Sequential multiscale modeling using sparse representation

Carlos J. García-Cervera,[1,*] Weiqing Ren,[2,†] Jianfeng Lu,[3,‡] and Weinan E[4,§]

[1]*Mathematics Department, University of California, Santa Barbara, CA 93106.*

[2]*Courant Institute of Mathematical Sciences, New York University, New York, NY 10012.*

[3]*Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544.*

[4]*Department of Mathematics and PACM, Princeton University, Princeton, NJ 08544.*

(Dated: June 8, 2007)

Abstract

The main obstacle in sequential multiscale modeling is the pre-computation of the constitutive relation which often involves many independent variables. The constitutive relation of a polymeric fluid is a function of six variables, even after making the simplifying assumption that stress depends only on the rate of strain. Precomputing such a function is considered too expensive. Hence the value of sequential multiscale modeling is limited to "parameter passing". Here we demonstrate that sparse representations can be used to drastically reduce the computational cost for precomputing functions of many variables. This strategy dramatically increases the efficiency of sequential multiscale modeling, making it very competitive in many situations.

In recent years, multiscale modeling has attracted a great deal of attention across a wide spectrum of disciplines in science and engineering [1–4]. This has opened up the possibility of analyzing the macroscopic behavior of a system based on first principles, by linking together macroscopic and microscopic models, bypassing the necessity of making ad hoc modeling assumptions such as the ones that underly the empirical constitutive relations in continuum mechanics. In broad terms, such multiscale methodologies can be divided into two categories, sequential coupling methods and concurrent coupling methods [1, 4]. In the sequential strategy, the needed model input for the macroscale model is computed from microscale models beforehand. One then has an effectively closed macroscale model which can be used for analytical or computational purpose. Such a strategy has a very long history. It is a standard practice to obtain the transport coefficients of fluids such as viscosity or diffusion coefficients from kinetic theory or molecular dynamics simulations. Other examples of sequential coupling include calibrating empirical atomistic potentials used in molecular dynamics using models from quantum mechanics, determining the rates used in Monte Carlo simulation using molecular dynamics or quantum mechanics models, computing the equation of state for gases using kinetic theory, etc. However, for a long time, this procedure has been limited to the passage of a few parameters, due to the fact that the computational cost associated with computing the full constitutive relation is often too expensive. For example, the constitutive relations for fluids in general depend on at least six variables, and precomputing a function of six variables is simply too expensive. Therefore, one has to make a priori assumption about the functional form of the constitutive relation, and microscale models are then used to determine a few parameters in the functional form. The assumed functional form is often quite ad hoc, and this has been the main drawback for sequential modeling.

The philosophy of concurrent coupling is to access such information "on-the-fly" as the computation proceeds. The advantage of such a concurrent strategy is quite clear: Even though the needed constitutive relation may depend on many variables, in any particular simulation, one does not need to know the constitutive relation within the full range of these variables – only the values that actually occur in the simulation are needed, and these might be a very small subset of the entire range. The best example for illustrating the advantage of such a concurrent approach is the Car-Parrinello molecular dynamics in which the needed constitutive relation is the atomic potential. This function may depend on the coordinates

of all the atoms in the system, which can easily be a function of tens of thousands variables. However, in any particular simulation, one does not need to know this function entirely, but only the values needed for the particular sequence of atomic configurations that occur in the simulation, and this is a tiny subset of the tens of thousands dimensional space [5]. A very informative discussion of the relative merits of sequential and concurrent coupling strategies can be found in [1].

However, whenever possible, it is still advantageous to have the constitutive relations precomputed, since this information can be used for many other purposes, such as analyzing the properties of the system. Knowing the constitutive relations of a fluid helps us to understand the nature of the macroscopic response of the fluid, whether it is shear thinning or shear thickening, for example. The main purpose of the present paper is to demonstrate that the sequential coupling strategy can be made much more powerful through the use of *sparse representation*. For example, instead of representing the functions on tensor-product grids, one can use sparse grids and this drastically decreases the number of function values needed. There exist efficient interpolation algorithms over such grids, which allow us to make use of such a table over the sparse grids for many different purposes, including sequential multiscale modeling [6].

We begin with a brief review of sparse representation. Let f be a smooth function of $\boldsymbol{x} = (x_1, \cdots, x_d)$ on the d-dimensional hypercube Ω^d. We would like to represent f efficiently. The simplest way to do this is to use a *regular grid*. Let us consider the family of standard rectangular grids $\Omega_{\boldsymbol{l}}$ with multi-index $\boldsymbol{l} = (l_1, \cdots, l_d) \in \mathbb{N}^d$. In $\Omega_{\boldsymbol{l}}$ the grid points are:

$$\boldsymbol{x}_{\boldsymbol{l},\boldsymbol{i}} = (x_{l_1,i_1}, \cdots, x_{l_d,i_d}) = \boldsymbol{i} \cdot \boldsymbol{h}_{\boldsymbol{l}}, \quad \boldsymbol{0} \le \boldsymbol{i} \le 2^{\boldsymbol{l}}, \tag{1}$$

where $\boldsymbol{h}_{\boldsymbol{l}}$ is the mesh size given by

$$\boldsymbol{h}_{\boldsymbol{l}} = (h_{l_1}, \cdots, h_{l_d}) = 2^{-\boldsymbol{l}}. \tag{2}$$

Here we have adopted the component-wise operations for multi-indices:

$$\boldsymbol{a} \cdot \boldsymbol{b} = (a_1 b_1, \cdots, a_d b_d) \quad \text{and} \quad 2^{\boldsymbol{a}} = (2^{a_1}, \cdots, 2^{a_d}).$$

On each grid $\Omega_{\boldsymbol{l}}$, we define the linear space $V_{\boldsymbol{l}}$ spanned by piecewise d-linear functions

$$V_{\boldsymbol{l}} = \text{span}\{\phi_{\boldsymbol{l},\boldsymbol{i}},\ \boldsymbol{0} \le \boldsymbol{i} \le 2^{\boldsymbol{l}}\}. \tag{3}$$

where $\phi_{\boldsymbol{l},\boldsymbol{i}}$ is of the form:

$$\phi_{\boldsymbol{l},\boldsymbol{i}}(\boldsymbol{x}) = \prod_{j=1}^{d} \phi_{l_j,i_j}(x_j), \tag{4}$$

ϕ_{l_j,i_j} is the standard one-dimensional hat function with support $[(i_j-1)h_{l_j}, (i_j+1)h_{l_j}] \cap [0,1]$. For $\boldsymbol{l} = (n, \cdots, n)$, we get the uniform grids Ω_n with $2^n + 1$ grid points in each direction and the associated space V_n. It is easy to see that Ω_n has $\mathcal{O}(2^{nd})$ grid points. Denote by f_n the interpolation of f in V_n, we have the error estimate

$$\|f - f_n\|_{L^p} = \mathcal{O}(h_n^2). \tag{5}$$

Therefore, regular grid requires $\mathcal{O}(2^{nd})$ grid points to achieve $\mathcal{O}(h_n^2)$ accuracy in L^p norm.

A different strategy is to use the *sparse grids* introduced by Zenger [7], which is based on a high-dimensional multiscale basis derived from a one-dimensional multiscale basis by tensor product construction. Following [6], define the difference spaces

$$W_{\boldsymbol{l}} = V_{\boldsymbol{l}} \backslash \bigcup_{j=1}^{d} V_{\boldsymbol{l}-\boldsymbol{e}_j}, \tag{6}$$

where $\boldsymbol{e}_j$ denotes the j-th unit vector and for notational ease, we set $V_{\boldsymbol{l}}$ to be the empty set if $\boldsymbol{l} \ngeq \boldsymbol{0}$. Using the difference spaces, it is easy to see that regular grids can be rewritten as

$$V_n = \bigoplus_{|\boldsymbol{l}|_\infty \le n} W_{\boldsymbol{l}}, \tag{7}$$

where $|\cdot|_\infty$ is the discrete l^∞ norm. Consider the subspace $V_n^{(s)}$ of V_n obtained by replacing $|\boldsymbol{l}|_\infty \le n$ by $|\boldsymbol{l}|_1 \le n + d - 1$ (with $\boldsymbol{l} \ge \boldsymbol{0}$) in (7):

$$V_n^{(s)} = \bigoplus_{|\boldsymbol{l}|_1 \le n+d-1} W_{\boldsymbol{l}}, \tag{8}$$

where $|\cdot|_1$ is the discrete l^1 norm. Sparse grids are the grids corresponding to the approximation spaces $V_n^{(s)}$, denoted by $\Omega_n^{(s)}$. A straightforward calculation [6] shows that there are $\mathcal{O}(n^{d-1}2^n)$ grid points in $\Omega_n^{(s)}$. Denote by $f_n^{(s)}$ the interpolation of f using sparse grids, it was proven [6] that the discretization error satisfies

$$\|f - f_n^{(s)}\|_{L^p} = \mathcal{O}(n^{d-1}h_n^2), \tag{9}$$

provided that the mixed derivative $\partial^{2d} f / \prod_j \partial x_j^2$ exists. For the same accuracy, we now need much fewer grid points than the regular grids.

Here we have only discussed sparse representation using piecewise d-linear interpolation on grid points. The same idea can be extended to other representation, such as high order polynomials or Fourier series [8]. In this paper, we will limit ourselves to sparse grids.

To put this idea into practical use in sequential coupling, we need to: (1) precompute the constitutive relation on a sparse grid; (2) efficiently interpolate the computed values in order to obtain the constitutive relation elsewhere. Let us first discuss the second part, since the first part is problem-dependent.

Given the values of f on grid points in $\Omega_n^{(s)}$, we use the combination technique for interpolation to construct the approximation function f_n in $V_n^{(s)}$. This is an extrapolation method which works on a sparse grid [9, 10]. The interpolation formula is

$$f_n^{(c)}(\boldsymbol{x}) = \sum_{q=0}^{d-1} (-1)^q \binom{d-1}{q} \sum_{|\boldsymbol{l}|_1 = n+(d-1)-q} f_{\boldsymbol{l}}(\boldsymbol{x}). \tag{10}$$

Here $f_{\boldsymbol{l}}(\boldsymbol{x})$ is the standard interpolation on the rectangular grids $\Omega_{\boldsymbol{l}}$, given by

$$f_{\boldsymbol{l}}(\boldsymbol{x}) = \sum_{\boldsymbol{0} \le \boldsymbol{i} \le 2^{\boldsymbol{l}}} f(\boldsymbol{x}_{\boldsymbol{l},\boldsymbol{i}}) \phi_{\boldsymbol{l},\boldsymbol{i}}(\boldsymbol{x}). \tag{11}$$

The combination solution $f_n^{(c)}$ is in general not equal to the Galerkin solution $f_n^{(s)}$, but its accuracy is usually of the same order [9].

We now turn to some concrete examples.

For our first example, we demonstrate how to analyze the macroscopic elastic deformation of solids using an atomistic model. In elasticity theory, one often obtains the required constitutive relation using empirical stress-strain relation such as Hooke's law in linear elasticity. Our purpose is to obtain such constitutive relations directly from a microscopic model. To this end, we precompute the stress-strain relation using the Cauchy-Born rule [11]. To fix ideas, we consider a rectangular sample of Aluminum, occupying a volume $\Omega \subset \mathbb{R}^3$. In the undeformed state, the atoms are arranged in a face-centered cubic (FCC) lattice. We denote by V_0 the unit cell of the crystal, and by $\{\boldsymbol{a}_i\}_{i=1}^3$ a basis for the crystal lattice.

Given a deformation $\boldsymbol{u} : \Omega \to \mathbb{R}^3$, we denote by $\boldsymbol{F} = \nabla \boldsymbol{u}$ the deformation tensor, and define the elastic energy density as

$$\omega[\boldsymbol{F}] = \frac{1}{|V_0|} W_{\mathrm{CB}}[\boldsymbol{F}], \tag{12}$$

where the Cauchy-Born energy, $W_{\mathrm{CB}}[\boldsymbol{F}]$, is the energy of the deformed unit cell, obtained by transforming the basis vectors as $\boldsymbol{b}_i = \boldsymbol{F}\boldsymbol{a}_i$, $i = 1, 2, 3$. Due to frame indifference, the

elastic energy depends only on the right Cauchy-Green strain tensor (RCGST), $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$, so we can write $\omega[\boldsymbol{F}] = \omega[\boldsymbol{C}]$, and $W_{\mathrm{CB}}[\boldsymbol{F}] = W_{\mathrm{CB}}[\boldsymbol{C}]$.

At the atomistic level we consider the embedded-atom method (EAM) [12], and compute $W_{\mathrm{CB}}[\boldsymbol{C}]$ using the glue potential parameterized by Ercolessi and Adams [13].

Consider now a finite element discretization of the domain Ω, defined by a triangulation $\mathcal{T}$, and the space V_h generated by the basis functions $\{\boldsymbol{\varphi}_i\}_{i=1}^M$. A deformation is represented by

$$\boldsymbol{u}(\boldsymbol{x}) = \sum_{i=1}^{M} \boldsymbol{u}_i \boldsymbol{\varphi}_i(\boldsymbol{x}). \tag{13}$$

Given such a deformation, we write the total energy as

$$E[\boldsymbol{u}] = \sum_{K\in\mathcal{T}} \int_K \omega(\boldsymbol{F}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}, \tag{14}$$

where $\omega(\boldsymbol{F})$ is the energy density defined in (12). The integral on the element $K \in \mathcal{T}$ can be approximated using numerical quadrature,

$$\int_K \omega(\boldsymbol{F}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x} \approx \sum_{j=1}^{p} \alpha_j \omega(\boldsymbol{F}(\boldsymbol{x}_j)), \tag{15}$$

where $\{\boldsymbol{x}_j\}_{j=1}^p \subset K$ are the quadrature nodes.

The RCGST is a symmetric, positive definite tensor, and therefore the elastic energy density can be parameterized as a function in six-dimensional space. This parameterization is done using the sparse grids described above. To compute the force, we need the derivative of W_{CB} with respect to $\boldsymbol{u}$, which can be written as

$$\frac{\partial W_{\mathrm{CB}}}{\partial \boldsymbol{u}} = \sum_{i,j} \frac{\partial W_{\mathrm{CB}}}{\partial C_{ij}} \frac{\partial C_{ij}}{\partial \boldsymbol{u}}. \tag{16}$$

The derivatives of the RCGST are known explicitly, so we need to store in the interpolating table the energy and the gradient of the energy with respect to the RCGST.

We construct the table for the symmetric, positive definite tensor

$$\boldsymbol{C} = \begin{pmatrix} 1+\lambda_1 & \lambda_2 & \lambda_3 \\ \lambda_2 & 1+\lambda_4 & \lambda_5 \\ \lambda_3 & \lambda_5 & 1+\lambda_6 \end{pmatrix}, \tag{17}$$

where $\lambda_i \in [-0.25, 0.25]$, $i = 1, \ldots, 6$. We use seven hierarchical levels for the definition of the grids in six dimensional space. The table is composed of 923 grids, totaling $2,572,288$

grid points. We compared the interpolated values with the EAM potential computed on a uniform grid, which showed that the energy and the gradient can be reconstructed with up to four digits of accuracy. To achieve the same accuracy with a regular grid, $128^6 \geq 4 \times 10^{12}$ grid points would be required. The use of sparse grids, as opposed to a uniform grid, results in an improvement of about six orders of magnitude in this case.

By construction, the sparse grid representation is a union of grids, and the table can be computed on each grid independently. We take advantage of this by computing the tables in parallel. The combination technique described above for the interpolation on the sparse grids can also be performed in parallel.

As an illustration we show in Figure 1 the elastic energy for a deformation of the form

$$\boldsymbol{F} = \begin{pmatrix} 1+\lambda & \mu & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{18}$$

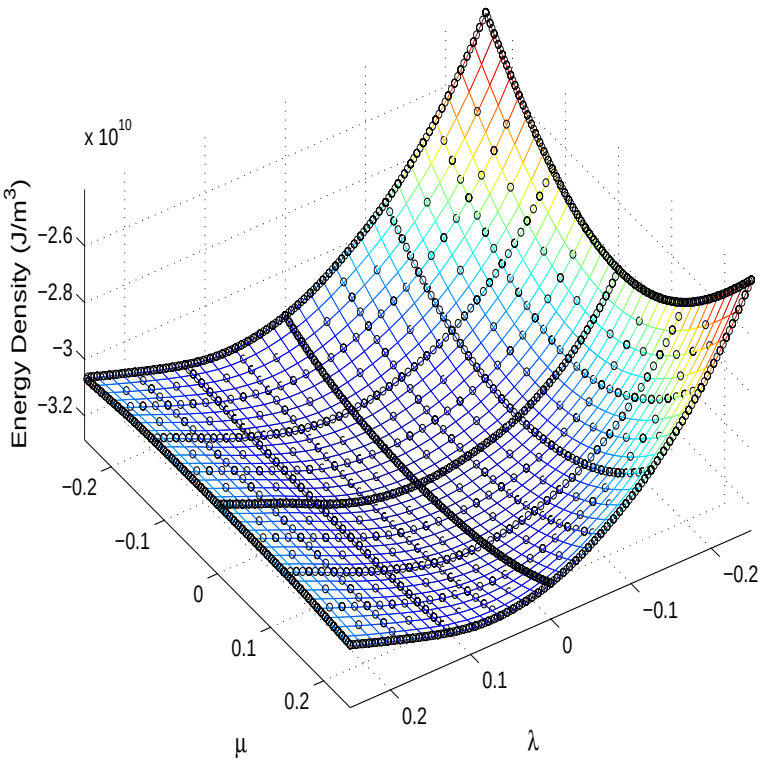

FIG. 1: Elastic energy for the shear and expansion deformation. We plot the energy interpolated to a uniform grid. The sparse grid nodes are superimposed. (color online)

In the plot we show the energy interpolated on a uniform grid. The two-dimensional sparse grid used is superimposed.

For the next example, we analyze the macroscopic behavior of fluids that consist of chain molecules. As before, we precompute the constitutive relation for the stress tensor from an underlying microscopic model. We assume that the stress depends only on the symmetric part of the rate of strain. Therefore, up to a rigid rotation, the constitutive relation is a function of two variables for 2d flows, and a function of six variables for 3d flows.

The stresses on a sparse grid of Ω for a polymer fluid are computed using non-equilibrium MD simulations. The polymers are modeled by the bead-spring model. Each polymer consists of 12 beads. Neighboring beads are connected by a spring force modeled by the FENE potential:

$$V(r) = \begin{cases} \frac{1}{2}kr_0^2 \ln\left(1 - \left(\frac{r}{r_0}\right)^2\right), & r < r_0, \\ \infty, & r \geq r_0. \end{cases} \tag{19}$$

Here we used $k = 30$ and $r_0 = 1.5$. In addition, the Lennard-Jones (LJ) potential is applied between all the beads, to prevent overlapping of the beads. In the following presentation, all the quantities are expressed in terms of the reduced LJ units. The MD simulation for each grid point in Ω is conducted on a finite box which deforms its shape according to the given rate of strain. To maintain a reasonable aspect ratio for the simulation box, the initial box is carefully chosen based on the results of Kraynik and Reinelt on reproducible lattices [14–16]. The number density of the beads is 0.81, the temperature is fixed at $T = 1.1$. The value of stress is obtained by averaging the Irving-Kirkwood formula over $50,000$ MD steps after $12,000$ steps of equilibration. The MD step size is 0.002.

The computed stress tensor can be used to solve various macroscale problems. As an example, we computed the dynamics of a driven cavity flow. The system is driven by a constant motion of the upper boundary in the horizontal direction. The system size is 3000×3000. The upper wall moves at a velocity $U = 10$. The conservation laws for the mass and momentum are solved using the projection method on a staggered grid with 100 grid points in each direction [17]. The velocity field at steady state and the normal stree interpolated to a uniform grid are shown in Figure 2.

We have discussed how to use sparse representation to precompute the constitutive relation and to do sequential multiscale modeling. The usefulness of sparse representation is not limited to these: It can be used to represent any smooth function of many variables.

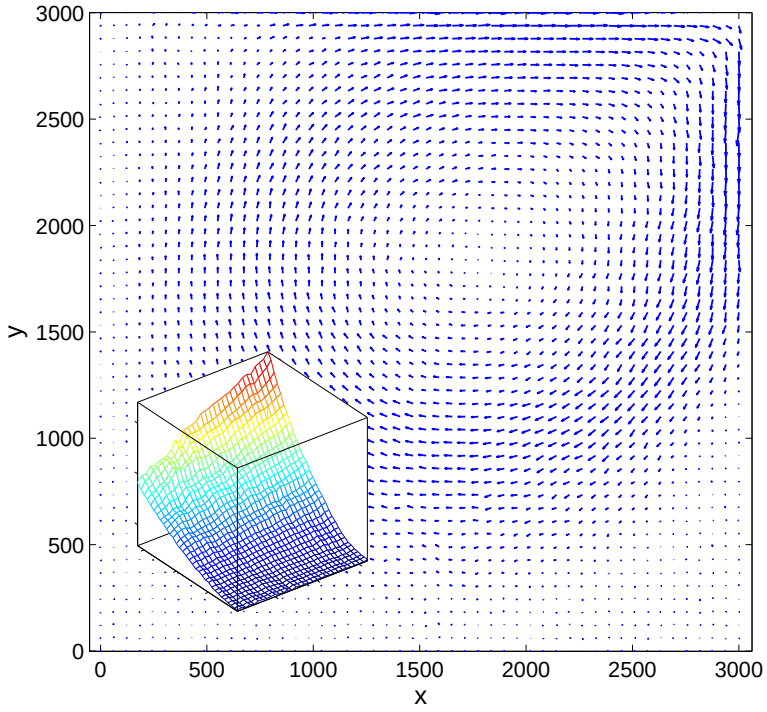

FIG. 2: Velocity field at steady state in the driven cavity flow. Inset in the figure is the normal stress. (color online)

Other examples that come to mind for which sparse representation can be useful include: Modeling and calibrating empirical potentials, exploration of free energy landscapes with many coarse-grained variables, and combined sequential-concurrent coupling methods. It should also be remarked that at the present time, sparse representation is only practical when the number of dependent variables is not too large. For example, it will not be able to replace concurrent strategies such as the Car-Parrinello method in the near future.

The work of Carlos J. García-Cervera is supported in part by NSF grants DMS-0411504 and DMS-0505738. The work of Weiqing Ren is supported in part by NSF grant DMS-0604382. The work of Jianfeng Lu and Weinan E is supported in part by ONR grant N00014-01-0674, DOE grant DE-FG02-03ER25587 and NSF grant DMS-0407866.

---

* Electronic address: cgarcia@math.ucsb.edu

† Electronic address: weiqing@cims.nyu.edu

‡ Electronic address: jianfeng@math.Princeton.EDU

§ Electronic address: weinan@Princeton.EDU

[1] F. F. Abraham, J. Q. Broughton, N. Bernstein, and E. Kaxiras, Phys. Rev. B **60**, 2391 (1999).

[2] A. Brandt, in *Multiscale and multiresolution methods* (Springer, Berlin, 2002), vol. 20 of *Lect. Notes Comput. Sci. Eng.*, pp. 3–95.

[3] W. E and B. Engquist, Notices Amer. Math. Soc. **50**, 1062 (2003).

[4] W. E, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden, Commun. Comput. Phys. **2**, 367 (2007).

[5] R. Car and M. Parrinello, Phys. Rev. Lett. **55**, 2471 (1985).

[6] H.-J. Bungartz and M. Griebel, Acta Numer. **13**, 147 (2004).

[7] C. Zenger, in *Parallel algorithms for partial differential equations (Kiel, 1990)* (Vieweg, Braunschweig, 1991), vol. 31 of *Notes Numer. Fluid Mech.*, pp. 241–251.

[8] J. Shen and Y. Zhao (2007), preprint.

[9] M. Griebel, M. Schneider, and C. Zenger, in *Iterative methods in linear algebra (Brussels, 1991)* (North-Holland, Amsterdam, 1992), pp. 263–281.

[10] C. Pflaum and A. Zhou, Numer. Math. **84**, 327 (1999).

[11] M. Born and K. Huang, *Dynamical theory of crystal lattices* (Oxford University Press, Oxford, UK, 1954).

[12] M. S. Daw and M. I. Baskes, Phys. Rev. B **29**, 6443 (1984).

[13] F. Ercolessi and J. B. Adams, Europhys. Lett. **26**, 583 (1994).

[14] A. M. Kraynik and D. A. Reinelt, Int. J. Multiphase Flow **18**, 1045 (1992).

[15] B. D. Todd and P. J. Davis, Phys. Rev. Lett. **81**, 1118 (1998).

[16] A. Baranyai and P. T. Cummings, J. Chem. Phys. **110**, 42 (1999).

[17] W. Ren and W. E, J. Comput. Phys. **204**, 1 (2005).