Multiscale algorithms for capturing the macroscale behavior of a system with the help of microscopic models

Weinan E

Princeton University

Supported by ONR

Focus on the following class of multiscale problems:

1. we are interested only in the *macro behavior* of the system;
2. a reliable *microscopic* model is available, but too expensive to be used directly.

Traditional approach: *Sequential* multiscale modeling

- Write down an empirical macroscale model
- Compute coefficients or parameters using microscale model.

Example:

1. hopping rates in kinetic Monte Carlo
2. equation of state in gas dynamics ($p = P(\rho, T)$)
3. interatomic potentials in molecular dynamics ($V = V(\mathbf{x}_1, \cdots, \mathbf{x}_N)$)

OK, but limited to the situation when the unknown components of the macroscale model depend on few parameters.

Modern approach: *Concurrent* ("on-the-fly") coupling

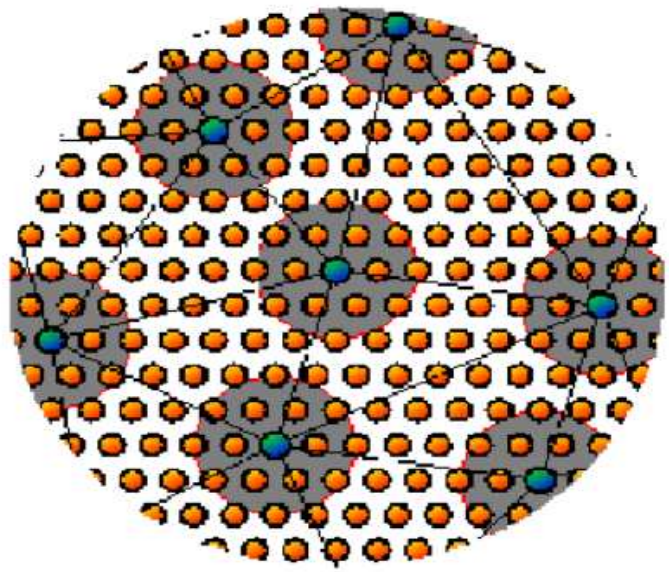
The Car-Parrinello molecular dynamics (1985)

- Macro behavior of interest: molecular dynamics of the atoms (nuclei)
- Unknown: Force field (don't want to use empirical force field such as Lennard-Jones)
- Micro model: Electronic structure models (e.g. density functional theory)
- Key: "On-the-fly" (concurrent) coupling.

The local quasicontinuum method (Tadmor et al. 1996)

- Macro behavior of interest: Elastic deformation of crystals
- Unknown: Elastic energy functional
- Micro model: Atomistic (potential energy in terms of positions of atoms)

1. Select representative atoms and form finite element mesh
2. Energy of trial function: Average the potential in small clusters



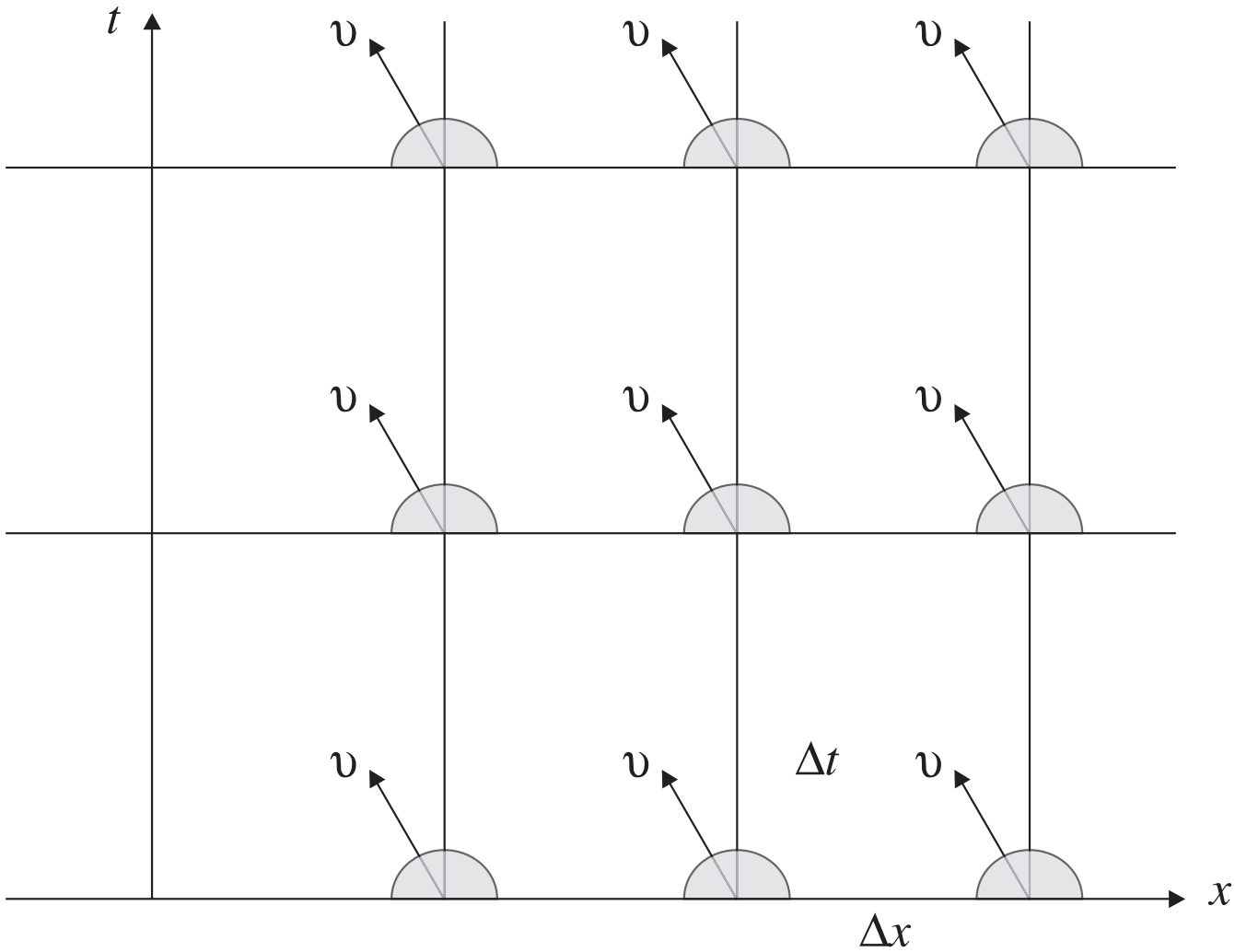
Kinetic schemes for gas dynamics (Deshpande, 80's)

- Macro behavior of interest: gas dynamics
- Unknown: constitutive laws
- Micro model: kinetic theory

Macro variables $U = (\rho, m, E)$ – density of conserved quantities.

1. Reconstruction: From U^n, find consistent initial condition for the kinetic equation.
2. Solve kinetic equation in neighborhood of cell boundaries.
3. Perform the appropriate averages to find the corresponding fluxes, and use them to find U^{n+1}:

$$U_j^{n+1} = U_j^n - \frac{\Delta t}{\Delta x}(F_{j+1/2}^n - F_{j-1/2}^n)$$


t
υ
υ
υ
υ
υ
υ
υ
υ
υ
Δt
x
Δx

Superparametrization

Grabowski and Smolarkiewicz (1999), *cloud-resolving convective parametrization*

- Starting with the usual "Reynolds-averaged" type of equations.
- The "Reynolds stress" terms computed by embedding a 2D cloud scale model in each column of the large scale model.
- "Embarassingly parallel" turbulence models.

Summary

Common features:

1. Capture the macroscale behavior with the help of microscopic model/simulation.
2. Make use of scale separation.
   - Solve the microscopic problem in small domains (local QC, kinetic scheme, superparametrization).

Looking for general strategies

A. Brandt (2000): *Multiscale scientific computation: Review 2001*

Extension of multi-grid method:

- For the purpose of capturing only the large scale behavior (not resolving all the small scale details), without the need of an explicit macroscale model.

  "At sufficient coarse level, this entire algorithm effectively produces *macroscopic 'equations'* for the simulated system .... This can yield a macroscopic numerical description for the fluid even for those cases where the traditional derivation of closed-form differential equations is inapplicable."

- Micro model (KMC, MD, etc) used at the finest level of grids.

- Linking macro and micro states through “interpolation” and “projection”.
  1. Interpolation: macro to micro
  2. Projection: micro to macro
- Microscopic model is simulated on subdomains, for a small number of iterations.

  “few sweeps are enough, due to the fast CMC equilibration. This fast equilibration also implies that the interpolation can be done just over a restricted *subdomain*, serving as *window*: In the window interior fine-level equilibration is reached.”

Summary of key suggestions:

1. Capturing macro behavior using micro models, with no explicit macro models
2. Going back and forth between macro and micro states
3. Solve micro problems on "windows" and with "few sweeps"

How do we realize these ideas?
Not clear to me.

Alternative strategies based on the same general philosophy:

- Heterogeneous multiscale method (HMM)
- "Equation-free"

'Equation-free (EF)' (Kevrekidis, Gear, ...)

"Enabling microscopic simulators to perform system-level tasks".

A collection of techniques that explicitly take into account scale separation:

1. Coarse bifurcation (a la the RPM of Shroff and Keller)
2. Projective integrators (for time)
3. Gap-tooth schemes (for space)
4. Patch dynamics (for time and space)

Basic building block: Time-stepper

Extended multi-grid	*"Equation-free"*
Interpolation	Lifting
Equilibration	Evolutions (equilibration)
Restriction (projection)	Restriction
	Extrapolation

Example: Projective Integrators

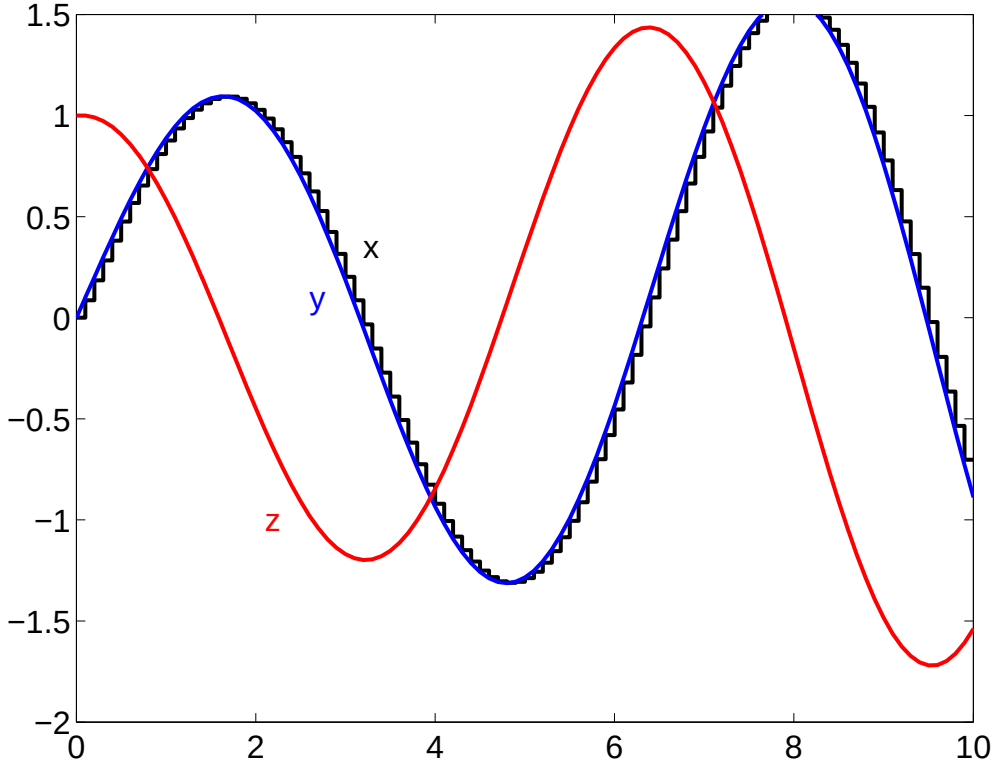
$$\dot{x} = f^{\varepsilon}(x), \quad U = \text{macro variable}$$

- Lift: From U^n, create initial condition $x^{n,0}$.
- Evolve:
$$x^{n,m+1} = x^{n,m} + \delta t f^{\varepsilon}(x^{n,m})$$
$m = 0, 1, 2, \cdots, M-2$
- Restrict: Compute $\tilde{U}_1$ from $\{x^{n,M-1}\}$ and $\tilde{U}_2$ from $\{x^{n,M}\}$
- Extrapolate:
$$\frac{U^{n+1} - \tilde{U}_2}{\Delta t} = \frac{\tilde{U}^2 - \tilde{U}^1}{\delta t}$$

Higher order schemes are obtained using higher order polynomial extrapolation.

Examples (Gil Ariel, Austin):

$$x' = -\frac{1}{\varepsilon}(x - y), \quad x(0) = 0$$
$$y' = z, \quad y(0) = 0$$
$$z' = -x, \quad z(0) = 1$$



HMM (Heterogeneous multiscale method)

E, Engquist, Vanden-Eijnden ..... (2003)

1. *Macroscale solver*: *Assume a form of macro model*, and then choose a stable numerical scheme for the model.
2. *Estimating the missing data*: Some data needed in the macro-solver are missing due to the incomplete knowledge of the macro model. These data are estimated from the microscopic model.

Macro-solver – micro-solver – data estimator

$$-\nabla \cdot (a^{\varepsilon}(x)\nabla u) = f(x)$$

- Macro model: $-\nabla \cdot (A(x)\nabla U) = f(x)$.
  Macro solver: standard finite element with macro mesh size H, e.g. piecewise linear.
- Data to be estimated: stiffness matrix

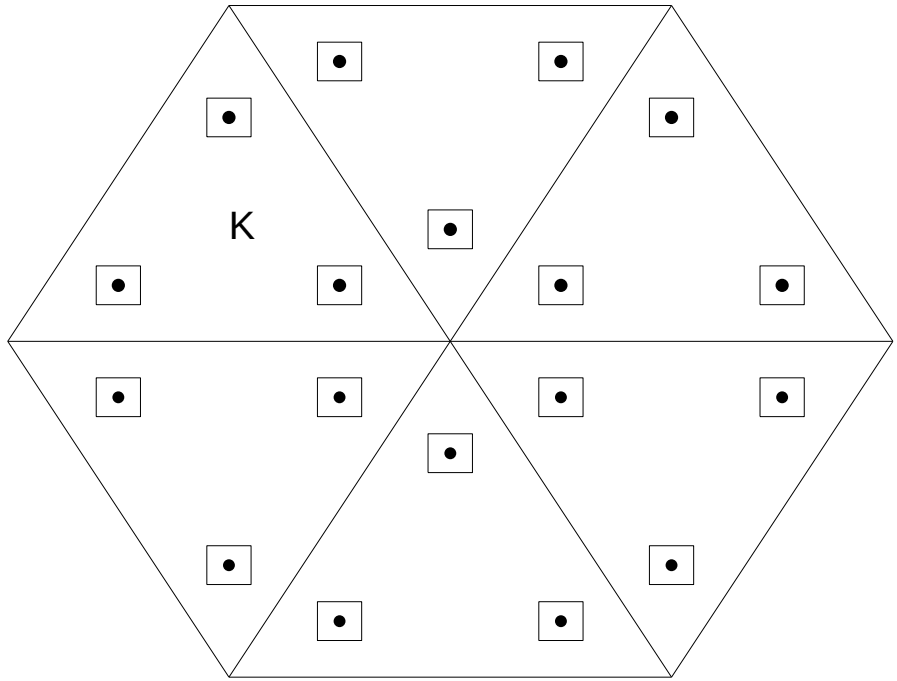
$$\mathcal{A} = (A_{ij}), \quad A_{ij} = \int_D \underbrace{\nabla \Phi_i(x)^T A(x) \nabla \Phi_j(x)}_{f_{ij}(x)} dx$$

Compute A_{ij} by numerical quadrature

$$A_{ij} = \int_D f_{ij}(x)dx = \sum_K |K| \sum_{x_\ell \in K} w_\ell f_{ij}(x_\ell)$$

x_ℓ = quadrature points, w_ℓ = quadrature weights

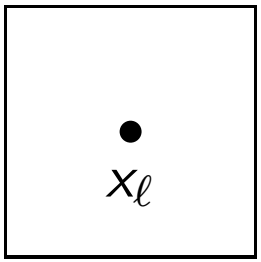
Illustration of HMM



$$\varepsilon \ll \delta \ll H$$

$H =$ size of triangle, $\quad \delta =$ size of box

Approximating $f_{ij}(x_\ell)$ by solving locally constrained microscopic problem at x_ℓ



Constraints: $\frac{1}{\delta^d}\int_{I_\delta(x_\ell)} \nabla\varphi_i^\varepsilon dx = (\nabla\Phi_i)(x_\ell)$

$$\begin{cases} -\nabla\cdot(a^\varepsilon(x)\nabla\varphi_i^\varepsilon(x)) = 0, & I_\delta(x_\ell) \\ a^\varepsilon(x)\dfrac{\partial\varphi_i^\varepsilon}{\partial n} = \lambda^T\cdot n & \partial I_\delta(x_\ell) \end{cases}$$

$\lambda =$ Lagrange multiplier for the constraints

Other boundary conditions:
Periodic: $\varphi_i^\varepsilon(x) - \Phi_i(x)$ is periodic with period $I_\delta(x_\ell)$
Dirichlet: $\varphi_i^\varepsilon(x) - \Phi_i(x) = 0$ on $\partial I_\delta(x_\ell)$

Error Estimates (E, Ming, Zhang, JAMS, 2005)

General abstract error estimate: Under uniform ellipticity condition and $U_0 \in W^{k+1,\infty}(\Omega)$:

$$\|U_0 - U_{\mathrm{HMM}}\|_{H^1(D)} \leq C\left(H^k + e(\mathrm{HMM})\right)$$

$$\|U_0 - U_{\mathrm{HMM}}\|_{L^2(D)} \leq C\left(H^{k+1} + e(\mathrm{HMM})\right)$$

$$\|U_0 - U_{\mathrm{HMM}}\|_{W^{1,\infty}(D)} \leq C\left(H^k + e(\mathrm{HMM})\right)|\ln H|$$

U_0: Solution for a macroscopic model (e.g. homogenized problem)

$$e(\mathrm{HMM}) = \max_{K\in\mathcal{T}_H}\max_{x_\ell\in K}|A_H(x_\ell) - A(x_\ell)|$$

$$= \text{error in data estimation}$$

A : effective coefficients in the equation for U_0

Case study: homogenization problem

- For periodic coefficients:

$$e(\mathrm{HMM}) \leq \begin{cases} C_\varepsilon & l_\delta = l_\varepsilon \\ C\left(\frac{\varepsilon}{\delta} + \delta\right) & \text{otherwise} \end{cases}$$

- For random coefficients: $\kappa \approx \frac{6}{25}$

$$\mathbb{E}e(\mathrm{HMM}) \leq \begin{cases} C\left(\frac{\varepsilon}{\delta}\right)^{\kappa} & d = 3 \\ \text{remains open} & d = 2 \\ C\left(\frac{\varepsilon}{\delta}\right)^{1/2} & d = 1 \end{cases}$$

Example: FMM-HMM (J. Huang, ...):

Evaluate

$$\phi(x) = \int_{\Omega} \frac{q(y, \frac{y}{\varepsilon})}{|x-y|} dy$$

q is smooth, periodic in the 2nd variable with period I.

- Direction application of FMM: Cost $= O(\varepsilon^{-3})$.
- HMM strategy:
  - Macro-solver: FMM
  - Data needed: Coefficients of multipole expansion:

$$M_{k,j}^{p} = \int_{C_{k,j}} q(y, \frac{y}{\varepsilon})(y - x_{k,j})^{p} dy$$

where $(C_{k,j}, x_{k,j})$ =j-th (box, box-center) at kth-level.

$$M_{k,j}^{p} \simeq \int_{C_{k,j}} \int_{I} q(y, z)(y - x_{k,j})^{p} dy dz$$

Total cost $= O(1)$.

Simulating gas dynamics using MD

$$m_j \frac{d^2 \mathbf{x}_j}{dt^2} = \sum_k \mathbf{f}_{jk}$$

Define

$$\tilde{\rho}(\mathbf{y}, t) = \sum_j m_j \delta(\mathbf{y} - \mathbf{y}_j(t)),$$

$$\tilde{\mathbf{m}}(\mathbf{y}, t) = \sum_j m_j \mathbf{b}_j(t) \delta(\mathbf{y} - \mathbf{y}_j(t)),$$

$$\begin{aligned}\tilde{E}(\mathbf{y}, t) = &\sum_j \frac{1}{2} m_j |\mathbf{b}_j(t)|^2 \delta(\mathbf{y} - \mathbf{y}_j(t)) \\ &+ \frac{1}{2} \sum_j (\sum_{i \neq j} V_0(|\mathbf{y}_j(t) - \mathbf{y}_i(t)|)) \delta(\mathbf{y} - \mathbf{y}_j(t)).\end{aligned}$$

$$\partial_t \tilde{\rho} + \nabla_{\mathbf{y}} \cdot \tilde{\mathbf{m}} = 0,$$
$$\partial_t \tilde{\mathbf{m}} + \nabla_{\mathbf{y}} \cdot \tilde{\sigma} = 0,$$
$$\partial_t \tilde{E} + \nabla_{\mathbf{y}} \cdot \tilde{\mathbf{J}} = 0$$

Irving-Kirkwood formula:

$$\tilde{\sigma} = \sum_i m_i \mathbf{v}_i(t) \otimes \mathbf{v}_i(t) \delta(\mathbf{x} - \mathbf{x}_i(t))$$
$$+ \frac{1}{2} \sum_i \sum_{j \neq i} (\mathbf{x}_i(t) - \mathbf{x}_j(t)) \otimes \mathbf{f}_{ij}(t)$$
$$\int_0^1 \delta\left(\mathbf{x} - (\mathbf{x}_j(t) + \lambda(\mathbf{x}_i(t) - \mathbf{x}_j(t)))\right) d\lambda,$$

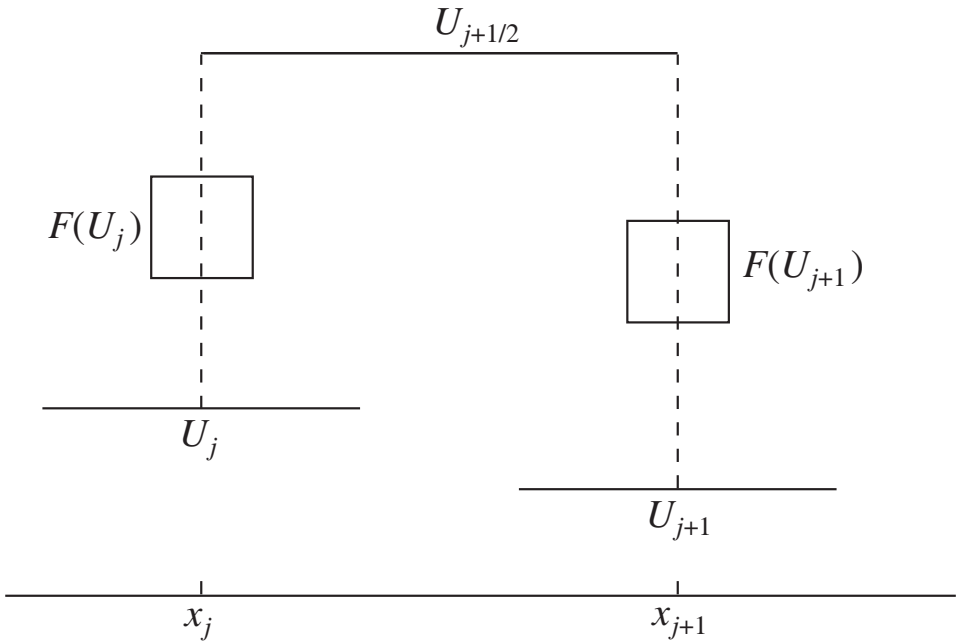
These are MD analog of the equations of gas dynamics.

1. Macro model: conservation law

$$U_t + \nabla F = 0$$

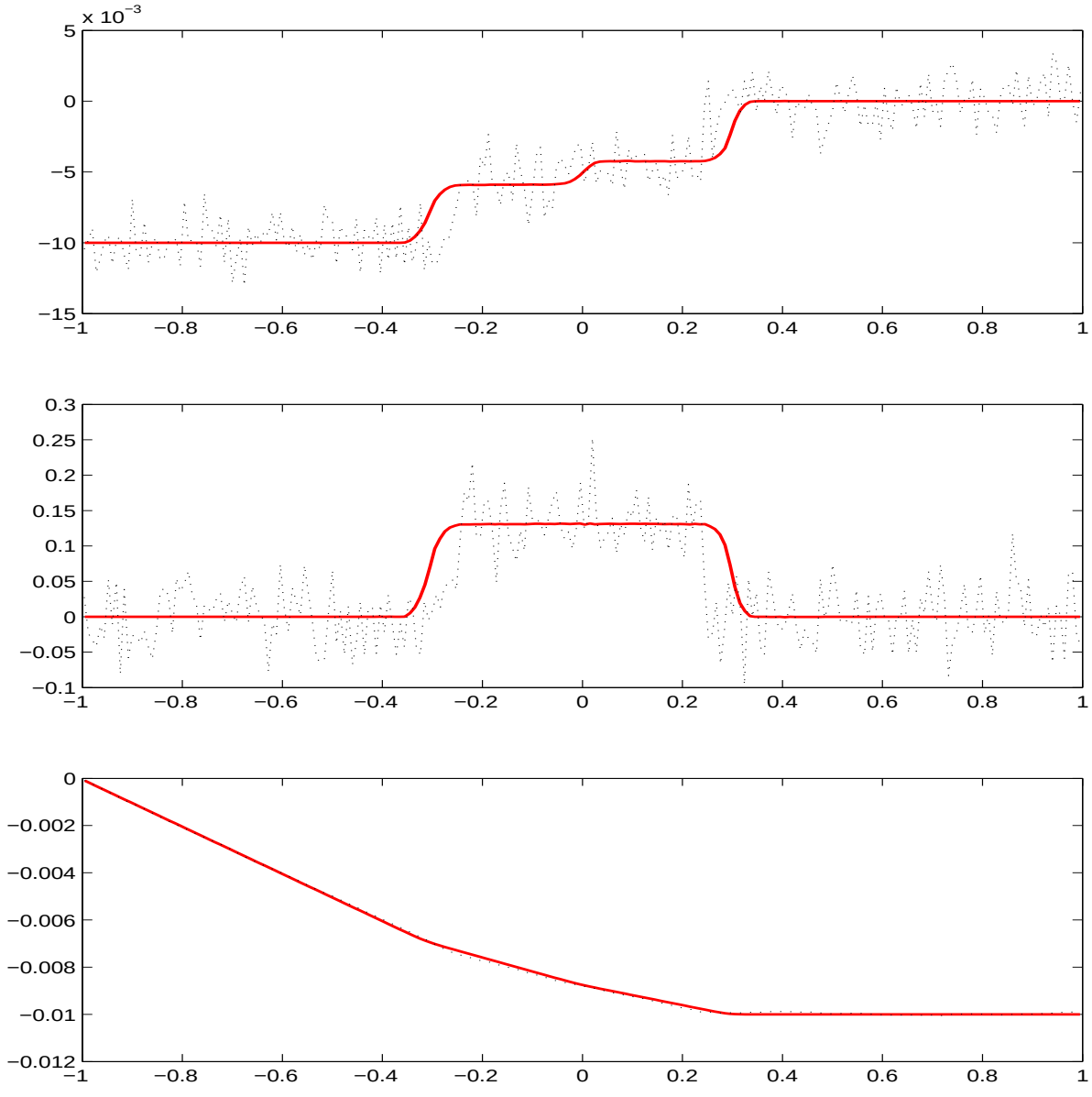
$U =$ mass, momentum, energy density. F is **unknown**.
Macro-solver: finite volume scheme -central type.

$$U_{j+1/2}^{n+1} = \frac{U_j^n + U_{j+1}^n}{2} - \frac{\Delta t}{\Delta x}\left(F_{j+1}^n - F_j^n\right).$$



2. Micro model: constrained MD, U = given (assume that F depends only on U)

3. Data estimator:
Need an expression for F, or F_j^n, from the MD data.



Solid line: computed solution; dashed line: full atom simulation (one realization). Top: strain; middle: velocity; bottom: displacement.

Stochastic Simulation Algorithm (SSA) with Disparate Rates

Species: $S_1, S_2, \cdots, S_N$; (x_i = number of molecules of species S_i).
Reaction channels: $R_1, R_2, \cdots, R_M$

$$R_j = (a_j(x), \nu_j)$$

$a_j(x)$ = rate function of j-th reaction
ν_j = state change vector

$$x \longrightarrow x + \nu_j$$

Multiscale reaction rates:

$$a(x) = (a^s(x), a^f(x))$$

$$a^s(x) = O(1), \quad a^f(x) = O(1/\varepsilon)$$

$$R^s = (a^s, \nu^s), \quad R^f = (a^f, \nu^f)$$

Example:

$$S_1 \underset{a_2^f}{\overset{a_1^f}{\rightleftarrows}} S_2, \qquad S_2 \underset{a_2^s}{\overset{a_1^s}{\rightleftarrows}} S_3, \qquad S_3 \underset{a_4^f}{\overset{a_3^f}{\rightleftarrows}} S_4,$$

$$\begin{aligned}
a_1^f &= 10^5 x_1, & \nu_1^f &= (-1, 1, 0, 0);\\
a_2^f &= 10^5 x_2, & \nu_2^f &= (1, -1, 0, 0);\\
a_3^f &= 10^5 x_3, & \nu_3^f &= (0, 0, -1, 1);\\
a_4^f &= 10^5 x_4, & \nu_4^f &= (0, 0, 1, -1);\\
a_1^s &= x_2, & \nu_1^s &= (0, -1, 1, 0);\\
a_2^s &= x_3, & \nu_2^s &= (0, 1, -1, 0).
\end{aligned}$$

Example: Heat shock response of E-Coli

Species	Initial value
DNA.σ^{32}	1
mRNA.σ^{32}	17
σ^{32}	15
RNAPσ^{32}	76
DNA.DnaJ	1
DNA.FtsH	0
DNA.GroEL	1
DnaJ	4640
FtsH	200
GroEL	4314
DnaJ.UnfoldedProtein	5×10^6
Protein	5×10^6
σ^{32}.DnaJ	2959
UnfoldedProtein	2×10^5

Table: List of species and their initial value (in number of molecules)

Scrivastava et al. *J. Theor. Biol.* (2002)

Reaction	Rate constant	Rates magnitude
Protein $\longrightarrow$ UnfoldedProtein $(\star)$	0.2	10^6
DnaJ+ UnfoldedProtein $\longrightarrow$ DnaJ.UnfoldedProtein $(\star)$	0.108	10^7
DnaJ.UnfoldedProtein $\longrightarrow$ DnaJ+ UnfoldedProtein $(\star)$	0.2	10^6
mRNA.σ^{32} $\longrightarrow$ σ^{32} + mRNA.σ^{32}	0.07	1.19
σ_{32} $\longrightarrow$ RNAPσ^{32}	0.7	10.5
RNAPσ^{32} $\longrightarrow$ σ^{32}	0.13	9.88
σ^{32} + DnaJ $\longrightarrow$ σ^{32}.DnaJ $(\star\star)$	3.62×10^{-3}	25.2
σ^{32}.DnaJ$\longrightarrow$ σ^{32} + DnaJ	4.4×10^{-4}	1.30
DNA.DnaJ + RNAPσ^{32} $\longrightarrow$ DnaJ + DNA.DnaJ + σ^{32}	8	3.71
σ^{32}.DnaJ + FtsH $\longrightarrow$ DnaJ + FtsH	1.42×10^{-5}	8.4
DNA.GroEL + RNAPσ^{32} $\longrightarrow$ GroEL + DNA.GroEL +σ^{32}	0.063	4.78
DNA.σ^{32} $\longrightarrow$ mRNA.σ^{32}	1.4×10^{-3}	1.4×10^{-3}
DnaJ $\longrightarrow$ degradation $(\star\star)$	6.4×10^{-10}	2.97×10^{-6}
FtsH $\longrightarrow$ degradation	7.4×10^{-11}	1.48×10^{-8}
GroEL $\longrightarrow$ degradation	1.8×10^{-8}	7.76×10^{-5}
mRNA.σ_{32} $\longrightarrow$ degradation	1.4×10^{-6}	2.38×10^{-5}
DNA.FtsH + RNAP.σ^{32} $\longrightarrow$ FtsH + DNA.FtsH + σ^{32}	4.88×10^{-2}	0

Table: Reaction list for the heat shock response model of E. Coli. The rate constant is the number c_i in $a_i(x) = c_i x_j$ for the reactions involving one species, or in $a_i(x) = c_i x_j x_k$ for the reactions involving two species.

Standard stochastic simulation algorithm (SSA)

Gillespie (1976), Bortz, Kalos and Lebowitz (1975)

$$a_0(x) = \sum a_j(x)$$

1. $r_1, r_2 =$ indep random variables with uniform distribution on [0, 1]

$$\delta t_{n+1} = -\frac{\ln r_1}{a_0(x)}$$

Define k_{n+1} by:

$$\frac{1}{a_0(x)} \sum_{j=1}^{k_{n+1}-1} a_j(x) < r_2 \leq \frac{1}{a_0(x)} \sum_{j=1}^{k_{n+1}} a_j(x).$$

2. Update the time and the state of the system

$$t_{n+1} = t_n + \delta t_{n+1} \ , \qquad x_{n+1} = x_n + \nu_{k_{n+1}}.$$

How do we deal with the multiscale rates

Previous work:
Scrivastava et al. (2001)
Haseltine and Rawlings (2002)
Takahashi et al. (2004)
Cao, Gillespie, Petzold (2005)

Issues remain to be addressed:

1. What is the effective process on slow time scale? What are the slow variables?
2. Develop multiscale algorithm **without** making closure assumptions on rate functions, e.g. $\langle x_j x_k \rangle = \langle x_j \rangle \langle x_k \rangle + \cdots$.

Nested SSA (E, Liu, Vanden-Eijnden, 2005)

1. Inner SSA (micro solver):
Run N replicas of SSA with fast reactions only, for time T_f.
Compute averaged slow rates:

$$\tilde{a}_j^s = \frac{1}{N} \sum_{k=1}^{N} \frac{1}{T_f} \int_0^{T_f} a_j^s(x_k(\tau)) d\tau,$$

2. Outer SSA (macro solver):
Run one step of SSA with the modified slow rates $\tilde{R}^s = (\tilde{a}^s, \nu^s)$.

Related work by D. Vlachos et. al (2005)

Features, Issues and Extensions

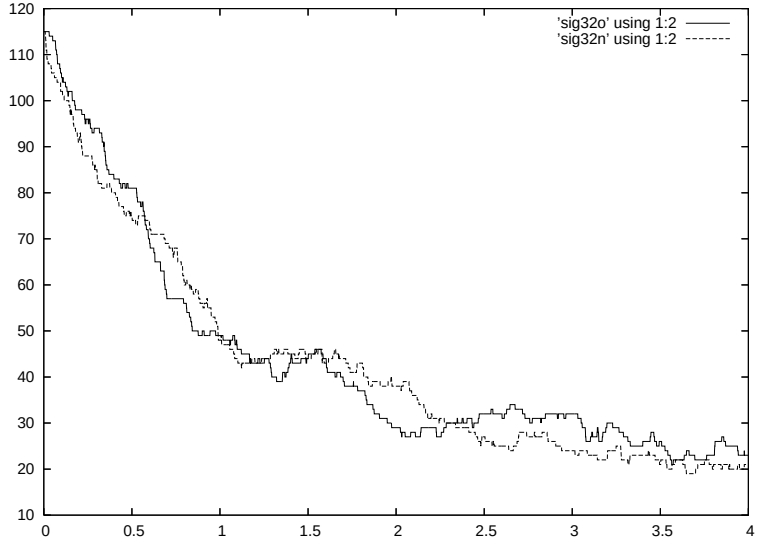
Features:

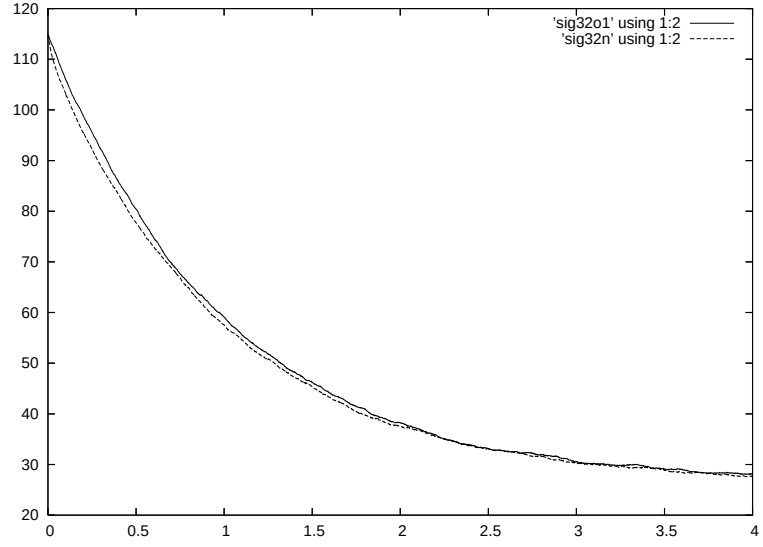
1. Simple: Involves a very simple change to the original SSA.
2. General: No ad hoc approximation of rate functions
3. Seamless: No need to think about fast/slow variables

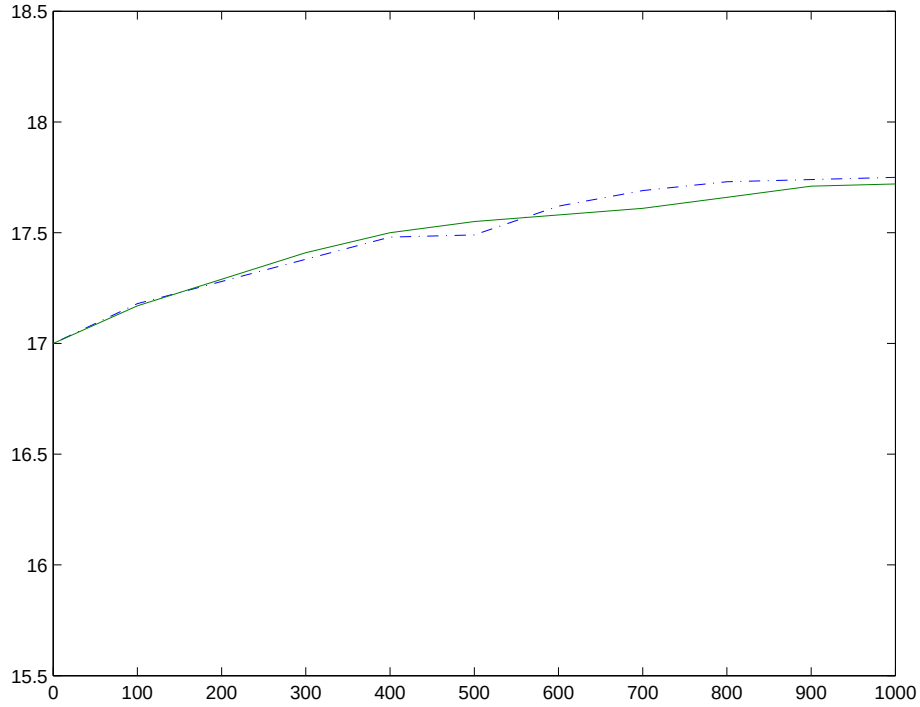
Extensions: More than two scales

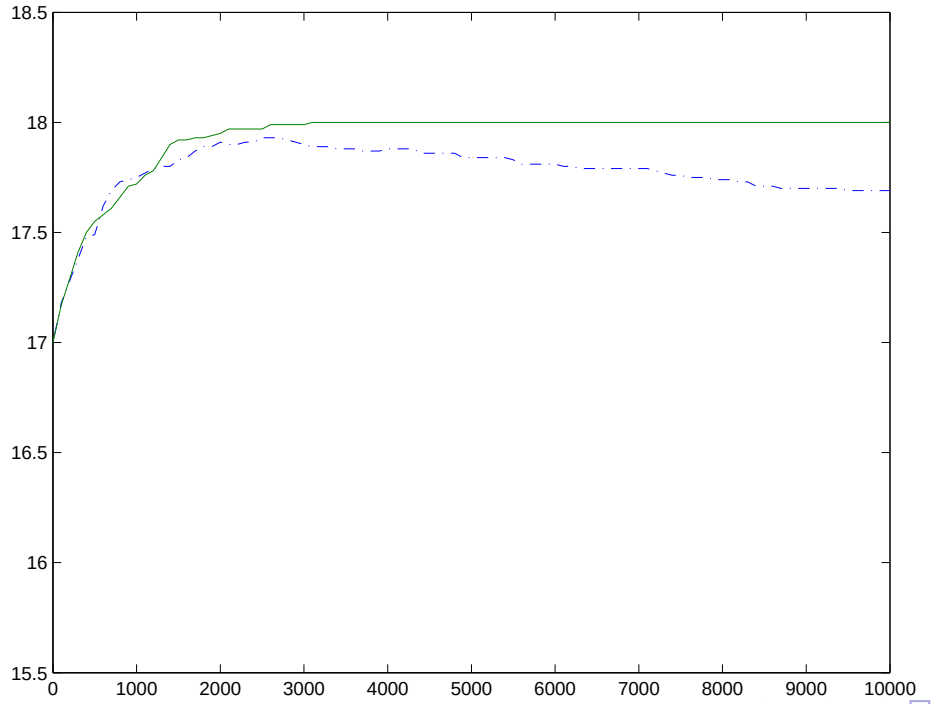
Hierachical view of SSA

Reaction	Rate constant	Rates magnitude
Protein $\longrightarrow$ UnfoldedProtein $(\star)$	0.2	10^6
DnaJ+ UnfoldedProtein $\longrightarrow$ DnaJ.UnfoldedProtein $(\star)$	0.108	10^7
DnaJ.UnfoldedProtein $\longrightarrow$ DnaJ+ UnfoldedProtein $(\star)$	0.2	10^6
mRNA.σ^{32} $\longrightarrow$ σ^{32} + mRNA.σ^{32}	0.07	1.19
σ_{32} $\longrightarrow$ RNAPσ^{32}	0.7	10.5
RNAPσ^{32} $\longrightarrow$ σ^{32}	0.13	9.88
σ^{32} + DnaJ $\longrightarrow$ σ^{32}.DnaJ $(\star\star)$	3.62×10^{-3}	25.2
σ^{32}.DnaJ$\longrightarrow$ σ^{32} + DnaJ	4.4×10^{-4}	1.30
DNA.DnaJ + RNAPσ^{32} $\longrightarrow$ DnaJ + DNA.DnaJ + σ^{32}	8	3.71
σ^{32}.DnaJ + FtsH $\longrightarrow$ DnaJ + FtsH	1.42×10^{-5}	8.4
DNA.GroEL + RNAPσ^{32} $\longrightarrow$ GroEL + DNA.GroEL +σ^{32}	0.063	4.78
DNA.σ^{32} $\longrightarrow$ mRNA.σ^{32}	1.4×10^{-3}	1.4×10^{-3}
DnaJ $\longrightarrow$ degradation $(\star\star)$	6.4×10^{-10}	2.97×10^{-6}
FtsH $\longrightarrow$ degradation	7.4×10^{-11}	1.48×10^{-8}
GroEL $\longrightarrow$ degradation	1.8×10^{-8}	7.76×10^{-5}
mRNA.σ_{32} $\longrightarrow$ degradation	1.4×10^{-6}	2.38×10^{-5}
DNA.FtsH + RNAP.σ^{32} $\longrightarrow$ FtsH + DNA.FtsH + σ^{32}	4.88×10^{-2}	0


'sig32o' using 1:2
'sig32n' using 1:2
120
110
100
90
80
70
60
50
40
30
20
10
0
0.5
1
1.5
2
2.5
3
3.5
4


'sig32o1' using 1:2
'sig32n' using 1:2
120
110
100
90
80
70
60
50
40
30
20
0
0.5
1
1.5
2
2.5
3
3.5
4


18.5
18
17.5
17
16.5
16
15.5
0
100
200
300
400
500
600
700
800
900
1000


18.5
18
17.5
17
16.5
16
15.5
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000


18.5

Slow variables and slow process

Slow variables do not change during fast reactions

$$v(x + \nu^f) = v(x)$$

If $v(x) = b \cdot x$, then

$$b \cdot \nu^f = 0$$

Slow variables:

$$y_j = b_j \cdot x$$

where $(b_1, b_2, \cdots b_J) =$ basis for span$\{\nu^f\}^\perp$.

Quasi-equilibrium: equilibrium states for the "virtual fast process", for fixed y, denote by $\mu_y(x)$.

Effective slow process:

$$\bar{a}_j(y) = \langle a_j(x) \rangle_y = \sum a_j(x) \mu_y(x)$$

Theorem: Assume f is a smooth function, then

$$|\mathbb{E} f(y_j^\varepsilon(t)) - \mathbb{E} f(y_j(t))| \leq C\varepsilon$$

for $t \in [0, T]$, where $y_j^\varepsilon = b_j \cdot x^\varepsilon$.

Error estimates

$$\mathbb{E}|\bar{a}_j - \tilde{a}_j^s| \leq C\Big(\frac{1}{1+T_f/\varepsilon} + \frac{1}{\sqrt{N(1+T_f/\varepsilon)}}\Big)$$

Optimal choice of N, T_f, given the error tolerance is λ

$$N = \frac{T_f}{\varepsilon} = \frac{1}{\lambda}$$

Cost (per unit time)

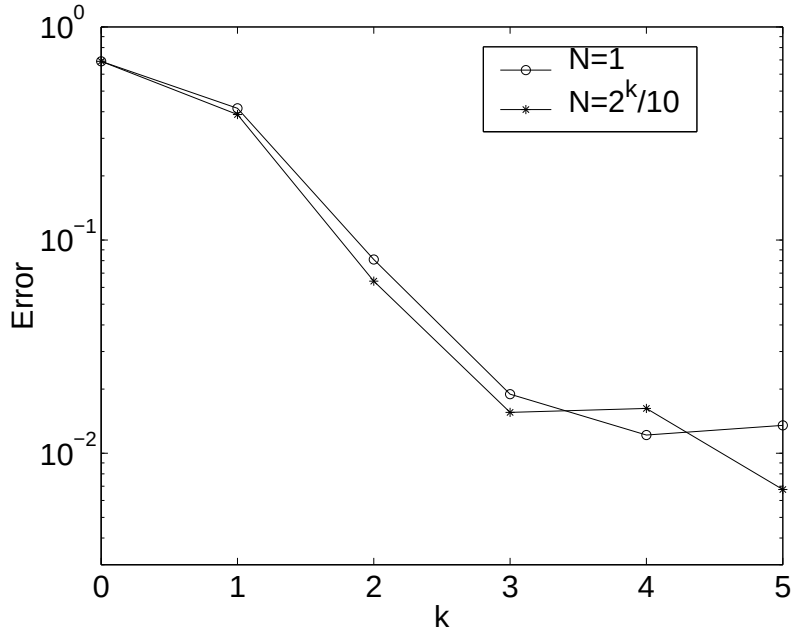
$$= \tilde{a}_0^s N T_f a_0^f = O(\frac{1}{\lambda^2})$$

Results: heat shock response model

$T = 10$

Direct SSA ($N_0 = 1000$)

$$\overline{\sigma^{32}} = 14.8 \pm 0.1, \qquad \mathrm{var}(\sigma^{32}) = 14.2 \pm 0.1.$$



Why start with the macro solver?

- need preconceived notion about how the macro model is like.
- unpopular at the time to do top-down.

Basic difficulty with bottom-up: Macroscopic character is important for designing stable and accurate algorithms.

Comment on coarse-grained Hamiltonian.

In practice, we often already know a lot about the macroscale behavior of the system. We should use that knowledge.

What if we make a wrong assumption about the macroscale model?

HMM is an "optimal approximation" strategy: It gives the optimal approximation within the class of models that it considers.

"Optimal approximation" based on what is known about the problem.

Common difficulty

All three methods, extended MG, HMM and 'equation-free' require going explicitly from macro to micro states, i.e. reconstruct initial conditions for the micro solver.
This can be complicated, e.g. when we reconstruct atomic positions from continuum fields.

	Macro to micro	micro to Macro
Extended multi-grid	interpolation	restriction (projection)
HMM	reconstruction	compression
Equation-free	lifting	restriction

Can we avoid doing this? Yes, in the HMM framework.

A simple trick: Modifying the parameters:

$$\dot{x} = 1 + y, \quad x(0) = 0,$$
$$\dot{y} = -\frac{1}{\varepsilon} z, \quad y(0) = 0,$$
$$\dot{z} = \frac{1}{\varepsilon} y \quad z(0) = 1,$$

- Change ε to a larger value ε'.
- Solve using standard methods.

- Used in Car-Parrinello molecular dynamics
  (Note: CPMD does not fit in the current HMM framework).
- Used in artificial compressibility method for incompressible flows (Chorin and Temam).

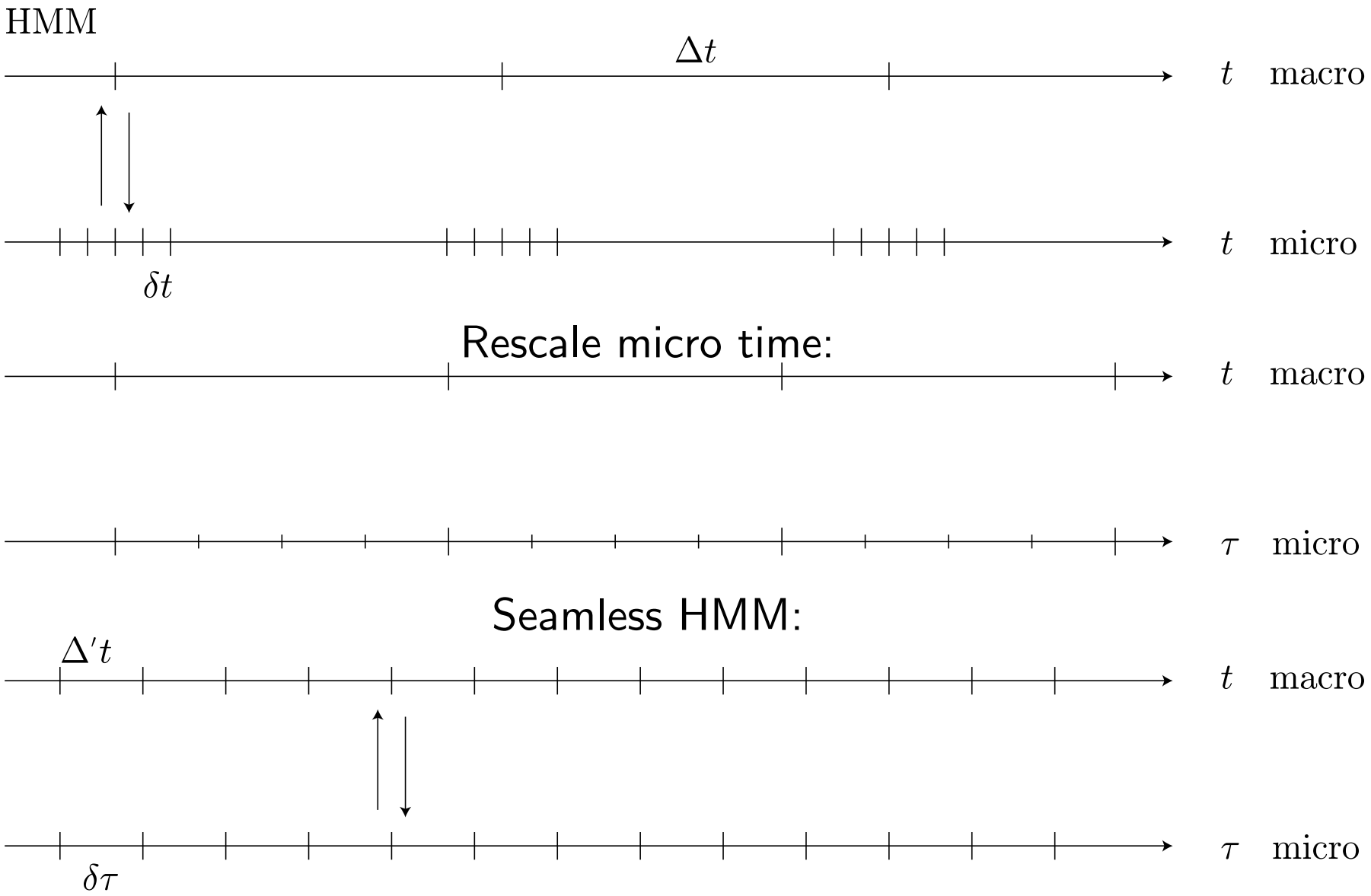
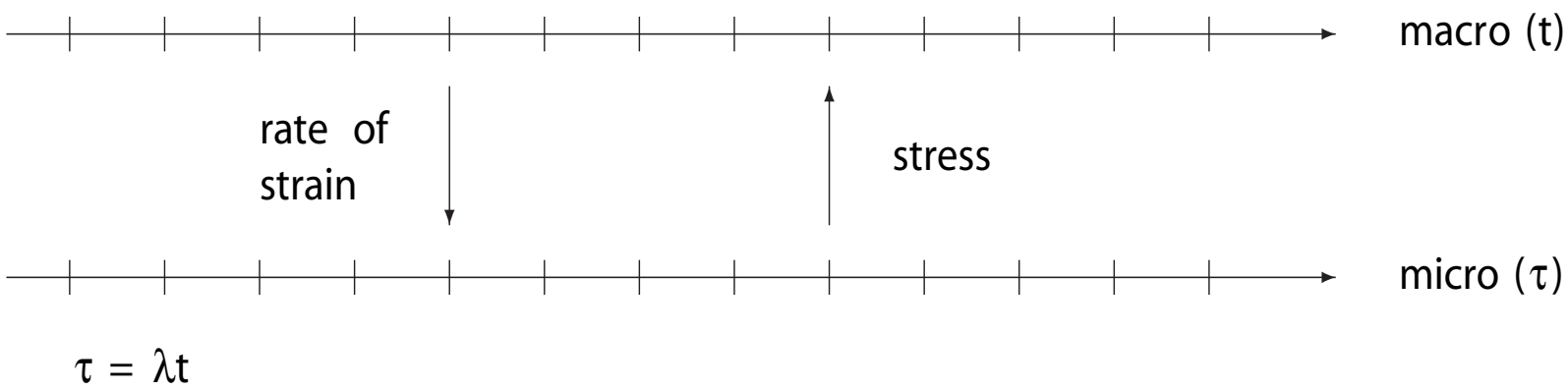


Figure: Illustration of HMM (upper panel) and the seamless algorithm (lower panel). Middle panel: rescaling the micro time scale.

A new seamless formulation (E, Ren and Vanden-Eijnden)

- Run micro-solver using its own time step δt
- Run macro-solver at a slower pace (time step $= \Delta t/M$) than what is necessary for accuracy. $M > \frac{\tau_\varepsilon}{\delta t} =$ relaxation steps needed for micro-solver
- Couple macro and micro every step



No need to reinitialize the micro-solver!

General formulation

Given $\{U^n, u^n\}$, find $\{U^{n+1}, u^{n+1}\}$ through:

- Macro-solver:
$$U^{n+1} = S_{\frac{\Delta t}{M}}(U^n; D^n)$$
where D^n stands for the needed data (such as stress). It may depend on the past (sliding averaging might be used).
- Micro-solver:
$$u^{n+1} = \mathcal{S}_{\delta t}(u^n; U^{n+1})$$
where U^{n+1} enters as constraints, such as boundary conditions.

M (which may depend on n) is the number of micro steps needed for relaxational process, i.e. $M\delta t >> \tau_\varepsilon$ (relaxation time scale).

This is seamless, very simple, and generalizable.

Examples of applications

- Complex fluids (Ren)
- Free energy calculations (Vanden-Eijnden et al.)
- Seamless string method (Vanden-Eijnden et al.)
- General homogenization problems (Assyr Abdulle's code)
- Stochastic simulation algorithms
- .....

Example: Fluids

Assume that the macro model is of the form:

$$\rho_0(\partial_t U + (U \cdot \nabla)U) + \nabla P = \nabla \cdot \tau$$

$$\nabla \cdot U = 0$$

$$\tau = \tau(\nabla U)$$

- Macro-solver: Projection method.
- Data to be estimated: $\tau = \tau(\nabla U)$.
- Micro-model: Molecular dynamics, bead-spring models for polymer chains.

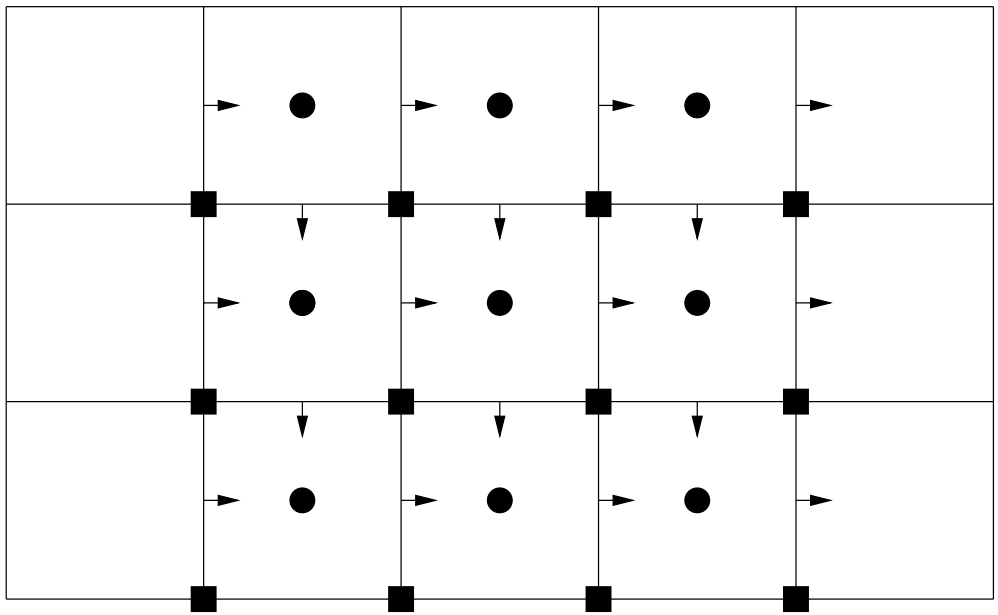
The assumed functional dependence of τ is important and is used as constraints in MD.

Macro-solver: Projection method (Chorin)

$$\rho_0 \frac{U^* - U^n}{\Delta t} = \nabla \cdot \tau^n$$

$$\rho_0 \frac{U^{n+1} - U^*}{\Delta t} + \nabla P^{n+1} = 0$$

$$\triangle P^{n+1} = \frac{\rho_0}{\Delta t} \nabla \cdot U^*$$



Computing stress from MD

Step 1. Constant rate-of-strain MD

See Weiqing Ren's work.

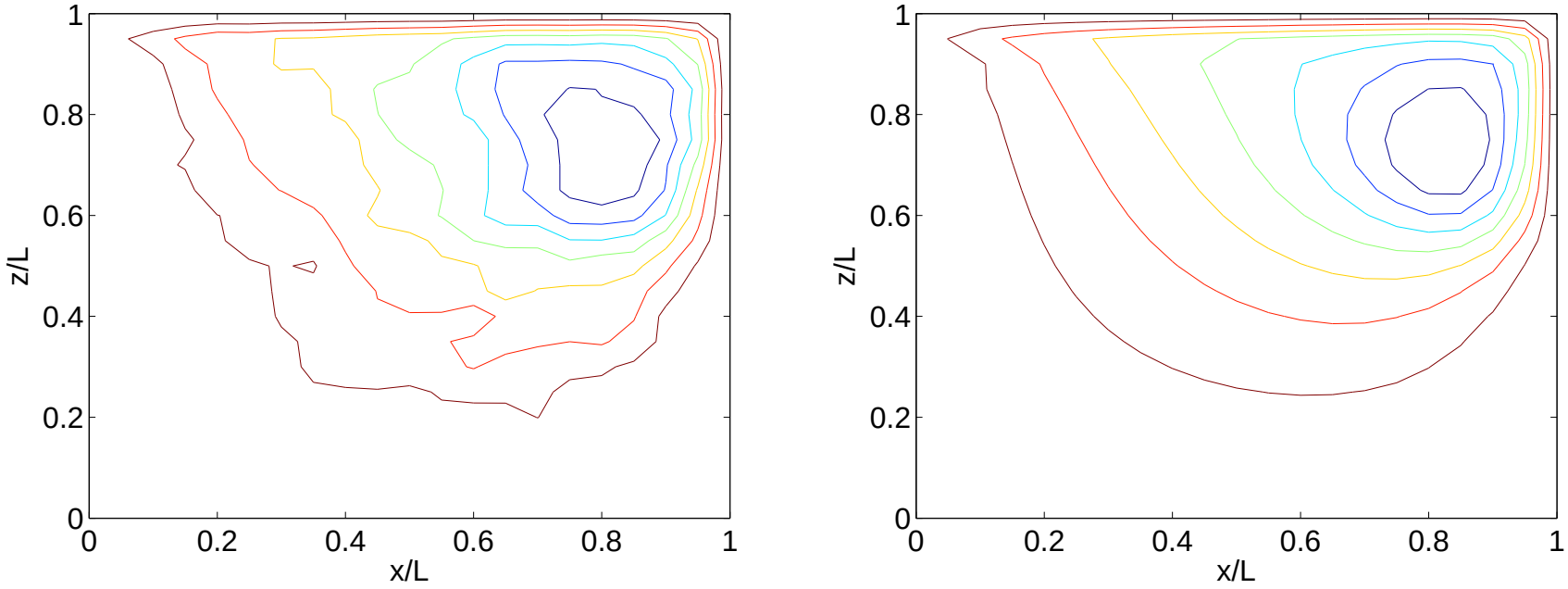


Figure: New: Flow lines of a Lennard-Jones fluid in the driven cavity flow at different times: $t = 7.5 \times 10^3,\ 1.0 \times 10^4,\ 1.25 \times 10^4,\ 2.25 \times 10^4$ from top to bottom. The left column is the result of the seamless multiscale method, and the right column is the numerical solution of the Navier-Stokes equation.

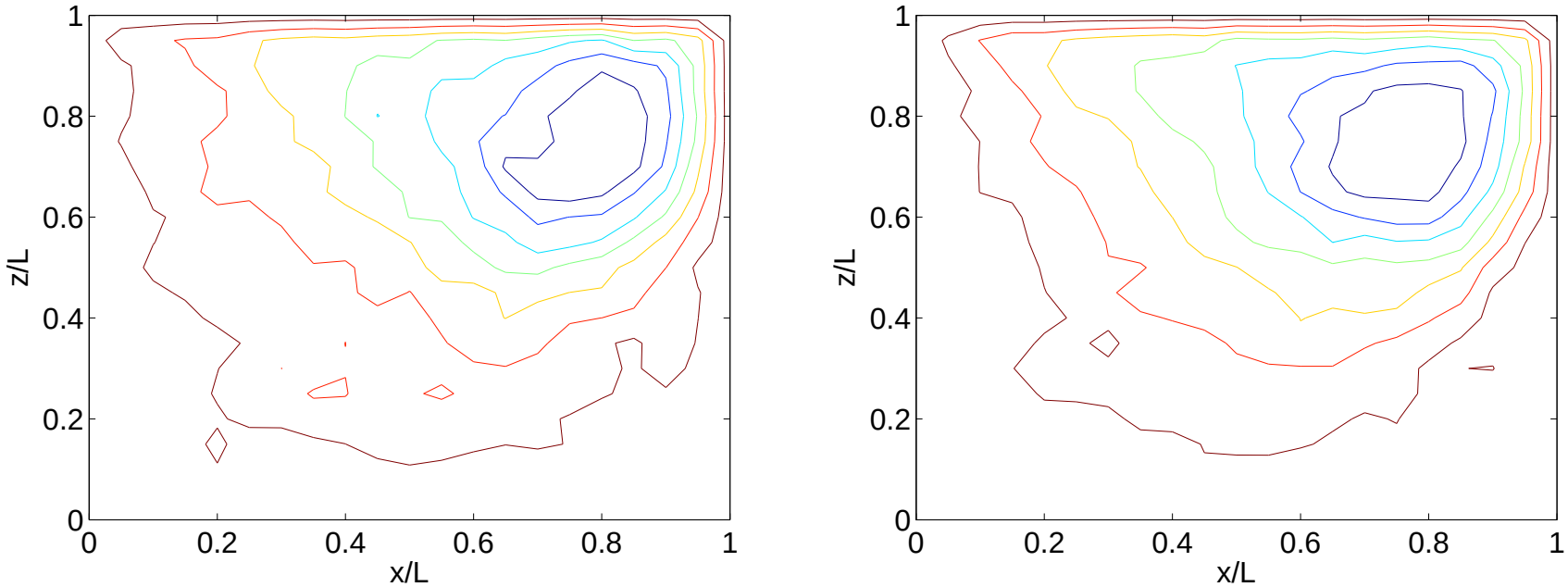


Figure: New: Flow lines of a polymer fluid in the driven cavity flow at different times: $t = 7.5 \times 10^3,\ 1.0 \times 10^4,\ 1.25 \times 10^4,\ 2.0 \times 10^4$ from top to bottom. The left column is the result of the seamless multiscale method with the macro time step $\Delta' t = 0.5$; the right column is the result of the multiscale method with a smaller macro time step $\Delta' t = 0.25$;

Why hasn't multiscale modeling been more successful?

- There are serious problems with the microscale models (e.g. SSA).
- Design constrained micro solver.
- The signal to noise ratio problem.
- Dealing with continuum of scales.

Another class of problems:

"Heterogeneous domain decomposition".

- Coupling quantum mechanics and classical mechanics (QM-MM, Warshel and Levitt, 1975).
- Coupling molecular dynamics and continuum mechanics.
- Coupling kinetic equation with fluid dynamics equations.

How can we do better?

Is HMM worthwhile?

Given that:

- similar ideas have been used in various applications before
- it is more of a philosophy, rather than a concrete algorithm
- we still have to make assumptions about how the macroscale model looks like (which was a rather unpopular strategy during the early days of multiscale modeling)
- .....

is HMM worthwhile? Yes, since

- It is a very nice way or organize our ideas about multiscale algorithms.
- It allows us to make best use of what we know about the problem at the different scales.
- It is the only existing general framework that allows us to perform error analysis.

Analogy: The finite element method.

Acknowledgement

- Gil Ariel, UT/Austin
- Bjorn Engquist, UT/Austin
- Xiantao Li, Penn State Univ.
- Di Liu, Michigan State Univ.
- Pingbing Ming, Chinese Academy of Science
- Weiqing Ren, Courant Institute
- Richard Tsai, UT/Austin
- Eric Vanden-Eijnden, Courant Institute