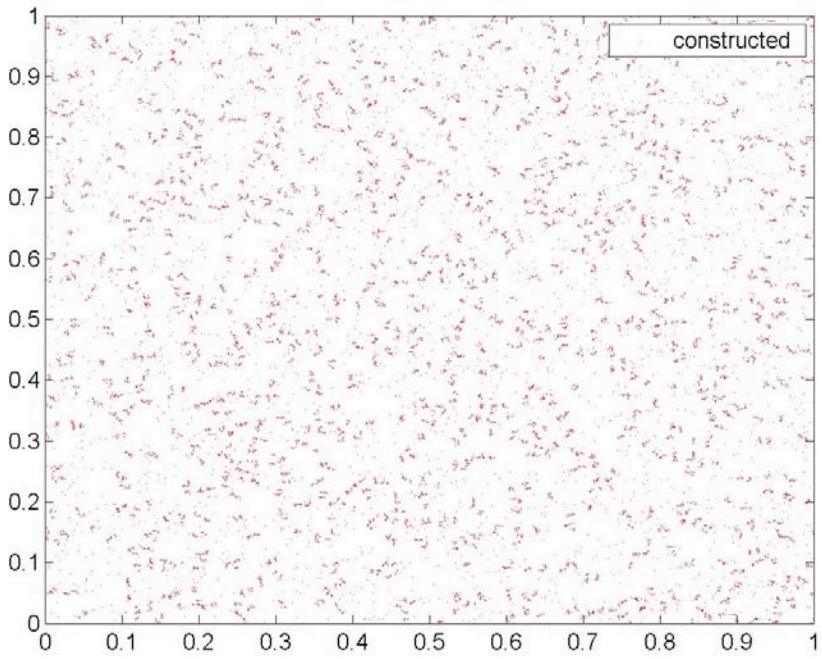


Mapping $g_0(r)$ – ***Lifting***

1. From any system, check to find interval $[r_m, r_{m+1})$, s.t. max$|g(r)/g_0(r)-1|$.
2. if $g(r) > g_0(r)$, remove the atom with the largest contribution to g(r), and place it randomly inside box.
3. if $g(r) < g_0(r)$, select atoms i, j, st $r_{ij} > r_m$ and contribute the least to g(r). Remove the less crowded atom between i and j, and place it at the distance of $[r_m, r_{m+1})$ from the other.

Comparison of actual system and *lifted* system based on





Fixed Point Calulation Using Newton-Raphson (2-parameters)

Iteration	Area	X Scale	Error
0	12.48	0.83	1.57
1	13.96	0.81	1.05
2	15.69	0.79	0.32
3	15.38	0.80	0.05

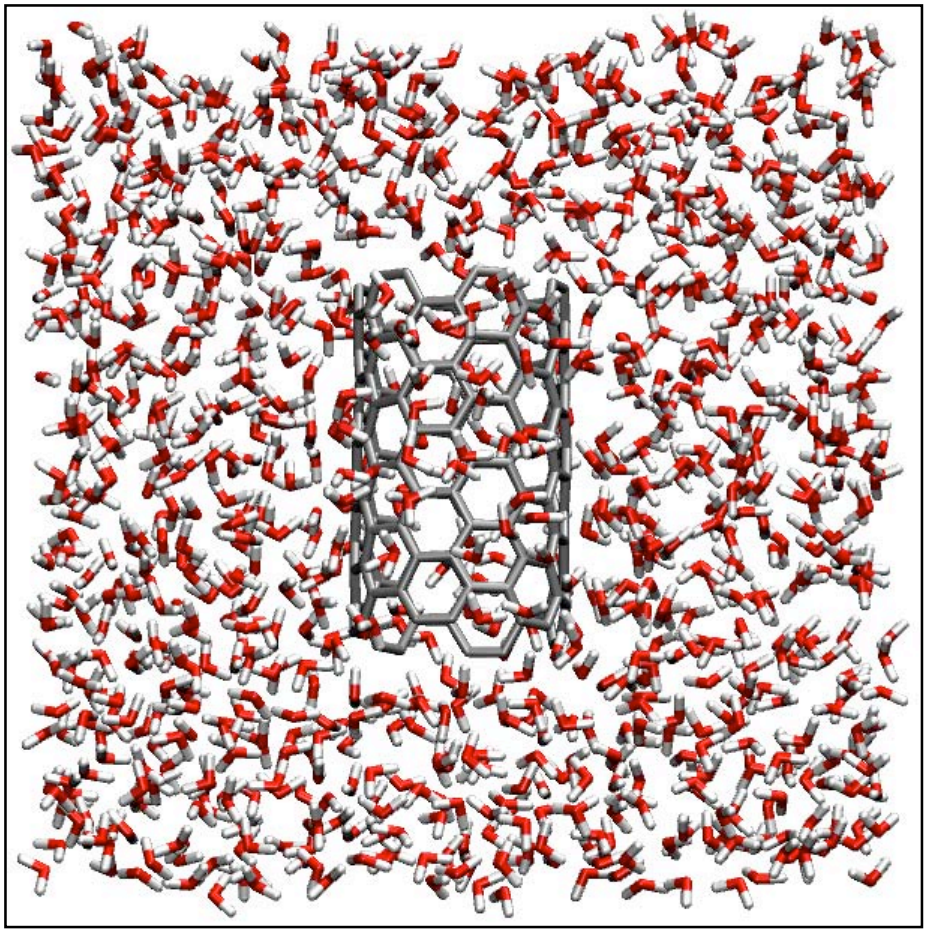
NLDbyMDofH2OinCNT

Water Confinement in Carbon Nanotube Molecular Channels

Model system for studying H_20 transport in channel pores of membrane proteins

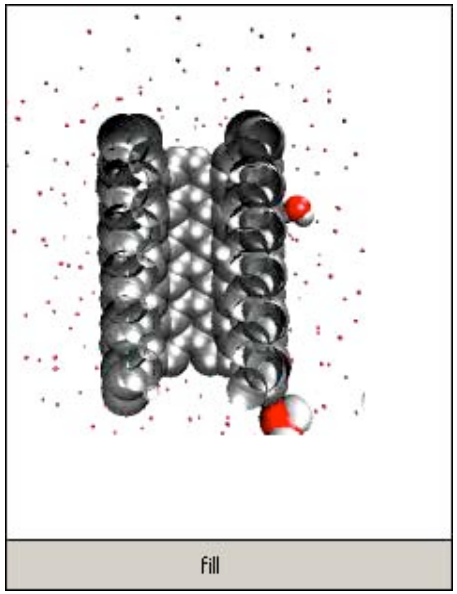
Classical Molecular Dynamics (MD)

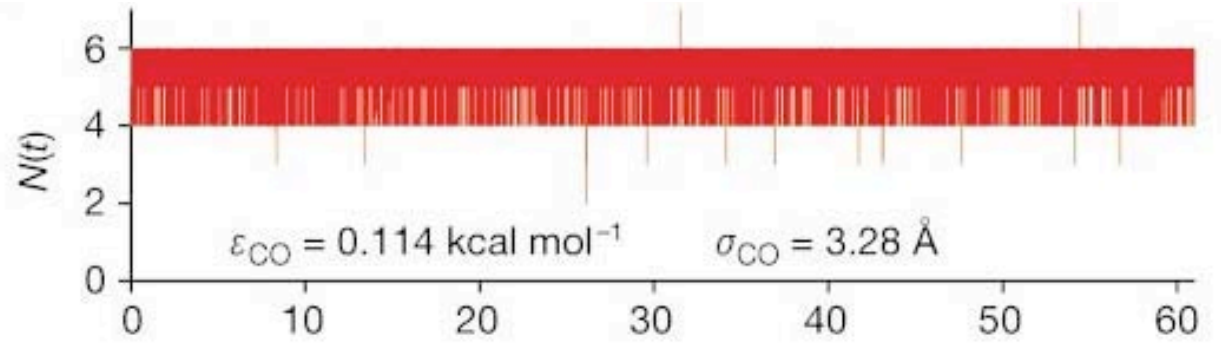
- Flexible CNT (L = 13.5 Å; R = 8.1 Å)
- Graphite parameters
- 1034 TIP3P water molecules
- AMBER 6.0
- Particle-mesh Ewald electrostatics
- Gerhard Hummer

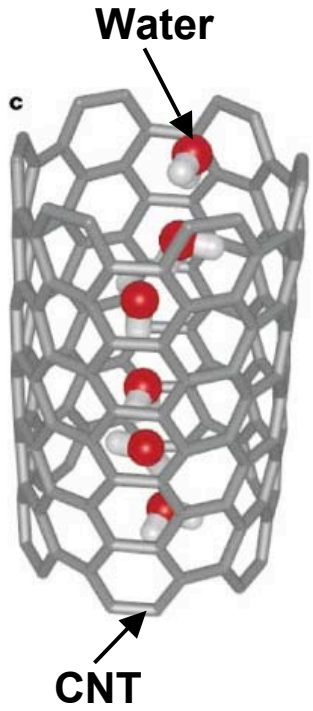


G. Hummer, J. C. Rasaiah, and J. P. Noworyta, Nature **414**, 188-190 (2001)

Filling-Emptying Transition of H_2O in CNT

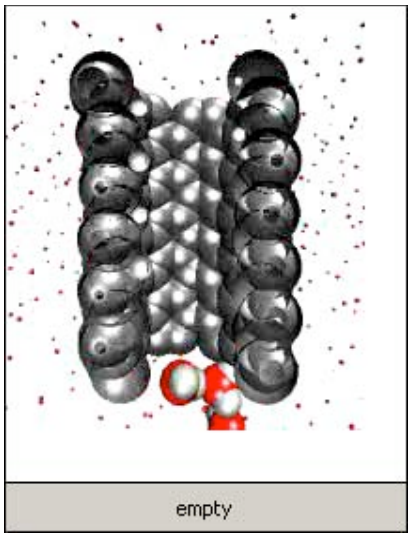


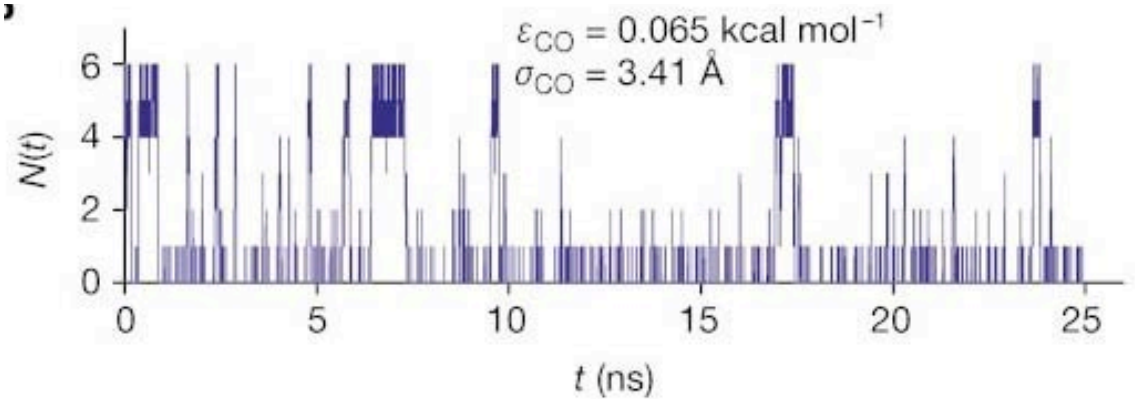




G. Hummer, J. C. Rasaiah, and J. P. Noworyta, Nature **414**, 188-190 (2001)

Filling-Emptying Transition of H_2O in CNT



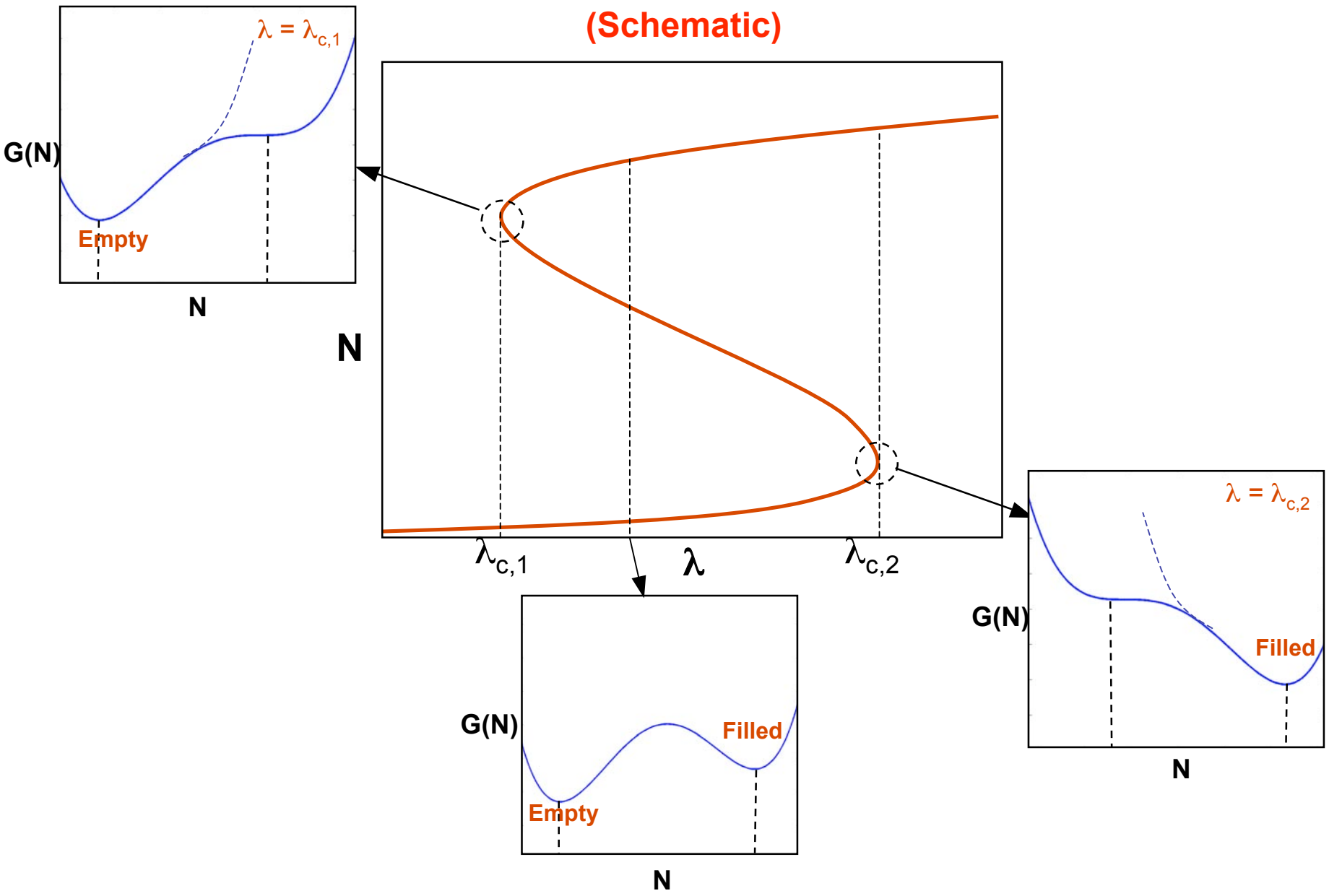


"Equation Free" Approach

Analyse effect of CNT-Water Interactions on the H_20 Occupancy

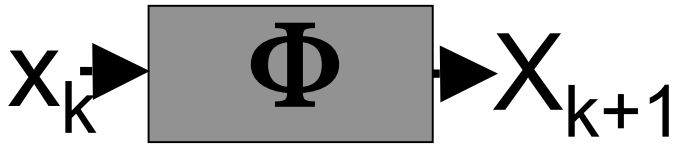
G. Hummer, J. C. Rasaiah, and J. P. Noworyta, Nature **414**, 188-190 (2001)

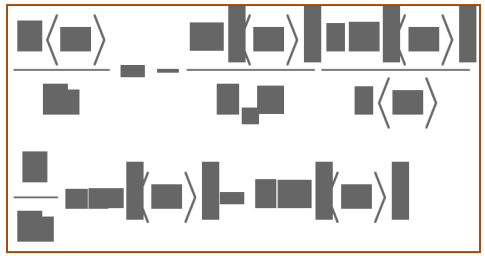
Effect of CNT-Water Attraction (λ) on Pore Hydration

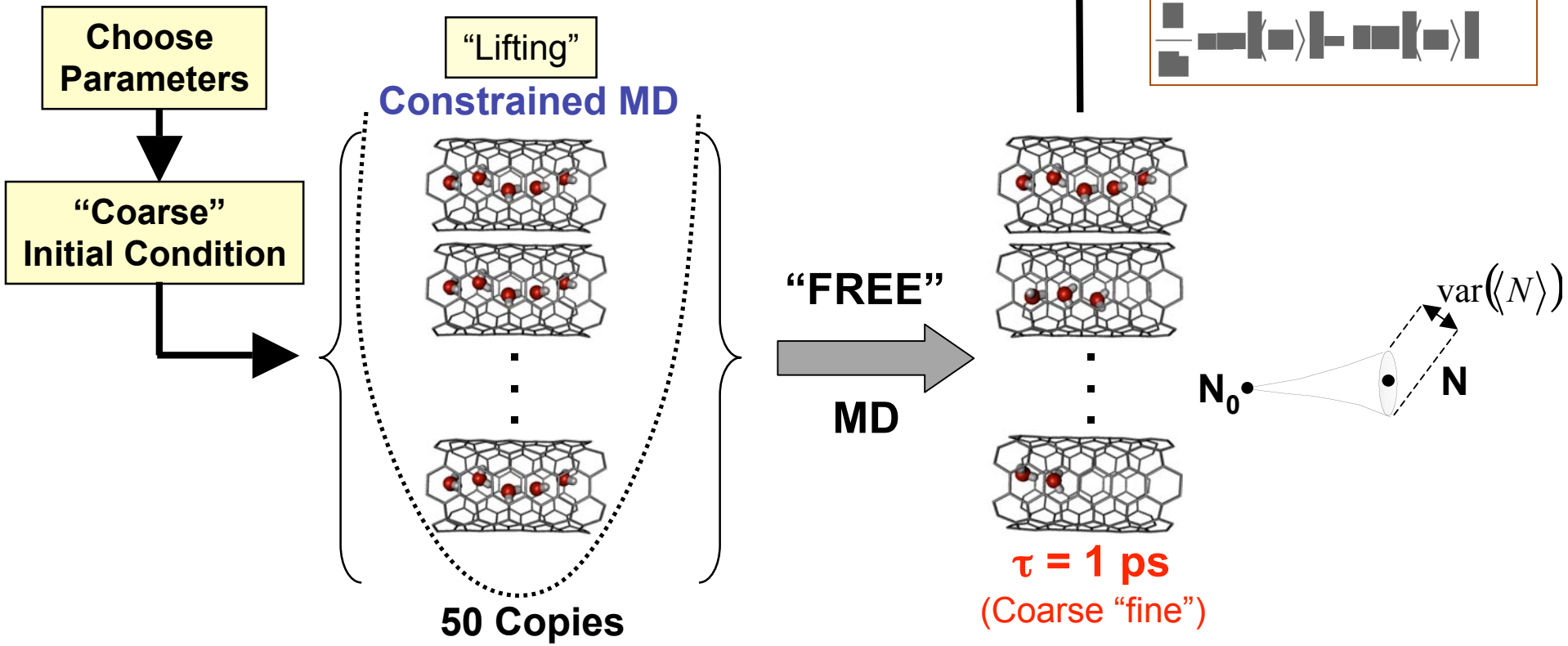


The Heart of the Method – The "Timestepper"

$$\Phi(N,\lambda) = \frac{1}{M}\sum_{i=1}^{M} N_i(\tau; N_0, t=0; \lambda) = \overline{N(\tau; N_0, t=0; \lambda)}$$

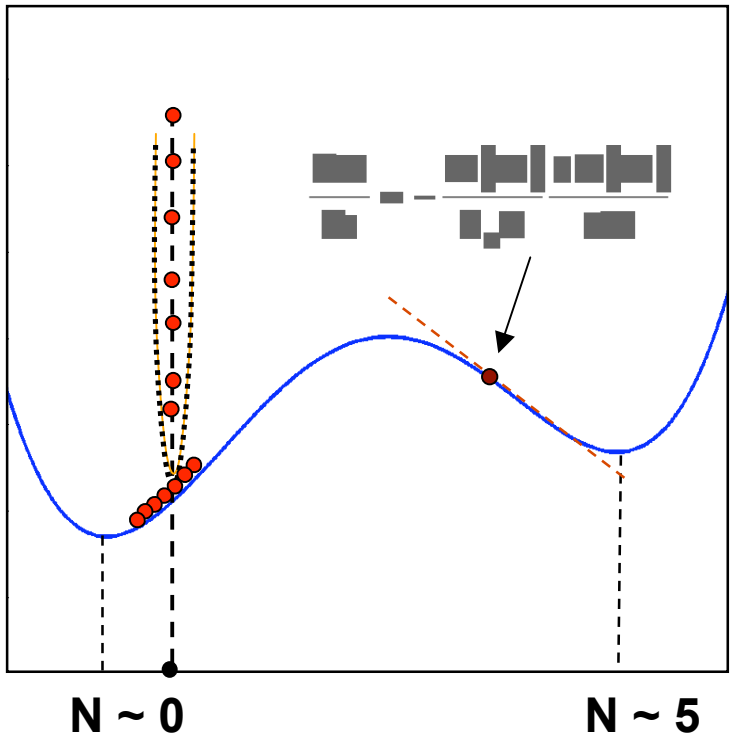






- Constrain for τ = 15 ps and sample configurations: τ > 10 ps
- Initialize replicas: velocities from Maxwell-Boltzmann distribution

Solving for the Branches



$$\frac{d\langle N\rangle}{dt} \approx -\frac{D(\langle N\rangle)}{k_B T}\frac{\partial G(\langle N\rangle)}{\partial\langle N\rangle} = F(N,\lambda)$$

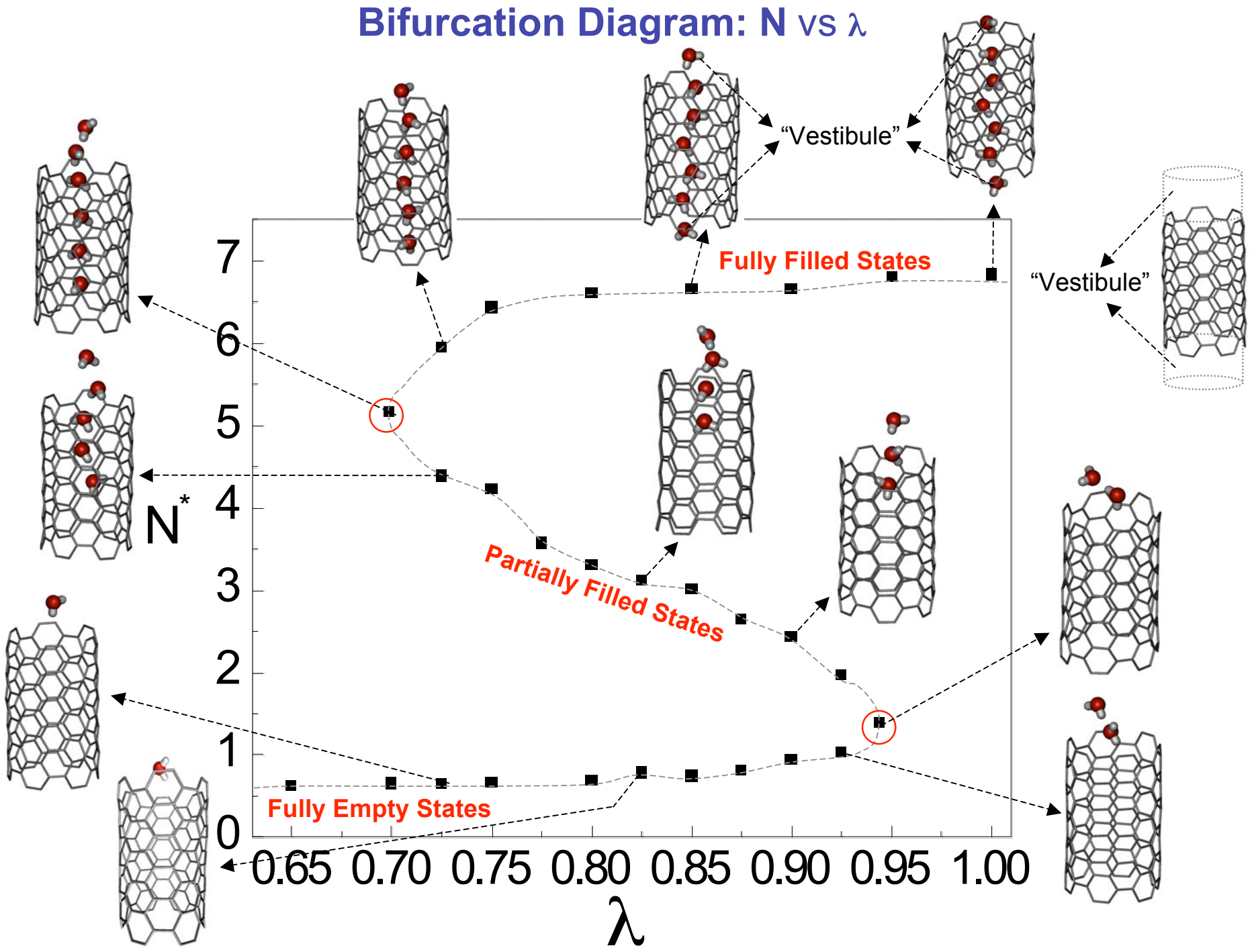
$$\frac{\partial G}{\partial N} = 0 \qquad \frac{dN}{dt} = 0 \leftrightarrow N - \Phi(N,\lambda) = 0$$

1. Set parameter (λ); specify coarse initial condition (N)
2. "Lifting": constrained MD
3. Let 50 replicas "free" for τ = 1 ps
4. Estimate Φ(N,λ) and J(N,λ) (Jacobian) (central differences for derivatives)
5. Do NR iterations; estimate new N
6. Go to (2) and repeat

Solve by Newton-Raphson

$$\frac{dN}{dt} = 0 \leftrightarrow N - \Phi(N,\lambda) = 0$$

$$\underline{\underline{J}} = \left[1 - \frac{\partial\Phi(N,\lambda)}{\partial N}\right] \qquad \left.\frac{\partial\Phi}{\partial N}\right|_{N_k} = \frac{\Phi(N_k + \Delta N_k,\lambda) - \Phi(N_k - \Delta N_k,\lambda)}{2\Delta N_k}$$


Bifurcation Diagram: N vs λ
"Vestibule"
"Vestibule"
Fully Filled States
Partially Filled States
Fully Empty States
N*
λ
0
1
2
3
4
5
6
7
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00

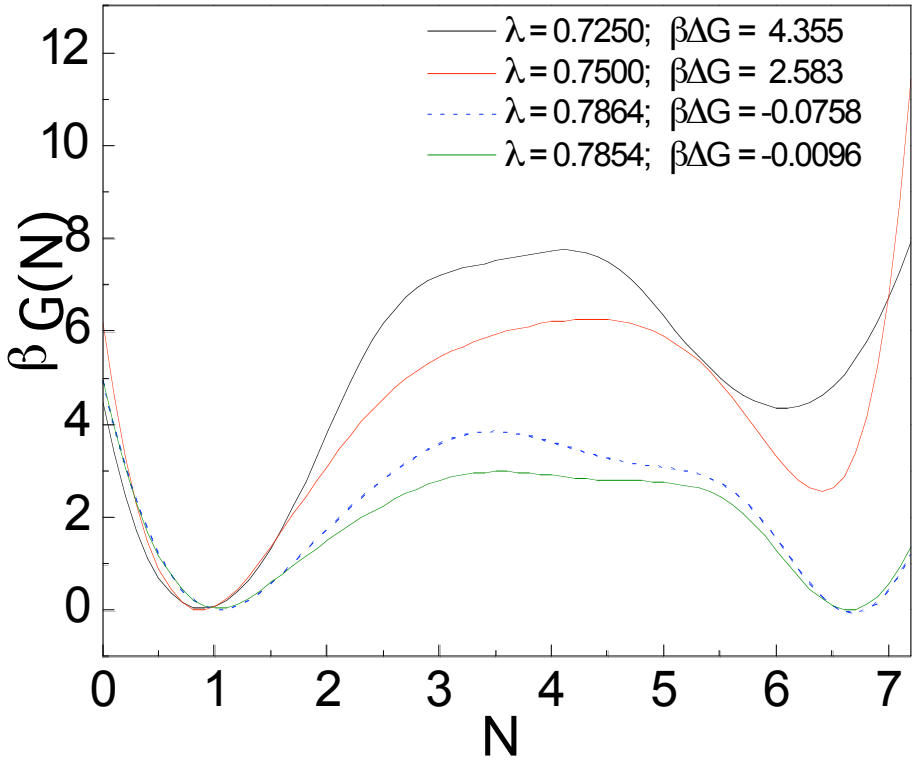
Effective Free Energy Profiles: First Order Phase Transition

λ	ΔG
0.725	4.354
0.750	2.583
0.7864	-0.0758
0.7854	-0.0096

Solve $\Delta G(\lambda) = 0$

NR: $\lambda_1 = \lambda_0 - \Delta G(\lambda=\lambda_0) / \Delta G'(\lambda)|_{\lambda 0}$

$\Delta G'(\lambda)|_{\lambda 0} = (\Delta G(\lambda_0+d\lambda)-\Delta G(\lambda_0)) / d\lambda$



- With λ = 0.7864 ⇒ CMD for G(N)
- Fit quadratic for first 3 data points and estimate λ = 0.7854
- With λ = 0.7854 ⇒ CMD for G(N)
- Fit quadratic again – no change in fit; estimated λ = 0.7853

Kinetics of Filling-Emptying Transitions

Average transition time for system to go from completely filled state to empty state

$$\tau \approx \frac{2\pi}{\overline{D}\sqrt{\beta\Phi''(N_{\min})\left|\beta\Phi''(N_{saddle})\right|}} e^{\beta\Delta G_{\min\leftrightarrow saddle}}$$

$$\overline{D} = \frac{1}{2}\left(D(N_{\min}) + D(N_{saddle})\right)$$

λ	$\tau_{FullI\rightarrow Empty}$ (ps)	$\tau_{Empty\rightarrow Full}$ (ps)	Method
0.7854	~ 125	~ 126	CMD
0.750	~ 183	~ 2680	CMD
0.7517	230 ± 70	2400 ± 700	Eqblm. MD

Eqblm. MD: 15 emptying transitions (Full → Empty); 14 filling transitions (Empty → Full)

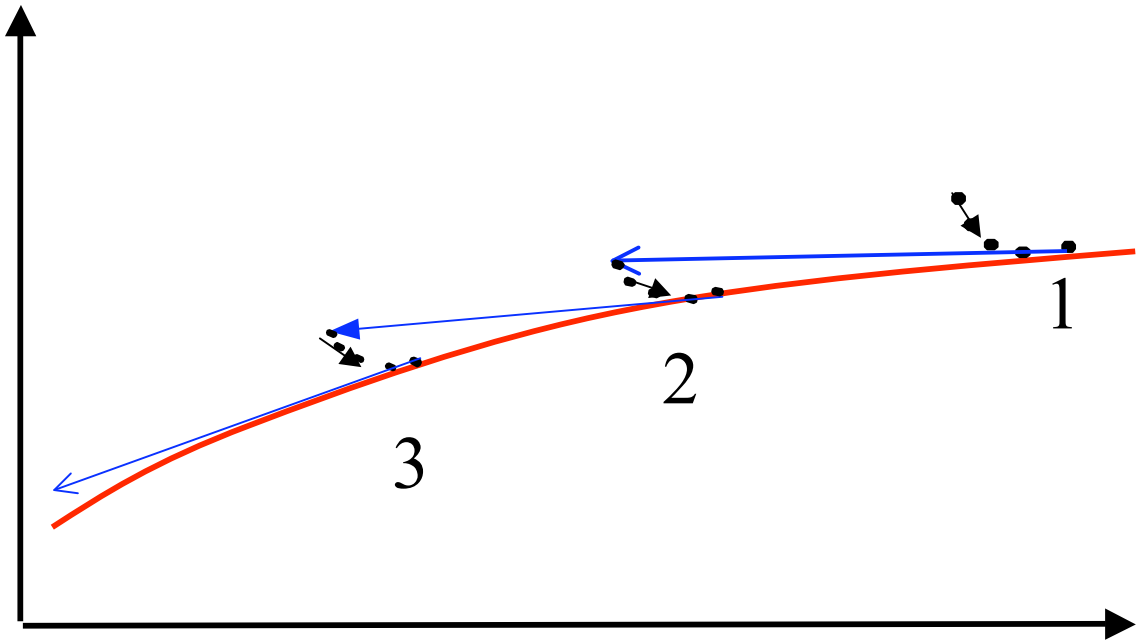
Consider states inside CNT $00000 \underset{k_2}{\overset{k_1}{\Leftrightarrow}} 11111$ $\quad k_1 = \frac{1}{\tau_{empty\rightarrow full}}; k_2 = \frac{1}{\tau_{full\rightarrow empty}}$

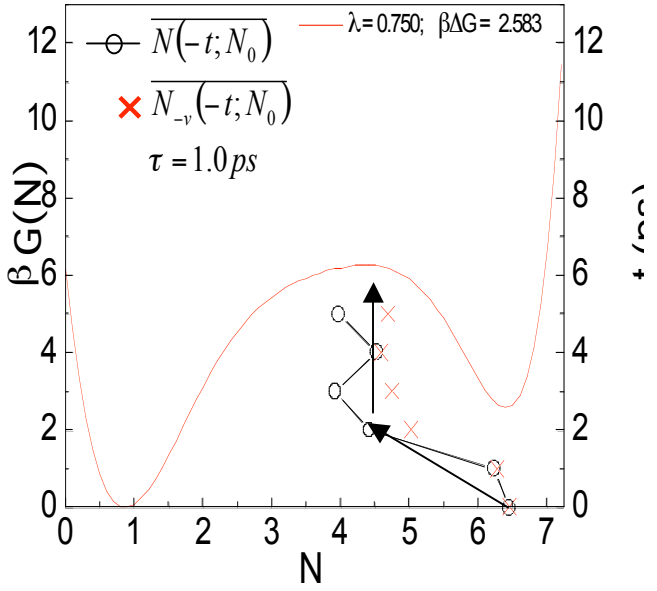
Relaxation time between filled and empty states $= \frac{1}{k_1 + k_2}$ **≈ 171 ps**

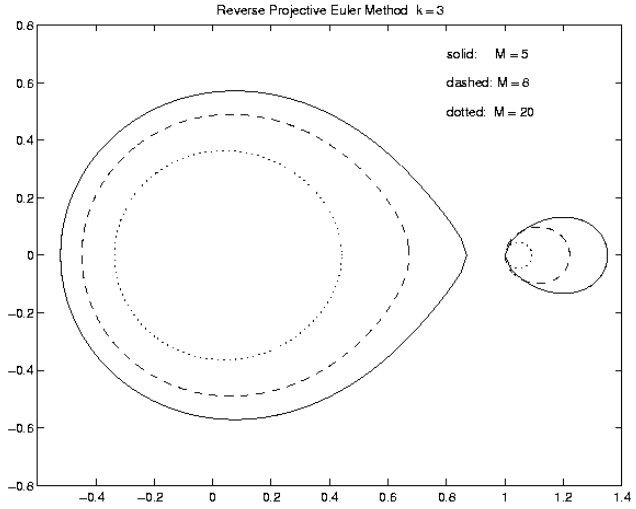
Relaxation time from coarse propagators: **165 ps**

Reverse Projective Integration –

a sequence of outer integration steps backward;
based on forward steps + estimation



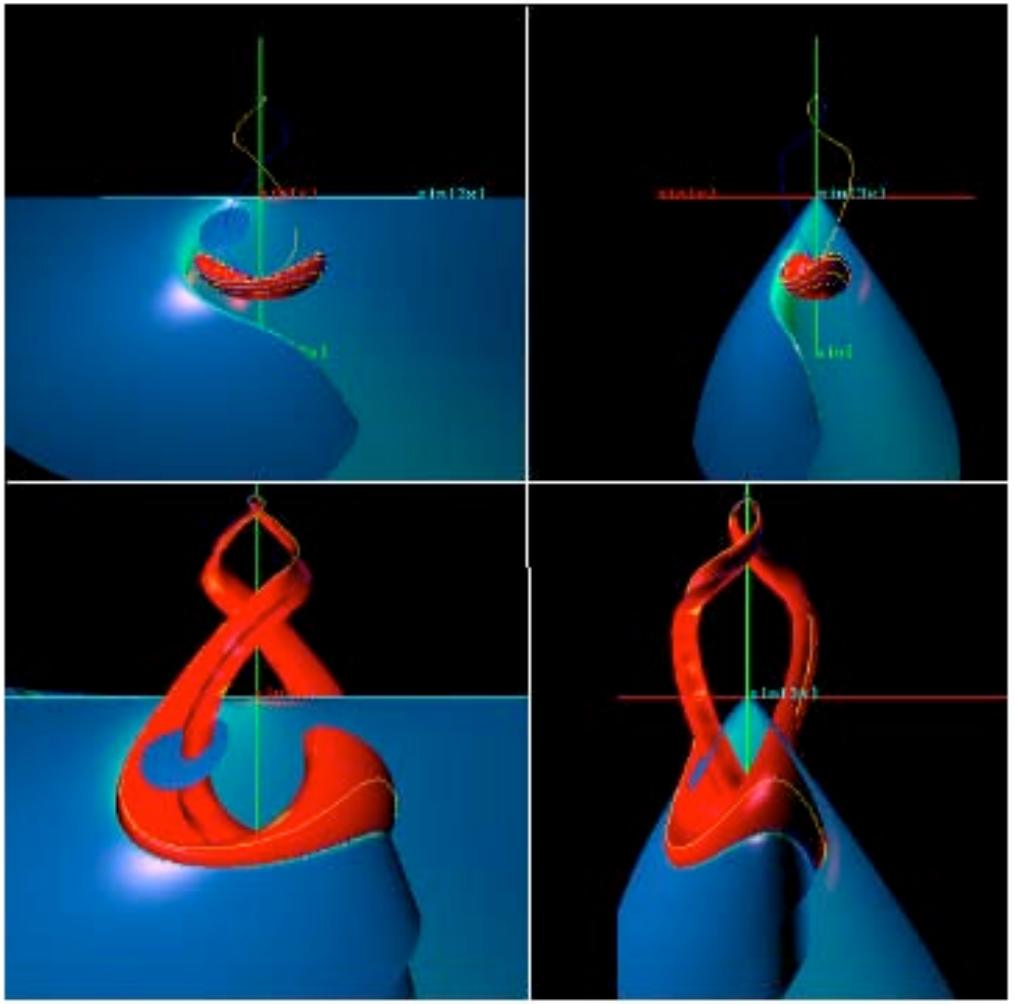




Reverse Integration: **a little forward,**
and then **a lot backward !**

We are studying the accuracy and stability of these methods

And so all our algorithms for computing stable manifolds can be used (in conjunction with coarse timesteppers and RI to approximate low-dimensional free energy surfaces !

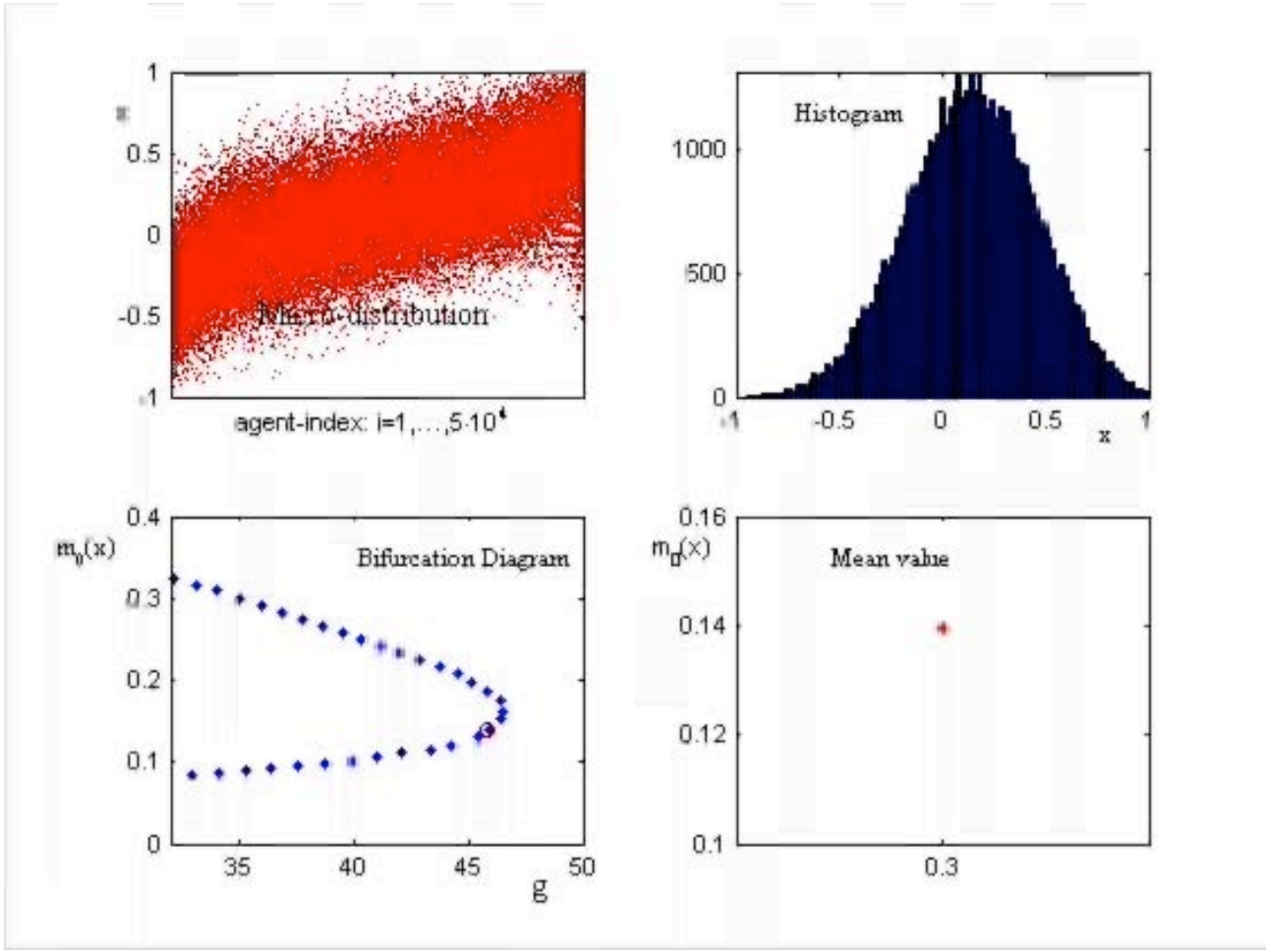


ALTERNATIVE Mathematics – inspired ENSEMBLES

The Oseberg Transition
Johnson, Jolly & K.
IJBC 2001

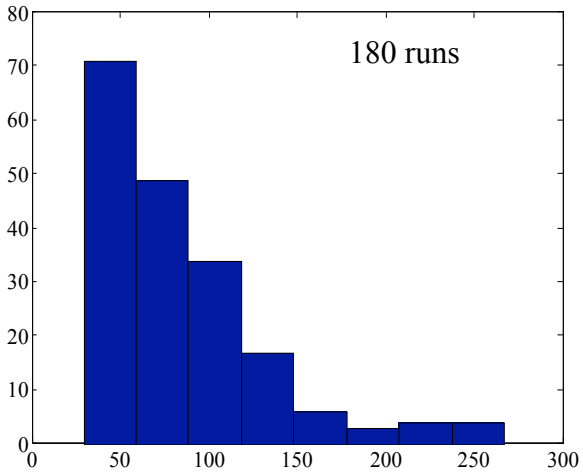
Rare Events: Escaping from stability

Open loop response, g = 45.8



Histogram of escape times

(x(t=0)=0)



Computer-Aided Analysis of Nonlinear Problems in Transport Phenomena

Robert A. Brown, L. E. Scriven and William J. Silliman

in HOLMES, P.J., New Approaches to Nonlinear Problems in Dynamics, 1980

ABSTRACT The nonlinear partial differential equations of mass, momentum, energy, Species and charge transport…. can be solved in terms of functions of limited differentiability, no more than the physics warrants, rather than the analytic functions of classical analysis… ….. basis sets consisting of low-order polynomials. …. systematically generating and analyzing solutions by fast computers employing modern matrix techniques.

….. nonlinear algebraic equations by the Newton-Raphson method. … The Newton-Raphson technique is greatly preferred because the Jacobian of the solution is a treasure trove, not only for continuation, but also for analysing stability of solutions, for detecting bifurcations of solution families, and for computing asymptotic estimates of the effects, on any solution, of small changes in parameters, boundary conditions, and boundary shape……

In what we do, not only the analysis, but *the equations themselves* are obtained on the computer, from short experiments with an alternative, microscopic description.

Coming full circle

No equations ?

Isn't that a little medieval ? Equations = "Understanding", right ?

AGAIN matrix free iterative linear algebra

$$A\,x = b$$

PRECONDITIONING, $B\,A\,x = B\,b$

B approximate inverse of A

Use "the best equation you have"

to *precondition* equation-free computations.

With enough initialization authority:

equation free *laboratory experiments*