ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

MATHICSE

Mathematics Institute of Computational Science and Engineering

School of Basic Sciences - Section of Mathematics

MATHICSE Technical Report
Nr. 17.2012
April 2012

The heterogeneous multiscale method

A.Abdulle, W. E, B. Engquist, E. Vanden-Eijnden

http://mathicse.epfl.ch

Phone: +41 21 69 37648

Address:
EPFL - SB - MATHICSE (Bâtiment MA)
Station 8 - CH-1015 - Lausanne - Switzerland

Fax: +41 21 69 32545

The Heterogeneous Multiscale Method

Assyr Abdulle [1], Weinan E [2], Björn Engquist [3], Eric Vanden-Eijnden [4]

Abstract

The heterogeneous multiscale method (HMM), a general framework for designing multiscale algorithms, is reviewed. Emphasis is given to the error analysis that comes naturally with the framework. Examples of finite element and finite difference HMM are presented. Applications to dynamical systems and stochastic simulation algorithms with multiple time scales, spall fracture and heat conduction in microprocessors are discussed.

Contents

[1] ANMC, Mathematics Section, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, assyr.abdulle@epfl.ch

[2] Beijing International Center for Mathematical Research, Peking University, Beijing, China and Department of Mathematics and PACM, Princeton University, Princeton, USA, weinan@math.princeton.edu

[3] Department of Mathematics, University of Texas, Austin, USA, engquist@ices.utexas.edu

[4] Courant Institute of Mathematical Sciences, New York University, New York, USA, eve2@cims.nyu.edu

1 Introduction

The heterogeneous multiscale method (HMM) is a general framework for designing multiscale algorithms for a wide variety of applications [62, 63, 64]. The word "heterogenous" was used in order to emphasize the fact that the algorithm may involve macro and micro models of very different nature, for example, the micro model may come from molecular dynamics and the macro model may come from continuum theory. In fact, at a very rough level, HMM can be thought of as a way of blending together models of very heterogeneous nature.

Most problems that we encounter in nature have a multiscale character. The multiscale character can come in a variety of ways. Take, for example, problems from material science. It could be that the material property, such as conductivity, has a multiscale nature. This is the case for composites. It could also be that the material can be viewed at different levels of detail: as a continuous medium in which case one naturally applies the principles of continuum mechanics; or it can also be seen at the atomic scale, in which case one naturally applies various atomistic models coming either from molecular dynamics or quantum mechanics. Each viewpoint has its merits and drawbacks. Continuum models are quite efficient but sometimes their accuracy is inadequate, particularly when defects are involved. Atomic models are typically more accurate. But they are much less efficient. This situation is not limited to material science but is quite common in most all areas of science and engineering. One of the main motivation for multiscale modeling is to develop models that have an accuracy close to that of microscopic models and efficiency close to that of macroscopic models.

From the viewpoint of numerical algorithms, we are interested in extracting useful information from the microscopic model which in principle has the required accuracy. If we use a traditional viewppoint, then we would have to solve the microscopic model in full detail which is practically impossible for engineering applications. In terms of computational complexity, the best one can do with such an approach is to have linear scaling algorithms: The complexity scales no less than the number of microscopic degrees of freedom. However, in many cases, we are not interested in the full microscopic solution or we can not afford the cost of getting the microscopic solution. Instead, we are only interested in the behavior of some macroscopic variables and/or the microscopic behavior in very small parts of the system, for example, near defects. The question is: Can we develop much more efficient algorithms such as *sublinear* scaling algorithms, that would give us such information? To develop these new types of algorithms, we not only have to compress the operators, as has been done in multi-grid methods, but also the variables. We have to be contend with getting information about only part of the system variables. These types of algorithms can not be completely general: One has to explore special features of the problem in order to construct such algorithms.

From the viewpoint of analysis, many analytical techniques have been developed in order to derive simplified models. Examples include averaging methods, homogenization methods, matched asymptotics, WKB methods, geometric optics approximations, renormalization group methods, etc [61]. The principles of such techniques are quite general, but in practice, they only give us explicit analytical models in very limited situations. In other situations, it is natural to ask whether one can devise efficient computational techniques based on these principles. This is the case that we are interested in and was one of the main motivations for developing HMM.

This was the background over which HMM was proposed. Of course, prior to HMM, there were already many techniques of a similar spirit in many different fields. Well-known examples include:

- the Car-Parrinello molecular dynamics in which electronic structure models are used together with molecular dynamics in order to predict the dynamics of the nuclei [45];
- the quasicontinuum method in which atomistic models are used to analyze the mechanical deformation of crystalline solids [133];
- superparametrization models in which cloud-resolving models are used in order to capture the large scale tropical dynamics of the atomsphere [84, 140].

HMM was proposed as a general framework that can be used on a variety of problems. It was not the only attempt. Other notable examples include the extended multi-grid method and the equation-free approach [40, 93]. A common theme of these approaches is that the microscopic models are used throughout the computational process. These should be contrasted with techniques such as model reduction methods, wavelet-based homogenization, variational multiscale methods, etc, in which the microscale model is only used in the beginning of the computation to obtain compressed effective operators.

Inspite of the competing efforts, HMM was the only general attempt based on a top-down philosophy, which at the time, was not the most popular viewpoint. In fact, at the early days of multiscale modeling, most efforts were devoted to a bottom-up approach, seeking strategies that would give us the needed information by working only with the microscale model, without any prior information about the system at the macroscale. This certainly sounds very attractive and at a first sight, may seem to be the only right thing to do. In a way, a key insight in HMM was the recognition that the bottom-up approach was bound to have technical difficulites and will for some time be limited to rather simple applications. One can appreciate such difficulties by noticing the fact even if the effective macroscale model is explicitly available, designing stable and accurate numerical algorithms for such macroscale models is still a very non-trivial task. Important constraints such as conservation properties, upwinding, etc, have to be implemented in order to guarantee that the algorithms give rise to the right numerical solutions. Implementing such constraints at the level of microscale models in the absence of any explicit knowledge about the macroscale model seems to be next to impossible. Therefore compromises have to be made, i.e., for many problems, one has to make a guess about the form of the macroscale models to begin with. Fortunately, in many cases, one does have some prior knowledge about the macroscale behavior of the system under consideration, and this knowledge is often enough for us to make the adequate guess.

Since multiscale modeling is a vast subject, touching almost all aspects of modeling, we will not be able to do justice to all the work that has been done on this subject. Instead we will focus on HMM. For a general introduction to multiscale modeling, we refer to the monograph [61].

2 The HMM framework

2.1 The main components of HMM

We will use U to describe the set of macroscopic variables, and u the set of microscopic variables. They are related by:

$$U = Q(u), \tag{2.1}$$

Q is called the compression operator. Any operator that reconstructs u from U is called a reconstruction operator:

$$u = R(U). \tag{2.2}$$

For consistency, Q and R should satisfy the relation

$$Q(R(U)) = U. \tag{2.3}$$

In HMM, we assume that we have an incomplete macroscale model to begin with:

$$F(U, D) = 0, \tag{2.4}$$

Here D represents the missing part of the model. For example, if this is a model in continnum mechanics, then D might be the constitutive relation for the stress. If this is a model for molecular dynamics, then D might be the inter-atomic forces. If this is a model for heat conduction in composite materials, then D might be the macroscale effective conductivity tensor.

HMM proceeds by estimating the missing data on the fly using the microscale model, at each location where some missing data is needed. To do this, the microscale model has to be constrained so that its macro state is the same as the macro state we are interested in.

$$f(u, d(U)) = 0. \tag{2.5}$$

Here $d(U)$ represents the constraint for the microscale model. For example, if the microscale model is the NVT ensemble of molecular dynamics, d might be the average density, momentum and energy.

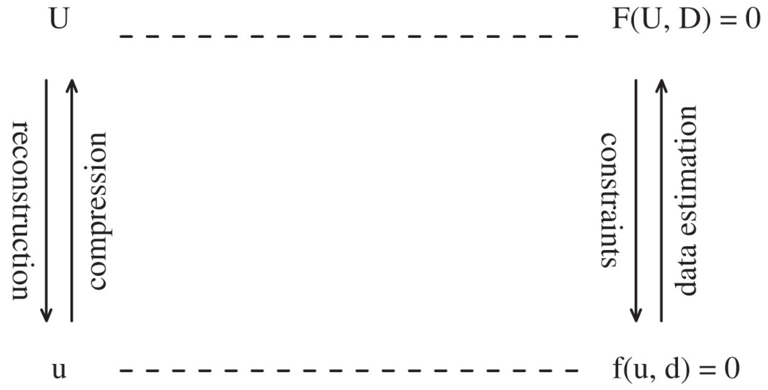


Figure 1: Schematics of the HMM framework

If we use H and h to denote the macro and micro numerical parameters such as mesh size, one can write HMM abstractly in the following form:

$$\begin{aligned} F_H(U_H, D_H(u_h)) &= 0, \\ f_h(u_h, d_h(U_H)) &= 0. \end{aligned} \tag{2.6}$$

In practical terms, the basic components of HMM are:

1. *A macroscopic solver.* Based on whatever knowledge that is available on the macroscale behavior of the system, we make an assumption about the form of the macroscale model, from which we select a suitable macroscale solver. For example, if we are dealing with a variational problem, we may use a finite element method as the macroscale solver.

2. *A procedure for estimating the missing macroscale data D using the microscale model.* This is typically done in two steps:

   (a) *Constrained microscale simulation*: At each point where some macroscale data is needed, perform a series of microscopic simulations which are constrained so that they are consistent with the local value of the macro variable.

   (b) *Data processing*: Use the results from the microscopic simulations to extract the macroscale data needed in the macroscale solver.

For dynamical problems, we can write down the HMM procedure formally as follows. At each macro time step:

1. Given the current state of the macro variables U^n, reinitialize the micro-variables:

$$u^{n,0} = RU^n. \tag{2.7}$$

2. Evolve the micro variables for some micro time steps:

$$u^{n,m+1} = \mathcal{S}_{\delta t}(u^{n,m}; U^n), \quad m = 0, \cdots, M-1; \tag{2.8}$$

3. Estimate D:

$$D^n = \mathcal{D}_M(u^{n,0}, u^{n,1}, \cdots, u^{n,M}); \tag{2.9}$$

4. Evolve the macro-variables for one macro time step using the macro-solver:

$$U^{n+1} = S_{\Delta t}(U^n; D^n). \tag{2.10}$$

Here R is some reconstruction operator which plays the same role as the interpolation or prolongation operators in the multi-grid method, $\mathcal{S}_{\delta t}$ is the micro solver, which also depends on U^n through the constraints, as indicated. $\mathcal{D}_M$ is some data processing operator which in general involves spatial/temporal/ensemble averaging. This is sometimes referred to as the data estimator. Finally $S_{\Delta t}$ is the macro solver.

For static problems, the procedure is very similar, particularly in the context of iterative algorithms: One just has to replace macro time step by macro iteration step.

For dynamic problems, there are two important time scales that we need to consider. The first, denoted by t_M, is the time scale for the dynamics of the macro-variables. The second, denoted by τ_ε, is the relaxation time for the microscopic model. We will need to distinguish two different cases. The first is when the two time scales are comparable, i.e. $\tau_\varepsilon \sim t_M$. In this case, from the viewpoint of numerical efficiency, there is not much room to play with as far as time scales are concerned. We just have to evolve the microscale model along with the macroscale model. The second case is when $\tau_\varepsilon \ll t_M$. This is the case we will focus on. The general guideline in this case is:

1. Choose Δt to accurately resolve the t_M time scale.
2. Choose M such that $M\delta t$ covers sufficiently the τ_ε time scale to allow equilibration to take place in the micro model.

2.2 Simple examples

Examples of such a setup include the following:

- ODEs with disparate time scales, where U is a complete set of slow variables, u is the full set of variables. In this case, the macroscale model could just be an ODE system for U, the missing data could just be the force. If we know that the macroscale system has additional structure, then we can take that into account when selecting the macro-solver. For example, if the macroscale system is a Hamiltonian system, then we can use a symplectic integrator as the macro-solver.

- Elliptic equations with multiscale coefficients such as the ones that arise in the modeling of the behavior of composite materials, where U is the averaged displacement field, u is the microscale displacement field. In this case, the macroscale model is still an elliptic equation, the missing data could be the coefficients in the macroscale model. If we use the finite element method as the macro-solver, then the missing data could just be the stiffness matrix, which has to be evaluated with the help of the full microscale model.

- Molecular dynamics models of complex fluids such as polymer fluids. Here U is the set of hydrodynamic variables, which in the simplest case would be the field of mass, momentum and energy densities. u is the set of microscopic variables, i.e. the position and momentum of all of the participating particles in a molecular dynamics model. The macroscale model might be the set of local conservation laws for U. The missing data might be the fluxes in these conservation laws.

- Microscopic pore-scale models of the pressure distribution in a porous medium. Here U is the macroscale pressure distribution. u is the set of variables in the pore-scale model, which could be a network-based model. The macroscale model is a Darcy-law type of model. The missing data are the coefficients in this model.

- Microscale models of the moving contact line. Here U is the set of macroscopic variables (velocity and pressure fields, position of the interface between the fluid phases). u is the set of variables in the microscopic description, say using molecular dynamics, of the contact line region. The macroscale model might just be the standard model in two-phase flows. The missing data might be the boundary conditions at the contact line.

Example 1. Stiff ODEs. Consider

$$\begin{cases} \dfrac{dx}{dt} = f(x,y), \\ \dfrac{dy}{dt} = -\dfrac{1}{\varepsilon}(y - \varphi(x)). \end{cases} \tag{2.11}$$

Here $U = x, u = (x,y)$. The macroscale should of course again be an ODE, which can be written as

$$\frac{dx}{dt} = F(x) = f(x, \varphi(x)). \tag{2.12}$$

The missing data is F which should be: $F(x) = f(x, \varphi(x))$ (we are pretending that this is hard to calculate analytically).

Let us choose the simplest solver for (2.12), the forward Euler method. HMM would then proceed as follows:

1. Initialize the micro-solver, e.g. $y^{n,0} = y^{n-1,M}$;

2. Apply the micro-solver for M micro steps:

$$y^{n,m+1} = y^{n,m} - \frac{\delta t}{\varepsilon}(y^{n,m} - \varphi(x^n)), \tag{2.13}$$

   $m = 0, 1, \cdots, M-1$.

3. Estimate $F(x)$:

$$F^n = f(x^n, y^{n,m}); \tag{2.14}$$

4. Apply the macro-solver:

$$x^{n+1} = x^n + \Delta t\, F^n. \tag{2.15}$$

Example 2. Stiff stochastic ODEs. Consider the stochastic ODE:

$$\begin{cases} \dfrac{dx}{dt} = f(x,y), \\ \dfrac{dy}{dt} = -\dfrac{1}{\varepsilon}(y - \varphi(x)) + \sqrt{\dfrac{2}{\varepsilon}}\dot{w}, \end{cases} \tag{2.16}$$

where $\dot{w}(t)$ is the standard white noise. The averaging theorems suggest that the effective macroscale equation should again be of the form of an ODE:

$$\frac{dx}{dt} = F(x). \tag{2.17}$$

HMM with forward Euler as the macro-solver proceeds as follows:

1. Initialize the micro-solver, e.g. $y^{n,0} = y^{n-1,M}$;

2. Apply the micro-solver for M micro steps:

$$y^{n,m+1} = y^{n,m} - \frac{\delta t}{\varepsilon}(y^{n,m} - \varphi(x^n)) + \sqrt{\frac{2\delta t}{\varepsilon}}\xi^{n,m}, \tag{2.18}$$

   $m = 0, 1, \cdots, M-1$. Here the $\{\xi^{n,m}\}$'s are independent normal random variables with mean 0 and variance 1;

3. Estimate $F(x)$:

$$F^n = \frac{1}{M}\sum_{m=1}^{M} f(x^n, y^{n,m}); \tag{2.19}$$

4. Apply the macro-solver:

$$x^{n+1} = x^n + \Delta t\, F^n. \tag{2.20}$$

Example 3. Elliptic PDE with with multiscale coefficients. Consider

$$-\nabla \cdot (a^\varepsilon(x)\nabla)u^\varepsilon(x) = f(x). \tag{2.21}$$

Abstract homogenization theory tells us that the macroscale model should be of the form

$$-\nabla \cdot (a^0(x)\nabla)U(x) = f(x) \tag{2.22}$$

The missing data are the coefficients $a^0(x)$. Naturally, as the macroscale solver, we choose standard finite element methods, e.g. the piecewise linear finite element method, over a coarse mesh. The data that need to be estimated is the stiffness matrix for the finite element method. If $a^0 = a^0(x)$ were known, we would simply follow standard practice and use numerical quadrature to compute the elements in the stiffness matrix. Since a^0 is not known, we set up a microscale simulation around each quadrature point in order to estimate the needed function value at that quadrature point. The details of this procedure will be discussed later.

Example 4. The parabolic homogenization problem Consider

$$\partial_t u^\varepsilon = \partial_x \cdot (a(x, \frac{x}{\varepsilon}, t)\partial_x u^\varepsilon), \tag{2.23}$$

where $a(x, y, t)$ is a smooth function and is periodic in y, say with period 1. The macroscale model is of the form

$$\partial_t U = \partial_x \cdot D, \tag{2.24}$$

$$D = \langle a(x, \frac{x}{\varepsilon}, t)\partial_x u^\varepsilon \rangle, \tag{2.25}$$

where $\langle\cdot\rangle$ means taking spatial averages.

We will choose a finite volume method as the macro-solver. then D needs to be evaluated at the cell boundaries [9]. We will make the assumption that the flux D depends on the local values of U and $\partial_x U$ only. Consequently for the micro model, we will impose the boundary condition that $u^\varepsilon(x,t) - Ax$ is periodic where $A = \partial_x U$ is evaluated at the location of interest.

Denote the micro-solver as:

$$u^{n+1} = \mathcal{S}_{\delta\tau,\delta x}(u^n; A). \tag{2.26}$$

Assuming that we have the numerical approximation $\{U_j^n\}$ (where $t^n = n\Delta t, U_j^n \sim U(n\Delta t, j\Delta x)$) at the n-th macro time step, we obtain the numerical approximation at the next macro time step by the following procedure:

1. For each j, let $A_j^n = (U_j^n - U_{j-1}^n)/\Delta x$.

2. Reinitialize the micro-solver, such that $u_j^0(x) - A_j^n x$ is periodic for each j.

3. Apply the micro-solver M steps:

$$u_j^{n,m+1} = \mathcal{S}_{\delta\tau,\delta x}(u_j^{n,m}; A_j^n),$$

with $m = 0, 1, \cdots, M-1$.

4. Compute

$$D_{j-1/2}^{n+1} = \langle a(x, \frac{x}{\varepsilon}, t^n)\partial_x u_j^{n,M}\rangle, \tag{2.27}$$

5. Evolve the macro-state variables using

$$U_j^{n+1} = U_j^n + \Delta t \frac{D_{j+1/2}^{n+1} - D_{j-1/2}^{n+1}}{\Delta x}, \tag{2.28}$$

Example 5. Incompressible polymeric fluid flow. Here U is the macroscale velocity field. The macroscale model should be of the form:

$$\rho_0(\partial_t U + (U \cdot \nabla)U) = \nabla \cdot \sigma,$$

$$\nabla \cdot U = 0.$$

These are simply statements of the conservation of momentum and mass, for a fluid of constant density ρ_0. The missing data is the stress field σ: $D = \sigma$.

Let us assume that the micro model is a molecular dynamics model for the particles that make up the fluid:

$$m_j \frac{d^2 y_j}{dt^2} = f_j, \quad j = 1, 2, \cdots, N \tag{2.29}$$

Here m_j, y_j are respectively the mass and position of the j-th particle, f_j is the force acting on the j-th particle. u is the set of variables in this model.

Given that the macroscale model is in the form of an incompressible flow equation, it is natural to select the projection method as the macro-solver [49]. In the implementation of the projection method, we will need the values of σ at the appropriate grid points. These are the data that need to be estimated. At this point, we have to make an assumption about what σ depends on, this enters in the constraints that we put on the microscale model. Let us assume that

$$\sigma = \sigma(\nabla U). \tag{2.30}$$

We will constrain the molecular dynamics in such a way that the average strain rate is given by the value of ∇U at the relevant grid point. In general, implementing such constraints is the most difficult step in HMM. For the present example, one possible strategy is discussed in [126].

From the results of molecular dynamics, we need to extract the values of the needed components of the stress. For this purpose, we need a formula that expresses stress in terms of the output of the molecular dynamics. This can be obtained by modifying the Irving-Kirkwood formula [90]. These details will be explained in Section 5. We refer to [126] for some numerical results obtained using this strategy.

2.3 Error analysis

The basic idea, as was explained in [63], is to compare the HMM solution with the solution of the selected macroscale solver for the effective macroscale model. Their difference is caused by an additional error in the HMM solution due to the error in the data estimation process. This new error term is called the HMM error, denoted by $e(\text{HMM})$. We will assume that both the HMM and the macro-solver for the effective macroscale model can be expressed in the form

$$U^{n+1}_{\text{HMM}} = U^n_{\text{HMM}} + \Delta t F^\varepsilon(U^n_{\text{HMM}}, U^{n-1}_{\text{HMM}}, \cdots), \tag{2.31}$$

$$\bar{U}^{n+1}_H = \bar{U}^n_H + \Delta t \bar{F}(\bar{U}^n_H, \bar{U}^{n-1}_H, \cdots). \tag{2.32}$$

Note that

$$||Qu^\varepsilon - U_{\text{HMM}}|| \le ||Qu^\varepsilon - \bar{U}|| + ||\bar{U}_H - \bar{U}|| + ||U_{\text{HMM}} - \bar{U}_H||, \tag{2.33}$$

where $\bar{U}$ is the solution of the macroscale model, $\bar{U}_H$ is the numerical solution to the effective macroscale model computed using (2.32), U_{hmm} is the HMM solution. The first term on the right hand side of (2.33) is due to the error of the effective model; the second term is due to the error in the macroscale solver; the third term is the HMM error, due to the error in the estimated data. Normally we expect that estimates of the following type hold:

$$||Qu^\varepsilon - \bar{U}|| \le C\varepsilon^\alpha, \tag{2.34}$$

$$||\bar{U} - \bar{U}_H|| \le C(\Delta t)^k, \tag{2.35}$$

where k is the order of accuracy of the macro-solver. In addition, define

$$e(\text{HMM}) = \max_U \|\bar{F}(U^n, U^{n-1}, \cdots) - F^\varepsilon(U^n, U^{n-1}, \cdots)\|. \tag{2.36}$$

Then under general stability conditions, one can show that [63]:

$$||U_{\text{HMM}} - \bar{U}_H|| \leq Ce(\text{HMM}), \tag{2.37}$$

for some constant C. Therefore, we have

$$||Qu^{\varepsilon} - U_{\text{HMM}}|| \leq C(\varepsilon^{\alpha} + (\Delta t)^k + e(\text{HMM})). \tag{2.38}$$

The key in getting concrete error estimates and thereby giving guidelines to designing multiscale methods lies in the estimation of e(HMM). This is specific to each problem. But typically, e(HMM) contains several contributions:

- the relaxation error, due to the influence of the transients;

- the sampling error, due to the fact empirical averages are used in order to estimate the ensemble average;

- the size effect, due to the fact that a computational domain of much smaller size is used to estimate quantities on domains of much bigger size.

Examples of these error components will be discussed in subsequent sections.

2.4 Difficulties with HMM

The most significant shortcoming of HMM is that it is based on a preconceived macroscale model. If the form of the macroscale model is chosen incorrectly, one cannot expect the resulting HMM procedure to produce accurate results. For example, if the effective macroscale model should be a stochastic ODE, but one makes the assumption that it is a deterministic ODE, then the stochastic component of the macroscale solution will not be captured correctly by an HMM based on such an assumption.

There is a important reason for starting with the macro-solver: Even for problems for which we do have a sufficiently accurate macroscale model, finding an effective numerical algorithm for that macroscale model may still be a significant task. Indeed this has been the focus of the computational mathematics community for more than 50 years. One example is Euler's equation in gas dynamics whose solutions typically contain shocks, i.e. discontinuities [100]. In this case, the numerical algorithms have to satisfy certain constraints in order to be able to solve Euler's equation accurately. Obviously this should also be a concern for multiscale methods.

For practical problems of interest, we often have accumulated some knowledge about what the macroscale model should be like. Such information can be used when making assumptions about the macroscale model used in HMM. In cases when one makes a wrong assumption, one can still argue that HMM produces an "optimal approximation" for the macroscale behavior of the solution in the class of the models considered. In this sense, HMM is a way of addressing the following question: What is the best one can do given the knowledge we have about the problem at all scales?

Another difficulty is that HMM, as was presented, is not very seamless for two reasons:

- It requires knowing a complete set of macroscale variables to begin with.
- It require converting back and forth between the macro- and micro-states of the system. This can become rather difficult in actual implementations, particularly when constructing discrete micro-states (needed for example in molecular dynamics) from continuous macroscale variables.

The seamless strategy proposed in [69] is intended to overcome the second difficulty. This will be explained next. The first difficulty is more significant. It can be dealt with in various situations with different levels of satisfaction, as we explain in the section 3.1.2.

2.5 Seamless HMM

To motivate the seamless algorithm, let us consider the stiff ODE example (2.11). If we want an efficient algorithm for capturing the behavior of x without resolving the detailed behavior of y, we can simply change the small parameter ε to a bigger value ε', the size of which is determined by the accuracy requirement

$$\begin{cases} \dfrac{dx}{dt} = f(x,y) \\ , \dfrac{dy}{dt} = -\dfrac{1}{\varepsilon'}(y - \varphi(x)). \end{cases} \tag{2.39}$$

This is then solved using standard ODE solvers. This idea of *boosting* the parameters is very simple, but it can also be quite useful, as in Car-Parrinello molecular dynamics [45].

We can look at this differently. Instead of changing the value of ε, we may change the clock for the microscale model, i.e. if we use $\tau = t\varepsilon/\varepsilon'$ in the second equation in (2.39), then (2.39) can be written as:

$$\begin{cases} \dfrac{dx}{dt} = f(x,y) \\ , \dfrac{dy}{d\tau} = -\dfrac{1}{\varepsilon}(y - \varphi(x)). \end{cases} \tag{2.40}$$

If we discretize this equation using standard ODE solvers but with different time step sizes for the first and second equations in (2.40), we obtain the following algorithm:

$$y^{n+1} = y^n - \frac{\delta\tau}{\varepsilon}(y^n - \varphi(x^n)), \tag{2.41}$$

$$D^{n+1} = y^{n+1}, \tag{2.42}$$

$$x^{n+1} = x^n + \Delta' t f(x^n, D^{n+1}). \tag{2.43}$$

Here $y^n \sim y(n\delta\tau)$ and $x^n \sim x(n\Delta' t)$. The value of $\delta\tau$ is the time step size we would use if we attempt to solve (2.40) accurately. If (2.40) were the molecular dynamics equations, then $\delta\tau$ would be the standard femtosecond time step size. $\Delta' t$ is the time step one would use for (2.39). It satifies

$$\frac{\Delta' t}{\varepsilon'} = \frac{\delta\tau}{\varepsilon}. \tag{2.44}$$

In general, $\Delta' t$ should be chosen such that one not only resolves the macro time scale, but also allows the micro state to sufficiently relax, i.e. to adjust to the changing macroscale environment. For example, if Δt is the time step size required for accurately resolving the macroscale dynamics and if τ_ε is the relaxation time of the microscopic model, then we should choose $\Delta' t = \Delta t/M$ where $M \gg \tau_\varepsilon/\delta\tau$.

The advantage of this second viewpoint is that it is quite general, and it does not require tuning parameters in the microscopic model. In a nutshell, the basic idea is as follows.

1. Run the (constrained) micro solver using its own time step $\delta\tau$.
2. Run the macro solver at a pace that is slower than a standard macro model: $\Delta' t = \Delta t/M$.
3. Exchange data between the micro- and macro-solvers at every step.

Because data is exchanged at every time step, there is no need to reconstruct new microscale states to initialize the microscale simulation. Intuitively, what one does is to force the microscale model to accommodate the changes in the macroscale environment (here the change in x) at a much faster pace. This is illustrated in Figure 2.

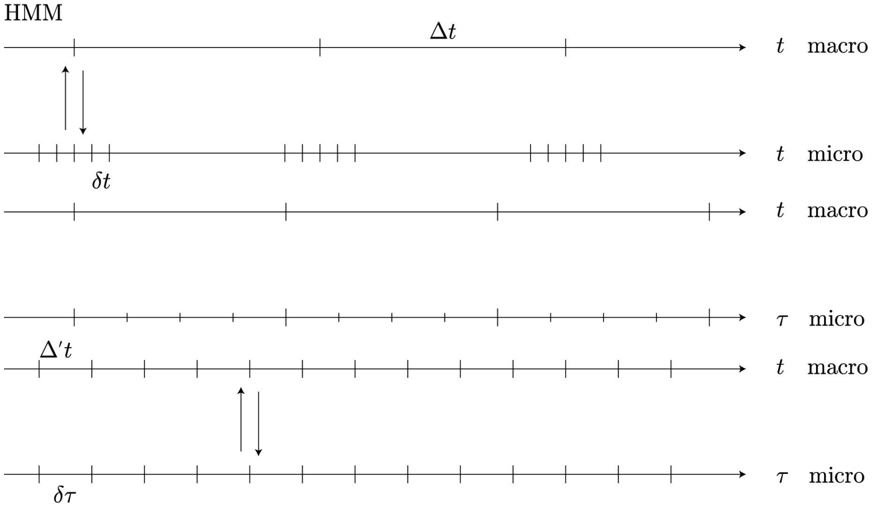


Figure 2: Illustration of HMM (upper panel) and the seamless algorithm (lower panel). Middle panel: rescaling the micro time scale.

The general form of the seamless algorithm can be written as follows:

1. Given the current state of the micro variables $u(\tau)$ and the macro variables $U(t)$, evolve the micro variables for one time step
$$u(\tau + \delta\tau) = \mathcal{S}_{\delta\tau}(u(\tau); U(t)); \tag{2.45}$$
2. Estimate D:
$$D = \mathcal{D}(u(\tau + \delta\tau)); \tag{2.46}$$
3. Evolve the macro variables
$$U(t + \Delta' t) = S_{\Delta' t}(U(t); D). \tag{2.47}$$

In this algorithm, we alternate between the macro- and micro-solvers, each running with its own time step (therefore the micro- and macro-solvers use different clocks). At every step, the needed macroscale data is estimated from the results of the micro-model (at that step) and is supplied to the macro-solver. The new values of the macro-state variables are then used to constrain the micro-solver.

From the consideration of time scales alone, the computational savings in the seamless algorithm come from the fact that effectively the system evolves on the time step $\Delta' t$. In the case when the time-scales are disparate, $\Delta' t$ can be much larger than $\delta\tau$. Therefore one can define the savings factor:
$$C_S = \frac{\Delta' t}{\delta\tau} = \frac{\Delta t}{M\delta\tau}. \tag{2.48}$$
For example, assume that the microscopic model is molecular dynamics, and the time step size is femtoseconds ($\delta\tau = 10^{-15}$ seconds), to simulate one second of physical time, one needs to run the simulation for 10^{15} steps. On the other hand, assume that the relaxation time is on the order of picoseconds (10^{-12} seconds) which is about 10^3 micro time steps, then $M = 10^5$ is a reasonable choice. Simulating one second of physical time using the seamless algorithm requires 10^5 steps. This is a saving by a factor 10^{10}. The price to be paid is that we no longer obtain accurate dynamic information at the level of microscopic detail: we can only hope to get the distribution of the fast variables accurately.

The price we pay is that at the level of the microscopic details, the information we get is not as accurate as the full microscopic simulation.

A slightly different interpretation of the seamless algorithms will be given in the next section.

Example: SDEs with multiple time scales Going back to the SDE (2.16), the seamless algorithm with forward Euler scheme is simply:
$$y^{n+1} = y^n - \frac{\delta\tau}{\varepsilon}(y^n - \phi(x^n)), + \sqrt{\frac{2\delta\tau}{\varepsilon}}\xi^n \tag{2.49}$$
$$x^{n+1} = x^n + \Delta' t\, f(x^n, y^{n+1}), \tag{2.50}$$

where the $\{\xi^n\}$'s are independent normal random variables with mean 0 and variance 1. Note that for HMM, we have $x^n \sim x(n\Delta t)$, but for the seamless algorithm, we have $x^n \sim x(n\Delta' t) = x(n\Delta t/M)$.

2.6 HMM for type A problems

There are two major types of multiscale problems: type A problems for which multiscale modeling is needed to resolve local defects or singularities and type B problems for which multiscale modeling is needed to supply constitutive relations at the macroscale [63]. HMM is most easily formulated for type B problems, and so far the formulations we have discussed are mostly for these problems. But HMM can also be used for type A problems, particularly in situations for which the relaxation time scale for the local structure near defects is much faster than the macroscopic dynamic time scale. In that case, the effect of the microscale structure near defects can be viewed as providing effective boundary conditions for the macroscale model at the defects.

One example is the moving contact line problem. There the macroscale model is simply the continuum two-phase flow model. The missing data are the boundary conditions at the contact line, which is then extracted from a molecular dynamics simulation of the local structure around the contact line.

3 ODEs and dynamical systems

3.1 General considerations

3.1.1 Standard HMM scheme

It is useful to revisit the ideas discussed in the last section in the context of more general stiff ODEs. Consider a generic multiscale system of the type

$$\begin{cases} \dfrac{dx^\epsilon}{dt} = f(x^\epsilon, y^\epsilon), \\ \dfrac{dy^\epsilon}{dt} = \dfrac{1}{\epsilon} g(x^\epsilon, y^\epsilon), \end{cases} \tag{3.1}$$

and let us assume that in the limit as $\epsilon \to 0$ the dynamics of this system can be described by the parametric distribution

$$\mu_{X(t)}(dy), \tag{3.2}$$

where $\mu_x(dy)$ is the invariant distribution of

$$\frac{dY}{dt} = g(x, Y), \tag{3.3}$$

with x viewed as a fixed parameter, and $X(t)$ in (3.2) solves

$$\frac{dX}{dt} = F(X(t)), \tag{3.4}$$

where

$$F(x) = \int f(x,y)\mu_x(dy). \tag{3.5}$$

In other words, we assume that, as $\epsilon \to 0$, x^ϵ converges pathwise to X whereas y^ϵ converges in law to Y, in a way that is captured by the distribution (3.2). This way of pharsing the limit theorem encompasses all the situations of interest to HMM: for example, (3.1) may be a set of stiff ODEs with a slow manifold structure, in which case $\mu_x(y)$ is atomic on this manifold (i.e. for any x, $\mu_x(dy) = \delta_{y(x)}(dx)$ for some $y(x)$); a set of SDEs, in which case $\mu_x(y)$ is typically broad; etc [61].

The standard HMM scheme starts with an integrator for (3.4). We will denote the numerical map associated with this integrator by η_t^F: here the superscript F makes explicit the dependency of this map on the function F defined in (3.5), which we do not know explicitly – the aim of aim of HMM is to approximate it on-the-fly. Note that η_t^F could depend on the derivatives of F besides this function itself, in which case the HMM scheme will have to estimate these as well: below we will denote collectively by $(F(x), DF(x), ...)$ the input about F needed in the map η_t^F. In order to obtain this input, let us denote by ζ_t^x the numerical map associated with (3.3), where now x enters as a parameter. Finally, given the output $(\zeta_{\delta t}^x(y_0), (\zeta_{\delta t}^x)^{o2}(y_0), \ldots, (\zeta_{\delta t}^x)^{oN}(y_0))$ of this map for N micro-steps of size δt starting from the initial condition $Y(0) = y_0$, let us denote by

$$(\tilde{F}(x), D\tilde{F}(x), ...) = \mathcal{F}\left(\zeta_{\delta t}^x(y_0), (\zeta_{\delta t}^x)^{o2}(y_0), \ldots, (\zeta_{\delta t}^x)^{oN}(y_0)\right), \tag{3.6}$$

the estimator for $(F(x), DF(x), ...)$. The HMM scheme then iterates upon the following procedure:

1. **Microscale solver:** Given $X_{m\Delta t}$, generate

$$(\zeta_{\delta t}^{X_{m\Delta t}}(y_0), (\zeta_{\delta t}^{X_{m\Delta t}})^{o2}(y_0), \ldots, (\zeta_{\delta t}^{X_{m\Delta t}})^{oN}(y_0)).$$

2. **Estimator:** Use these data to estimate

$$(\tilde{F}(X_{m\Delta t}), D\tilde{F}(X_{m\Delta t}), ...) = (\zeta_{\delta t}^{X_{m\Delta t}}(y_0), (\zeta_{\delta t}^{X_{m\Delta t}})^{o2}(y_0) \ldots, (\zeta_{\delta t}^{X_{m\Delta t}})^{oN}(y_0)).$$

3. **Macroscale solver:** Set

$$X_{(m+1)\Delta t} = \eta_{\Delta t}^{\tilde{F}}(X_{m\Delta t}).$$

Note that in the microsolver step, an initial data y_0 is needed. A natural choice is to use the last update of the previous iteration, i.e. at iteration $m+1$ take $y_0^{m+1} = (\zeta_{\delta t}^{X_{m\Delta t}})^{oN}(y_0^m)$.

3.1.2 Seamless HMM schemes

Let us now revisit the question of developing seamless HMM schemes that was already discussed in Section 2.5. To this end, consider again the generic multiscale system (3.1) and let us denote by ϕ_t a map that approximate numerically the solution of

$$\begin{cases} \dfrac{dX}{dt} = f(X,Y), \\ \dfrac{dY}{dt} = 0, \end{cases} \tag{3.7}$$

and by ψ_t a map that approximate numerically the solution of

$$\begin{cases} \dfrac{dX}{dt} = 0, \\ \dfrac{dY}{dt} = g(X,Y), \end{cases} \tag{3.8}$$

Note that the map ψ_t associated with (3.8) leaves the slow variables x^ϵ unchanged and could e.g. be used in the microsolver of HMM: in the notations of the last section $\psi_t(x,y) = (x, \xi_t^x(y))$. In terms of these maps, a vanilla splitting scheme for (3.1) with timestep δt would for example iterate upon

$$\phi_{\delta t} \circ \psi_{\delta t/\epsilon}. \tag{3.9}$$

Alternatively, we could take advantage of the fact that the map associated with (3.7) probably remains stable with larger time steps than that associated with (3.8) and use

$$\phi_{M\delta t} \circ \psi^{\circ M}_{\delta t/\epsilon}. \tag{3.10}$$

instead of M iterates of (3.9) to advance time by $M\delta t$.

It is interesting to revisit the HMM within these notations. Suppose that as estimator in the HMM we simply pick the last value of the fast variables after M steps of size δt, then take a macro-timesep of size Δt. The resulting scheme can be written compactly as

$$\phi_{\Delta t} \circ \psi^{\circ M}_{\delta t/\epsilon}. \tag{3.11}$$

This scheme has exactly the same cost as (3.10) but it advances the variables by Δt instead of $M\delta t$, thereby resulting in a efficiency gain if

$$\Delta t > M\delta t, \tag{3.12}$$

If (3.1) were a stiff system of ODEs, the estimator we just picked to derive (3.11) is the standard one, and so we already know why HMM works. Remarkably, it works in the general case. To see why, note that (3.11) can alternatively be interpreted as a time splitting scheme for (compare (3.1))

$$\begin{cases} \dfrac{dx^{\epsilon'}}{dt} = f(x^{\epsilon'}, y^{\epsilon'}), \\ \dfrac{dy^{\epsilon'}}{dt} = \dfrac{1}{\epsilon'} g(x^{\epsilon'}, y^{\epsilon'}), \end{cases} \tag{3.13}$$

where $\epsilon' = \epsilon\Delta t/M\delta t$. Indeed (3.11) is consistent with (3.13) in the limit

$$\Delta t \to 0, \quad \delta t \to 0 \quad \text{with} \quad \Delta t/M\delta t \to \alpha \in (1, \infty), \tag{3.14}$$

where we used the inequality (3.12) to restrict the range of α. Thus (3.13) is a version of the original system (3.1) in which ϵ has been boosted to the higher value $\epsilon' = \epsilon\alpha > \epsilon$, thereby making it less stiff. The key observation that this re-interpretation permits is this: As long as

$$\epsilon\Delta t \ll M\delta t \quad \Leftrightarrow \quad \epsilon' = \epsilon\alpha \ll 1, \tag{3.15}$$

the limit theorem we used to justify the HMM scheme guarantees that the solution to (3.13) remains close to that of the original system (3.1) in the sense that $x^{\epsilon'} \approx x^{\epsilon}$ pathwise, and the stationary distribution of $y^{\epsilon'}$ conditional on the current value of $x^{\epsilon'}$ approximates that of y^{ϵ} conditional on the current value of x^{ϵ}. This also implies that the numerical solution generated by (3.11) gives an approximation to the solution of the original system (3.1) which is accurate in the same sense. Thus, inequalities (3.12) and (3.15) give the conditions necessary for the scheme to provide an efficiency gain and remain accurate, respectively.

To make the link with the discussion in Section 2.5, note that (3.13) can also be interpreted as a system in which ϵ remains the same but the clock of the fast variables has been changed consistent with the time-rescaling $\tau = t\epsilon/\epsilon'$.

In practice, the values of Δt for which (3.15) remain valid may be too large for the scheme (3.11) to be stable, which seems to put an unnecessary cap on the efficiency gain achievable by the scheme. In these cases, letting $\Delta' t = \Delta t/M$, we can use the following scheme whose cost is close to that of (3.11) and also advances the variables by Δt at every iterate (i.e. provides the same efficiency boost) without suffering from the same stability limitations:

$$\left(\phi_{\Delta' t} \circ \psi_{\delta t/\epsilon}\right)^{\circ M}. \tag{3.16}$$

In terms of $\Delta' t$, (3.15) becomes

$$\Delta' t \ll \frac{\delta t}{\epsilon}, \tag{3.17}$$

and if the numerical error of the vanilla time-spliting scheme (3.9) is $C(\delta t/\epsilon)^a$ for some $C > 0$, then that of the scheme (3.16) is

$$C(\Delta' t/\epsilon')^a + \bar{C}\epsilon'^b = C(\delta t/\epsilon)^a + \bar{C}(\epsilon\Delta' t/\delta t)^b, \tag{3.18}$$

where $\bar{C} > 0$ and the exponent b depends on the rate of convergence of the solution of (3.1) towards its limit as $\epsilon \to 0$. Note that in order to be useful, the constant C in the error estimate (3.18) needs to be independent of ϵ. This property is not standard for time dependent problems, but is expected to hold if the map ψ_t for (3.8) is chosen appropriately and takes advantage of the fact that the fast variables reach a stable (statistical) steady state conditional on the slow variables being fixed. (3.16) was first introduced in [75] and further developed in

[69]. It is a seamless version of HMM in which the slow and the fast variables are evolved concurrently, and no reinitialization of the fast variables is required. This last feature comes especially handy in situations where the values that the fast variables can take are constrained by those of the slow variables, a situation that arise e.g. in molecular dynamics simulations (for applications in this context see e.g. [106, 107, 21]).

The scheme (3.16) is so simple, both in its implementation and interpretation, that it may seems like a trivilization of HMM – just boost the value of ϵ in the original system (3.1), adjust the time step accordingly, and be done! Yet, it opens the door to more sophisticated strategies. Indeed, since $\bar{C}(\epsilon\Delta' t/\delta t)^b$ is the only extra term appearing in the error estimate (3.18), and this term is controlled by the rate of convergence of (3.1) towards its limit as $\epsilon \to 0$, this suggests to modify (3.1) in such a way that the limit is unchanged but convergence to it is faster. How to do so concretely should be examined on a case by case basis.

The seamless version of HMM also opens the door to schemes in which the slow variables do not need to be identified beforehand, as was first noted in [137]. To see why, suppose that (3.1) is replaced by

$$\frac{dz^\epsilon}{dt} = F(z^\epsilon) + \frac{1}{\epsilon} G(z^\epsilon), \tag{3.19}$$

and let us assume that there exists some diffeomorphism that maps z^ϵ onto the variables (x^ϵ, y^ϵ) satisfying (3.1). More specifically, we require that $x^\epsilon = \theta(z^\epsilon)$ for some θ such that

$$D\theta(z)G(z) = 0, \tag{3.20}$$

so that

$$\frac{dx^\epsilon}{dt} = D\theta(z^\epsilon)F(z^\epsilon), \tag{3.21}$$

i.e. the variable $x^\epsilon = \theta(z^\epsilon)$ are indeed slow. We also require that no other (hidden) slow variable exists, i.e. $x^\epsilon = \theta(z^\epsilon)$ are those entering the limit theorem which the HMM scheme is built upon. Even if we do not know the explicit form of θ we can make use of its existence. To see how, denote by Φ_t the map that approximates the solution to

$$\frac{dZ}{dt} = F(Z), \tag{3.22}$$

and by Ψ_t the map that approximates the solution to

$$\frac{dZ}{dt} = G(Z), \tag{3.23}$$

In terms of these maps a vanilla time-splitting integrator for (3.19) would e.g. read (compare (3.9))

$$\Phi_{\delta t} \circ \Psi_{\delta t/\epsilon}, \tag{3.24}$$

or (compare (3.10))

$$\Phi_{M\delta t} \circ \Psi^{\circ M}_{\delta t/\epsilon}. \tag{3.25}$$

This last scheme updates the variables by $M\delta t$ at every iterate, and the discussion above suggests to replace it instead by (compare (3.11))

$$\Phi_{\Delta t} \circ \Psi_{\delta t/\epsilon}^{\circ M}, \tag{3.26}$$

or

$$\left(\Phi_{\Delta' t} \circ \Psi_{\delta t/\epsilon}\right)^{\circ M}, \tag{3.27}$$

where again $\Delta' t = \Delta t/M$. In the limit (3.14), both (3.26) and (3.27) are consistent with a version of (3.19) where ϵ has been boosted to a higher value $\epsilon' = \epsilon\alpha = \epsilon\Delta t/M\delta t$,

$$\frac{dz^{\epsilon'}}{dt} = F(z^{\epsilon'}) + \frac{1}{\epsilon'}G(z^{\epsilon'}), \tag{3.28}$$

which can again be used to explain why the scheme is accurate – basically (3.15) needs to hold and if the numerical error of the vanilla time-spliting scheme (3.24) is $C(\delta t/\epsilon)^a$ for some $C > 0$, then that of the scheme (3.27) is again given by (3.18).

The scheme (3.27) was first proposed in [137]. Later a very close variant of this scheme was proposed in [134] under the name of FLAVORS. Specifically, if we denote by $\chi_t^{1/\epsilon}$ the map that approximates numerically the solution of (3.19), and assume that we can control the size of ϵ in this map, then the scheme proposed in [134] is (in our notations)

$$\left(\chi^0_{\Delta' t - \delta t} \circ \chi^{1/\epsilon}_{\delta t}\right)^{\circ M}. \tag{3.29}$$

As can be seen, compared to (3.27), this scheme amounts to a slight modification of the way the time-spliting is implemented: (3.27) evolves the variables using first the term G/ϵ for a step of δt, then the term F for a step $\Delta' t$, whereas (3.29) evolves the variables using first the full $F + G/\epsilon$ for a step of δt, then the term F alone for a step $\Delta' t - \delta t$. In terms of accuracy, this modification is inconsequential – like (3.27), (3.29) is consistent with (3.28) in the limit (3.15). It also leads to no significant change in cost.

3.2 ODEs with oscillatory solutions

One advantage of the HMM framework is that sophisticated data estimation techiques can be used in the estimator step in order to improve its performance. This is most easily seen in the context of ODEs with highly oscillatory solutions. The examples discussed here also illustrate the subtleties involved in knowing or not knowing the slow variables. To be consistent with the existing literature, we will also use slightly different notations here.

Ordinary differential equations (ODEs) with highly oscillatory solutions are naturally challenging for numerical simulation. Several different numerical approaches have been suggested, each appropriate to some class of ODEs. For harmonic oscillatory problems, traditional numerical approaches attempt to either filter out or fit fast, ϵ-scale oscillations to some known functions in order to reduce the complexity, e.g. [77, 95, 131], or use some notion of Poincaré map to determine slow changes in the orbital structure [79, 122]. A general class of approaches aiming at Hamiltonian systems are geometric integration schemes that preserve a discrete version of certain invariance. We refer the readers to [85] and [99] for more extensive list of literature. These types of algorithms require $\mathcal{O}(\epsilon^{-1})$ number of time steps. In certain applications, special considerations are given to the expensive cost of evaluating non-local potential in large systems, see e.g. the impulse method and its derivatives [99]. For a recent review on numerical methods for highly oscillatory systems see [53].

Substantial progress has been made in the proposed direction of HMM style integration during the past few years. See e.g. [67, 69], [41, 42, 46, 130][27], and [134].

3.2.1 A kernel averaging theorem

In the present context, the appropriate HMM is best illustrated by Figure 3: the upper directed axes represent the grid used for U imposed by the macro-solver, and the lower axes represent the finer grids on which solutions of the microscopic equation are constructed by the micro-solver, with initial conditions determined from the values of U. The downward pointing arrows symbolize the determination of a consistent initial condition for the microscale equation from U at t_n; this process is referred to as reconstruction. The upward pointing arrows relate the evaluation of the effective equation from the time history of microscale variables computed in a short time interval η; this process is referred to as data estimation. Data estimation includes an evaluation of dU/dt as well as the values of U at different times. The process of evaluating a consistent value of Uat time t^* from the time history of the microscopic variables is referred to as compression. Typically, this evaluation is accomplished through filtering with a chosen compactly supported kernel K.

Hence we may present the above procedures algorithmically as follows:

1. Force estimation:

   (a) Reconstruction: at $T = t_n$, $U^n \mapsto x_n^0 = RU^n$.

   (b) Solve
   $$\frac{dx_n}{dt} = f_\epsilon(x_n, t), \quad x(t_n) = x_n^0,$$
   for $t \in [t_n, t_n + \eta]$.

   (c) Compression: $U^* = Q[x_n(\cdot)]$.

   (d) Estimate force: $\bar{f}(t_n + \delta t_*) \sim \tilde{f}(t_n + \delta t_*) = K * f_\epsilon(x_n)$.

   (e) (If $n = 0$, prepare (reset) the initial data by $\tilde{K}$:$U^0 = \tilde{K} * x_n$.)

2. Evolve the macro variables: $\{U^j\}_{j=0}^n, U^*, \bar{f}(t_n + \delta t_*) \longrightarrow U^{n+1}$, $T = t_{n+1}$.

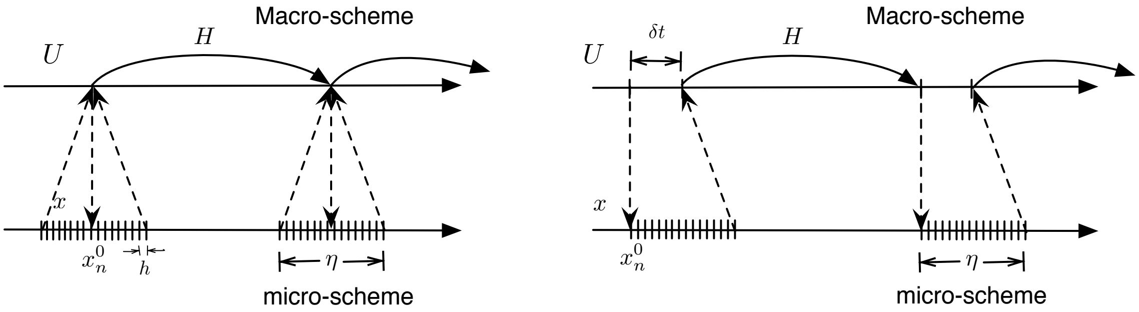


Figure 3: Two typical structures of the proposed multiscale algorithm. The structure on the left is for time reversible problems.

3. Repeat

In this subsection, the reconstruction operator R will be taken to be the identity operator; i.e. $RU^n = U^n$, and the compression $Q[x_n(\cdot)] = x_n(t_n + \delta t_*)$.

The two diagrams in Figure 3 differentiate the general structures of the HMM ODE solvers. There are two types of schemes being depicted there. In both schemes, the top axis represents the macro-grid on which we solve for the values of U (step 2), and under each grid node on the top axes, there corresponds a micro-grid over which step 1(b) takes place. The arrow pointing from each macro-grid point t_n down to a micro-grid denotes the action taken in step 1(a), while the arrows pointing from each micro-grid up towards the macro axis represent steps 1(c) and 1(d).

The schemes depicted in Figure 3 allows $\bar{f}$ and U_* (see steps 1 and 2) to be evaluated at the time location $t_n + \delta t_*$. The structure depicted on the left in Figure 3 requires $\delta t_* = 0$, and thus is suitable for implementing a linear multistep method as macro-solver. In this type of schemes, the effective force is usually evaluated at the left end of each fine scale evolution, therefore, a nonsymmetric kernel is used. In comparison, in the structure on the right, $\delta t_* = \eta/2$, and a one-step method such as Runge-Kutta can be used as a macro-solver. Due to this particular structure, a symmetric kernel can be used. In what follows, we use H and h for the discrete time steps used in the macro- and micro-grid respectively.

Force estimation by specialized kernels

In our formulation, we need to estimate the effective force locally at a point using the microscale data (Step 1(b)-(d)). Motivated by the analytic averaging techniques, see e.g. [61], we hypothesize that the effective force of a system of interest can be defined by

$$\bar{f}(t) = \lim_{\delta \longrightarrow 0} \left[\lim_{\epsilon \longrightarrow 0} \frac{1}{\delta} \int_t^{t+\delta} f_\epsilon(\tau) d\tau \right].$$

We assume that $\bar{f}$ is slowly varying in the sense that

$$\left|\frac{d^p}{dt^p}\bar{f}(t)\right| \leq C, \text{ for } 0 \leq p \leq s,$$

for some constant C independent of ϵ. Our goal in this section to show that time filtering using a kernel d $K_\eta^{p,q}$, defined below, with $\eta = \eta(\epsilon) \longrightarrow 0$ as $\epsilon \longrightarrow 0$ converges to $\bar{f}$:

$$K_\eta^{p,q} * f_\epsilon = K_\eta^{p,q} * \left(\bar{f} + g_\epsilon(t)\right) \longrightarrow \bar{f} \text{ as } \epsilon \longrightarrow 0. \tag{3.30}$$

In many situations, f_ϵ or g_ϵ assumes special forms such as $f_\epsilon(t) = f_\epsilon(t, t/\epsilon)$ that are periodic in the second variable. For example,

$$\frac{du}{dt} = f_\epsilon(u,t) = \frac{i}{\epsilon}\lambda u + \phi(t),$$

has solution

$$u(t) \;\; = \;\; e^{i\epsilon^{-1}\lambda t}(u_0 + \int_0^t e^{-i\epsilon^{-1}\lambda s}\phi(s)ds).$$

The force $f_\epsilon(t) = f_\epsilon(u,t) = \frac{i}{\epsilon}\lambda u + h(t)$ is of the form $f_\epsilon(t, t/\epsilon)$. In this case, we define

$$\bar{f}(t) = \int_0^1 f(t,s)ds,$$

and

$$g_\epsilon(t) = g(t, \frac{t}{\epsilon}) = f_\epsilon - \bar{f}(t).$$

We will use $\mathbb{K}^{p,q}$ to denote the kernel space discussed here. $K \in \mathbb{K}^{p,q}(I)$ if $K \in C_c^q(\mathbb{R})$ with $\text{supp}(K) = I$, and

$$\int_{\mathbb{R}} K(t)t^r dt = \begin{cases} 1, & r = 0; \\ 0, & 1 \leq r \leq p. \end{cases}$$

Furthermore, we will use $K_\eta(t)$ to denote the scaling of K:

$$K_\eta(t) := \frac{1}{\eta}K\left(\frac{t}{\eta}\right).$$

For convenience, we will also use $K^{p,q}$ to denote a function in $\mathbb{K}^{p,q}(I)$. We use $K \in \mathbb{K}^{p,q}([-1,0])$ if $\delta t^* = 0$ in step 1 of the algorithm described abovev, and $K \in \mathbb{K}^{p,q}([-1,1])$ if $\delta t^* = \eta/2$.

Theorem. Let $f_\epsilon(t) = f(t, t/\epsilon)$, where $f(t,s)$ 1-periodic in the second variable and $\partial^r f(t,s)/\partial t^r$ is continuous for $r = 0, \cdots, p-1$,For any $K \in \mathbb{K}^{p,q}$, there exists constants $C_{\bar{f},g_\epsilon}$and C_{g_ϵ}, independent of ϵ and η, such that

$$E = |K_\eta * f_\epsilon(t) - \bar{f}(t)| \leq C_{\bar{f},g_\epsilon}\eta^p + C_{g_\epsilon}\left(\frac{\epsilon}{\eta}\right)^q.$$

Furthermore, the error is minimized if η is chosen to scale with $\epsilon^{q/(p+q)}$.

3.2.2 Problem with hidden slow variables

We start by considering a class of system that has an explicit slow-fast grouping in the solution's components:

$$\begin{cases} \frac{dx}{dt} &= \frac{1}{\epsilon}f(x,y,t) + g(x,y,t), \\ \frac{dy}{dt} &= h(x,y,t). \end{cases} \tag{3.31}$$

Here the x components stay bounded but are highly oscillatory, and the y components are called the slow variables of the system, since their time derivatives are formally bounded. This would be the case for Hamiltonian systems written in the action and angle coordinates [61]. If $x(t)$ yields an invariant measure on some fixed manifold $\mathcal{M}$, then $y(t)$ can be consistently approximated in any constant time interval by an averaged equation

$$\frac{d\bar{y}}{dt} = \bar{h}(y,t) := \int_{\mathcal{M}} h(x,y,t)d\mu(x). \tag{3.32}$$

Such systems are widely studied to build multiscale scale methods. See e.g. [136][60]. In this case, it is reasonable to use y as the macroscopic variable; i.e. $U = \bar{y} \simeq y$, and $\mathcal{R}(U, D_R) = (x^*, y)$ where D_R gives the value $x^* \in \mathcal{M}$. For example, x^* may be taken from the x values in the previous microscale simulation. The compression $\mathcal{Q}$ may simply be $\mathcal{Q}(x,y) = y$.

However, if there are resonances among the oscillations, $x(t)$ is likely not to remain on a fixed invariant manifold [95], and more sophistication in the algorithm is needed. One can see the essence of this problem from the simple example,

$$\begin{cases} \frac{dx}{dt} &= \frac{i}{\epsilon}x + g(x), \;\; x(0) = 1, \\ \frac{dy}{dt} &= h(x), \;\; y(0) = y_0. \end{cases} \Longrightarrow \begin{cases} \frac{dw}{dt} &= e^{-\frac{i}{\epsilon}t}g(e^{\frac{i}{\epsilon}t}w), \;\; x(t) = e^{\frac{i}{\epsilon}t}w(t), \\ \frac{dy}{dt} &= h(e^{\frac{i}{\epsilon}t}w), \;\; y(0) = y_0. \end{cases}$$

Let us formally decompose $e^{-\frac{i}{\epsilon}t}g(e^{\frac{i}{\epsilon}t}w) = \bar{g}(w) + \alpha(e^{\frac{i}{\epsilon}t}, w)$, where $\bar{g}$ does not depend on any fast oscillations. If $\bar{g} \equiv 0$, $w(t)$ stays close to 1, due to strong self-averaging in α. Thus (3.32) corresponds to averaging h over the unit circle, and $d\mu$ is the arc-length element. Resonance in this system corresponds to the case where $\bar{g}$ is non-zero. Consequently, the averaging has to be performed with the correct measure

$$\frac{d\bar{y}}{dt} = \bar{h}(y,t) := \int_{\mathcal{M}(t)} h(x,y,t)d\mu(x;t).$$

For example, if $g(x) = x$, then $\bar{g}(w) = w$, and $\alpha \equiv 0$. Consequently, $\mathcal{M}(t)$ is a circle with radius equal to $w(t) = \exp(t)$. Without the knowledge of $w(t)$, it is impossible to define a consistent reconstruction operator $\mathcal{R}$, and consequently, it is impossible to build a convergent multiscale algorithm. In some literature, the issue caused by resonance is referred to as the system having hidden slow variables [75, 73, 24][134]. It is essential that a multiscale method computes accurately the effect of the hidden slow variables.

It is possible to design multiscale algorithms that compute the effective behavior of highly oscillatory dynamical systems by using slow variables, [25]. We continue our discussion using the previous example, but instead, we rewrite the equation for x as a system in $\mathbb{R}^2$:

$$\begin{cases} \frac{dx_1}{dt} &= \quad \frac{1}{\epsilon}x_2 + x_1, \\ \frac{dx_2}{dt} &= -\frac{1}{\epsilon}x_1 + x_2, \end{cases}$$

with initial conditions $(x_1(0), x_2(0)) = (0, 1)$. Thus $(x_1(t), x_2(t)) = (e^t \sin \epsilon^{-1}t, e^t \cos \epsilon^{-1}t)$.

Taking $I = x_1^2 + x_2^2$, we notice that I has a bounded derivative along the trajectory of the solution; i.e., $I' := (d/dt)I(x_1(t), x_2(t)) = 2I$ is independent of ϵ. For this particular example one can easily solve for I, $I(t) = I(0)e^{2t}$. In fact, the uniform bound on I' indicates the *slow* nature of $I(x_1(t), x_2(t))$ when compared to the fast oscillations in $(x_1(t), x_2(t))$. This type of characterization of the effective dynamics of highly oscillation systems are commonly used in the literature. In this example it was easy to find the slow quantity I. In general it is difficult and the proposed research aims at avoiding this difficulty. In classical mechanics, I is referred to as the action variable and correspondingly there is an angle variable whose gradient is τ.

We say that $\xi : \mathcal{D}_0 \subset \mathbb{R}^d \mapsto \mathbb{R}$ has a bounded derivative for $0 < \epsilon < \epsilon_0$ along $x^\epsilon(t)$ if

$$\sup_{x^\epsilon \in \mathcal{D}_0, \epsilon \in (0,\epsilon_0)} |\nabla \xi(x^\epsilon) \cdot \dot{x}^\epsilon| \le C. \tag{3.33}$$

Such functions are commonly referred to as slow variables of the system. See for example [94, 95, 96, 75, 78, 25, 26]. Other approaches to find slow variables includes, e.g. [29, 30].

Typically, one may expect that the trace of a slow variable along the dynamical system's solutions, $\xi \circ x^\epsilon$, converge to a smooth curve as $\epsilon \to 0$. We shall denote this limit as $\bar{\xi}(t; x_0)$. In designing multiscale algorithms for this type of problems, it is often convenient to aim at constructing accurate approximation of $\bar{\xi}$ by suitable filtering of the oscillations in $x^\epsilon(t)$.

3.2.3 Algorithm with hidden slow variables

Consider an ODE system of the general form

$$\frac{dx}{dt} = \frac{1}{\epsilon}f(x) + g(x), \qquad x(0) = x_0, \tag{3.34}$$

where $0 < \epsilon \le \epsilon_0$ is a small parameter that characterizes the separation of time scales in the problem. Let $\zeta(t)$ denote the solution of

$$\frac{d\zeta}{dt} = \bar{F}(\zeta) = \int_{S^1} F(\zeta, \sigma, \gamma = 0) d\sigma, \quad \zeta(0) = \xi(0). \tag{3.35}$$

The first step in our algorithm is to identify the slow variables $\xi(x)$. The slow variables can be determined analytically or numerically by identifying coefficients ina predetermined form. For detail, see [25]. Then, the ODE (3.34) is integrated using a two level algorithm; each level corresponds to the integration of (3.34) in a different time scale. The first is a Macro-solver, which integrates the averaged equation (3.35). The second level is a micro-solver that is invoked whenever the Macro-solver calls for it. Each time the micro-solver is invoked, it computes a short time solution of (3.34) using a suitable initial data.

1. Construction of slow variables:
   Find functions $\xi_1(x), \ldots, \xi_r(x)$ such that$|\nabla_x \xi \dot{\mathrm{x}}| \leq C_0$ and $\mathrm{rank}(\partial \xi / \partial x) = r$. See [25] for details.

2. Multiscale evolution:

   (a) Initial conditions: $x(0) = x_0$ and $n = 0$.

   (b) Force estimation:

      i. microscale solver: Solve (3.34) in $t \in [t_n, t_n + 2\eta]$ with initial conditions x_n.

      ii. averaging: approximate $\dot{\xi}(t_n + \eta)$ by
      $$\langle \dot{\xi} \rangle_\eta (t_n + \eta) = (K_\eta * \dot{\xi})(t_n + \eta) = (-\dot{K}_\eta * \xi)(t_n + \eta).$$

   (c) macroscale solver (forward Euler example):
   $x_{n+1} = x_n^{M/2} + H\delta x$, where δx is the least squares solution to the linear system
   $$\delta x \cdot \nabla \xi_i = \bar{F}_i(\xi) = \langle \dot{\xi}_i \rangle_\eta,$$
   for all $i = 1 \ldots r$.

   (d) $n = n + 1$. repeat steps (b) and (c) to time T.

Note that there is no need to actually change the original ODE (3.34) to a form with explicit use of the slow variables.

3.2.4 A Fermi-Pasta-Ulam model

The Fermi-Pasta-Ulam model [76] is a simple system of unit mass particles connected by springs. The springs alternate between stiff linear and soft non-linear ones. Recently, this model was considered by Hairer et. al. [85] as a benchmark problem for studying the long-time properties of numerical solutions to stiff ODEs using geometric integrators. The model is derived from the Hamiltonian

$$H(p,q) = \frac{1}{2} \sum_{i=1}^{2k} p_i^2 + \frac{1}{4} \epsilon^{-2} \sum_{i=1}^{k} (q_{2i} - q_{2i-1})^2 + \sum_{i=0}^{k} (q_{2i+1} - q_{2i})^4. \qquad (3.36)$$

The following linear change of variables is convenient, since it separates the elongations of the stiff springs and associated momentum:

$$x_i = \epsilon^{-1}(q_{2i-1} - q_{2i})/\sqrt{2}\ , \qquad v_i = (p_{2i-1} - p_{2i})/\sqrt{2}, \qquad (3.37)$$

and a second set of variables associated with the soft springs:

$$y_i = (q_{2i-1} + q_{2i})/\sqrt{2}\ , \qquad u_i = (p_{2i-1} + p_{2i})/\sqrt{2}, \tag{3.38}$$

Defining $y_0 = x_0 = y_{2k+1} = x_{2k+1} = 0$, the equations of motion become

$$\begin{cases} \dot{y}_i &= u_i, \\ \dot{x}_i &= \epsilon^{-1} v_i, \\ \dot{u}_i &= -(y_i - \epsilon x_i - y_{i-1} - \epsilon x_{i-1})^3 + (y_{i+1} - \epsilon x_{i+1} - y_i - \epsilon x_i)^3, \\ \dot{v}_i &= -\epsilon^{-1} x_i + (y_i - \epsilon x_i - y_{i-1} - \epsilon x_{i-1})^3 + (y_{i+1} - \epsilon x_{i+1} - y_i - \epsilon x_i)^3. \end{cases} \tag{3.39}$$

Typical initial conditions are $x_1 = y_1 = v_1 = u_1 = 1$ and zero otherwise, which means that initially $k-1$ of the stiff springs are at rest. The system admits $4k-1$ slow variables. First are all the degrees of freedom which are related to the soft springs: y_i and u_i, $i = 1 \dots k$. Second, the total energy (kinetic + potential) of the stiff springs, $I_i = x_i^2 + v_i^2$. Finally, the relative phases between the different stiff springs, $\phi_k = x_1 x_i + v_1 v_i$, $i = 1 \dots k-1$.

On the $O(1)$ time scale the system can be evolved using the HMM algorithm described above. Figure 4a depicts our results for systems with three stiff springs, $k = 3$, and with ten springs, $k = 10$, in Figure 5a. Simulation parameters for $k = 3$ are $\epsilon = 10^{-4}$, $h = \epsilon/15$, and $H = 0.02$ and $\eta = 15.4\epsilon$. For $k = 10$ we used $\epsilon = 10^{-4}$, $h = \epsilon/15$, and $H = 0.02$ and $\eta = 20.4\epsilon$. On the $O(\epsilon^{-1})$ time scale the dynamics become more interesting as the energies I_i begin to change [76, 85].

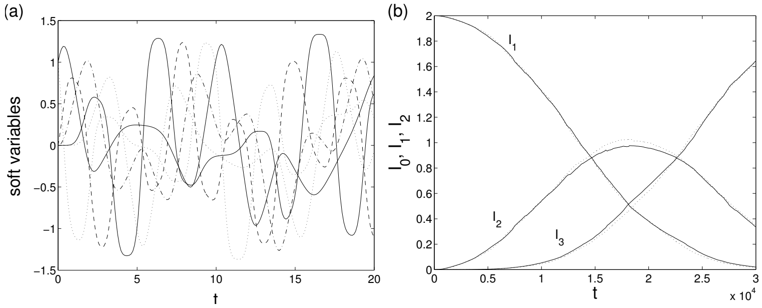


Figure 4: Comparison of the HMM approximation for the solution of the Fermi-Pasta-Ulam equations of motion (3.39) with 3 stiff springs, $k = 3$, to the one obtained using the Verlet method with step size of the order of ϵ. (a) soft variables on a $O(1)$ time scale and (b) I_1, I_2 and I_3 on a $O(\epsilon^{-1})$ scale. With the above parameters the HMM algorithm runs an order of magnitude faster than the Verlet one. The ratio between running times increases with smaller ϵ.

3.3 Stochastic simulation algorithms

Chemical kinetics can either be described using the rate equations or stochastic simulation algorithms. The former is suited for situations when the volume (or number) of the participating

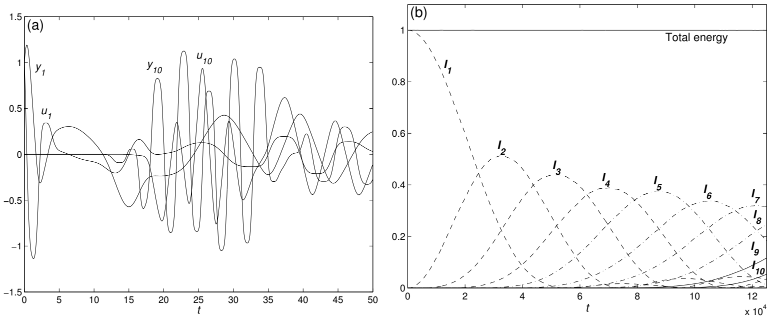


Figure 5: The HMM approximation for the solution of the Fermi-Pasta-Ulam equations of motion (3.39) with 10 stiff springs, $k = 10$. (a) y_1, u_1, y_{10} and u_{10} on a $O(1)$ time scale and (b) $I_1 \dots I_{10}$ on a $O(\epsilon^{-1})$ scale. The Verlet method takes too long to integrate.

species is large. The latter is necessary for situations when stochastic and/or discrete effects are important. This is particularly the case in biological applications. Since the rate of reactions often depend exponentially on physical parameters such as temperature, it is quite common that these rates are very disparate. Hence the rate equations are typically very stiff and the stochastic simulation algorithms often have a pronounced multiscale nature.

In this subsection, we discuss how HMM can be used to overcome the difficulties associated with the disparity of the rates in stochastic simulation algorithms. We begin with the general setup.

Let us assume that we have a total of N species of molecules, denoted by $S_1, \cdots, S_N$. The number of molecules of species S_k is denoted by x_k. The state vector is then given by $x = (x_1, \cdots, x_N)$. We will denote by $\mathcal{X}$ the state space where x lives in. Assume that there are M reaction channels, each described by its reaction rate and stoichiometric vector:

$$R_j = (a_j, \nu_j), \qquad R = \{R_1, \dots, R_M\}. \tag{3.40}$$

Given the state x, the occurrence of the reactions on an infinitesimal time interval dt is independent for different reactions and the probability for the reaction R_j to happen during this time interval is given by $a_j(x)dt$. After reaction R_j, the state of the system changes to $x + \nu_j$. In the chemistry and biology literature, this is often called the stochastic simulation algorithm (SSA) or Gilliespie algorithm [81], named after an algorithm that realizes this process exactly (see [81]).

Let $X(t)$ be the state variable at time t, and denote by $\mathbb{E}_x$ the expectation conditional on $X(0) = x$. Consider the observable $u(x,t) = \mathbb{E}_x f(X(t))$. $u(x,t)$ satisfies the following backward Kolmogorov equation:

$$\frac{\partial u(x,t)}{\partial t} = \sum_j a_j(x) \left(u(x+\nu_j, t) - u(x,t) \right) = (Lu)(x,t). \tag{3.41}$$

The operator L is the infinitesimal generator of the Markov process associated with the chemical kinetic system we are considering.

SSA proceeds as follows. Let

$$a(x) = \sum_{j=1}^{M_R} a_j(x). \tag{3.42}$$

Assume that the current time is t_n, and the system is at state X_n. We perform the following steps:

1. Generate independent random numbers r_1 and r_2 with uniform distribution on the unit interval $(0, 1]$. Let

$$\delta t_{n+1} = -\frac{\ln r_1}{a(X_n)}, \tag{3.43}$$

   and let k_{n+1} be the natural number such that

$$\frac{1}{a(X_n)} \sum_{j=0}^{k_{n+1}-1} a_j(X_n) < r_2 \leq \frac{1}{a(X_n)} \sum_{j=0}^{k_{n+1}} a_j(X_n), \tag{3.44}$$

   where $a_0 = 0$ by convention.

2. Update the time and the state of the system by

$$t_{n+1} = t_n + \delta t_{n+1}, \qquad X_{n+1} = X_n + \nu_{k_{n+1}}. \tag{3.45}$$

Then repeat.

In this algorithm, r_1 is used to update the clock and r_2 is used to select the particular reaction to be executed.

Assume now the rate functions have the following form

$$a(x) = \left(a^s(x), \frac{1}{\varepsilon} a^f(x) \right), \tag{3.46}$$

where $\varepsilon \ll 1$ represents the ratio of the fast and slow time scales of the system. The corresponding reactions and the associated stoichiometric vectors can be grouped accordingly:

$$R^s = \{(a^s, \nu^s)\}, \qquad R^f = \{(\frac{1}{\varepsilon} a^f, \nu^f)\}. \tag{3.47}$$

We call R^s the slow reactions and R^f the fast reactions. We have made the simplifying assumption that the rates are divided into two groups. Clearly the algorithm can be easily extended to situations when there are multiple groups. The more serious assumption here is that the groups of slow and fast reactions do not change in time. Some adaptive strategies have to be introduced in order to remove this restriction.

Our interest is on the dynamics of the slow processes, not the detailed dynamics of the fast processes. To this end, an effective system can be derived using standard averaging methods [97, 121]. For this purpose it is helpful to introduce an auxiliary process, called the *virtual* *fast process* [44]. This auxiliary process retains the fast reactions only, all slow reactions are turned off. Intuitively, each realization of the SSA consists of a sequence of realizations of the virtual fast process, punctuated by occasional firing of the slow reactions. Due to the time scale separation, with high probability, the virtual fast process has enough time to relax to equilibrium before another slow reaction takes place. Therefore the effective slow rates for the slow dynamics should simply be the original slow rates averaged over the equilibrium distributions associated with the virtual fast process.

Denote by μ_x the equilibrium distribution of the virtual fast process when it is initialized at x, and let

$$\tilde{a}_i^s(x) = \sum_{z \in \mathcal{X}} a_i^s(z) \mu_x(z). \tag{3.48}$$

It can be shown that the effective dynamics is governed by the set of reactions:

$$\bar{R} = (\bar{a}^s(x), \bar{\nu}^s). \tag{3.49}$$

The effective master equation is given by:

$$\frac{\partial u}{\partial t}(x,t) = \sum_{i=1}^{M_s} \tilde{a}_i^s(x) \left(u(x + \nu_i^s, t) - u(x,t) \right), \tag{3.50}$$

We will discuss an algorithm proposed in [65] which is a simple modification of the original SSA, by adding a nested structure according to the time scale of the rates. The process at each level of the time scale is simulated with an SSA with some effective rates. Results from simulations on fast time scales are used to compute the rates for the SSA at slower time scales.

Let t_n, X_n be the current time and state of the system respectively. The two-level nested SSA proceeds as follows:

1. **Microscale solver – the inner SSA:** Pick an integer N. Run N independent replicas of SSA with the fast reactions $R^f = \{(\varepsilon^{-1} a^f, \nu^f\})\}$ only, for a time interval of $T_0 + T_f$. During this calculation, compute the modified slow rates for $j = 1, \cdots, M_s$

$$\tilde{a}_j^s = \frac{1}{N} \sum_{k=1}^{N} \frac{1}{T_f} \int_{T_0}^{T_f + T_0} a_j^s(X_\tau^k) d\tau, \tag{3.51}$$

   where X_τ^k is the result of the k-th replica of this auxiliary virtual fast process at virtual time τ whose initial value is $X_{t=0}^k = X_n$, and T_0 is a parameter we choose in order to minimize the effect of the transients to the equilibrium in the virtual fast process.

2. **Macroscale sovler – the outer SSA:** Run one step of SSA for the modified slow reactions $\tilde{R}^s = (\tilde{a}^s, \nu^s)$ to generate (t_{n+1}, X_{n+1}) from (t_n, X_n).

Then repeat.

This algorithm is called the nested stochastic simulation algorithm (nested SSA). Unlike standard examples of HMM, here we do not need to know what the slow and fast variables are, in order to carry out the computation. The algorithm is formulated in terms of the original variables.

Convergence and efficiency of the nested SSA

The original SSA is an exact realization of the stochastic chemical kinetic system. The nested SSA, on the other hand, is an approximation. The errors in the nested SSA can be analyzed using the general strategy for analyzing HMM. The details can be found in [66].

Let f be a smooth function. Denote by $\tilde{X}_t$ the solution of the nested SSA. Consider the observable $v(x,t) = \mathbb{E}_x f(X_t)$ where the expectation is taken with respect to the randomness in the outer SSA only. Let $u(x,t)$ be the solution of the effective equation (3.50) with $u(x,0) = f(x)$. The following result is proved in [65, 66]:

Theorem. For any $T > 0$, there exist constants C and α independent of (N, T_0, T_f) such that,

$$\sup_{0\le t\le T, x\in\mathcal{X}} \mathbb{E}\,|v(x,t) - u(x,t)| \le C\left(\varepsilon + \frac{e^{-\alpha T_0/\varepsilon}}{1 + T_f/\varepsilon} + \frac{1}{\sqrt{N(1+T_f/\varepsilon)}}\right). \tag{3.52}$$

This result can be used to analyze the efficiency of the nested SSA. Given a chemical kinetic system with $R = \{(a_j, \nu_j)\}$, we assume that the total rate $a(x) = \sum a_j(x)$ does not fluctuate a lot in time: $a(x) \sim \mathcal{O}(\varepsilon^{-1})$. Given an error tolerance λ, we choose the parameters in the nested SSA such that each term in (3.52) is less than $\mathcal{O}(\lambda)$. One possible choice of the parameters is

$$T_0 = 0, \qquad N = 1 + \varepsilon^{-1} T_f = \frac{1}{\lambda}. \tag{3.53}$$

The total cost for the nested SSA over a time interval of $\mathcal{O}(1)$ is

$$\text{Cost} = \mathcal{O}(N(1 + T_0/\varepsilon + T_f/\varepsilon)) = \mathcal{O}\left(\frac{1}{\lambda^2}\right). \tag{3.54}$$

In comparison, the cost for the direct SSA is

$$\text{Cost} = \mathcal{O}\left(\frac{1}{\varepsilon}\right). \tag{3.55}$$

since the time step size is of order ε. When $\varepsilon \ll \lambda^2$, the nested SSA is much more efficient than the direct SSA.

Next we discuss the influence of the other numerical parameters on the efficiency. The parameter T_0 which plays the role of numerical relaxation time does not influence much the efficiency. Given the same error tolerance λ, for the last term in the error estimate (3.52) to be less than $\mathcal{O}(\lambda)$, we need to have

$$N\left(1 + \frac{\varepsilon}{T_f}\right) \ge \mathcal{O}\left(\frac{1}{\lambda^2}\right). \tag{3.56}$$

Therefore

$$\text{Cost} \geq \mathcal{O}(N\left(1+\frac{\varepsilon^{-1}}{T_f}\right)) = \mathcal{O}\left(\frac{1}{\lambda^2}\right). \tag{3.57}$$

which is the same as (3.55) regardless the value of T_0. The above argument also implies that the optimal cost for the nested SSA is $\mathcal{O}\left(\frac{1}{\lambda^2}\right)$ to achieve an error tolerance of λ.

Turning now to the effect of parameter N, the number of realizations for inner SSA, let us see what happens when we take $N = 1$. For the error estimate (3.52) to satisfy the same error tolerance λ, we have to choose

$$1 + \frac{\varepsilon}{T_f} = \frac{1}{\lambda^2}. \tag{3.58}$$

The cost of the nested SSA is given by

$$\text{Cost} = \mathcal{O}(N\left(1+\frac{\varepsilon}{T_f}\right)) = \mathcal{O}\left(\frac{1}{\lambda^2}\right), \tag{3.59}$$

which is the same as the cost if we use multiple realizations. This means that using multiple realizations in the inner SSA does not increase the efficiency of the overall scheme either. Obviously, using multiple realizations is advantageous for implementation on parallel computers.

Other versions of the nested SSA are discussed in [128, 127]. Although they appear to be quite different, it can be shown that they are essentially the same as the nested SSA discuss here.

A numerical example: A virus infection model

As a concrete example, we discuss a virus infection model studied in [65, 86]. The model was originally proposed in [132] as an example of the failure of modeling reacting networks with deterministic dynamics. The reactions considered in this model are listed in Table 1 with $M_R = 6$. The species that need to be modeled are *genome*, *struct*, *template* and *virus* ($N_s = 4$). *Genome* is the vehicle of the viral genetic information which can take the form of DNA, positive-strand RNA, negative-strand RNA, or some other variants. *Struct* represents the structural proteins that make up the virus. *Template* refers to the form of the nucleic acid that is transcribed and involved in catalytically synthesizing every viral component. The nucleotides and amino acids are assumed to be available at constant concentrations.

When *template* > 0, the production and degradation of *struct*, which are the third and fifth reactions marked with $*$ in table 1, are faster than the others. From the reaction rates, we can see that the ratio of time scales is about $\varepsilon = 10^{-3}$. In a system that consists of only the fast reactions, *struct* has an equilibrium distribution which is Poisson with parameter $\lambda = 500 \times \text{template}$ such that

$$\mathbb{P}_{template}(struct = n) = \frac{(500 \times template)^n}{n!} \exp(-500 \times template). \tag{3.60}$$

Notice that *struct* only shows up in the last slow reaction. The reduced dynamics in the form of the slow reactions ($a_{1,2,4,6}$) with the rates averaged with respect to the quasi-equilibrium of the fast reactions ($a_{3,5}$) can be given as a system with 4 reactions, shown in table 2. The initial

$$\begin{aligned}
nucleotides \quad &\xrightarrow{a_1=1.\times template} \quad genome \\
nucleotides \ + \ genome \quad &\xrightarrow{a_2=.025\times genome} \quad template \\
nucleotides \ + \ aminoacids \quad &\xrightarrow{a_3=1000\times template} \quad struct\ ^* \\
template \quad &\xrightarrow{a_4=.25\times template} \quad degraded \\
struct \quad &\xrightarrow{a_5=1.9985\times struct} \quad degraded/secreted\ ^* \\
genome \ + \ struct \quad &\xrightarrow{a_6=7.5d-6\times genome\times struct} \quad virus
\end{aligned}$$

Table 1: Reaction channels of the virus infection model

condition is chosen to be:

$$(struct,\ genome,\ template,\ virus) = (0, 0, 10, 0). \tag{3.61}$$

The mean value and the variance of *template* at time $T = 20$ are used as benchmark. A computation of these values by a direct SSA using $N_0 = 10^6$ realizations led to

$$\overline{template} = 3.7170 \pm 0.005, \qquad \mathtt{var}(template) = 4.9777 \pm 0.005. \tag{3.62}$$

For the nested SSA, we make a series of simulations in which we choose the size of the ensemble and the simulation time of the inner SSA according to

$$(N, T_0, T/\varepsilon) = (1, 0, 2^{2k}), \tag{3.63}$$

for different values of $k = 0, 1, 2, 3, \ldots$. The error estimate in (3.52) then implies that the error δ should decay with the rate:

$$\delta = O(2^{-k}). \tag{3.64}$$

which is consistent with the results in Figure 6. Table 3 gives the total CPU time and the obtained values of $\overline{template}$ and $\mathtt{var}(template)$ with the parameters of inner SSA chosen according to (3.63) and using $N_0 = 10^6$ realizations of the outer SSA (same as in the direct SSA).

$$\begin{aligned}
nucleotides \quad &\xrightarrow{a_1=1.\times template} \quad genome \\
nucleotides \ + \ genome \quad &\xrightarrow{a_2=.025\times genome} \quad template \\
template \quad &\xrightarrow{a_4=.25\times template} \quad degraded \\
genome \ + \ struct \quad &\xrightarrow{a_6=3.75d-3\times genome^2\times struct} \quad virus
\end{aligned}$$

Table 2: The reduced virus infection model

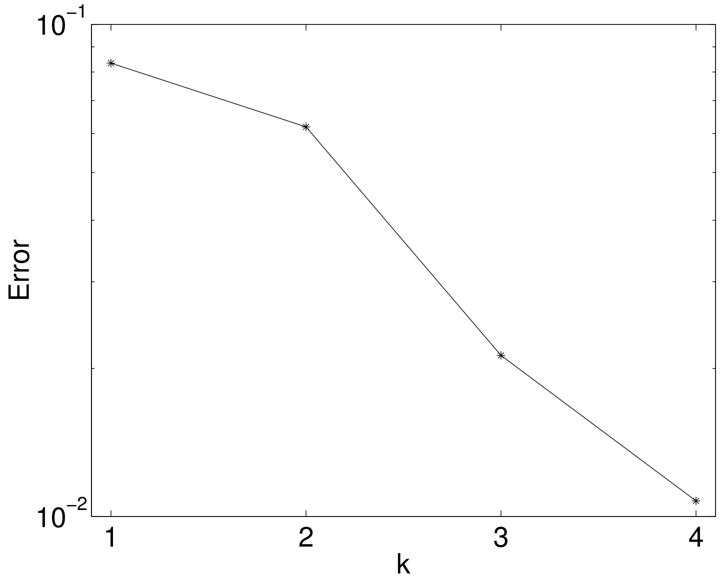


Figure 6: Relative errors of $\overline{template}$ using the nested SSA for the virus infection model (courtesy of Di Liu)

T_f/ε	1	4	16	64	"*exact*″
CPU	154.8	461.3	2068.2	9190.9	34806.4
$\overline{template}$	4.027	3.947	3.796	3.757	3.717 ± 0.005
$\texttt{var}(template)$	5.401	5.254	5.007	4.882	4.978 ± 0.005

Table 3: Efficiency of the nested SSA for the virus infection model

4 Finite element HMM

Finite element heterogeneous multiscale methods (FE-HMM) for homogenization problems first proposed in [63] have been developed in [2, 68, 3, 16] for elliptic problems, in [3] for elastic problems, in [112, 19] for parabolic problems, in [11] for wave problems and in [1, 87] for advection diffusion problems. Further developments for elliptic problems include coupling of FEs macroscopic method with spectral methods [10], with reduced basis methods [8], FE-HMM with special quadrature formulas [59], discontinuous Galerkin FE-HMM [4, 7].

Various numerical methods have been derived the past few years for multiscale partial differential equations, mostly for elliptic problems. We do not attempt here to review the literature on the subject and just give a few references. Numerical approaches for homogenization problems were pioneered by Babuška [31] for static problems and Engquist [71] for dynamic problems. For multiscale elliptic problems, an important early development is the generalized finite element method proposed by Babuška and Osborn [32], where the idea to adapt the finite element space to the particular fine-scale features of the problem is developed. Dorobantu, Engquist and Runborg [58],[72] proposed a method based on multi-resolution analysis, Neuss, Jäger and Wittum

[115] combined the multigrid method with homogenization in the coarsening process, Hou and co-workers proposed the multiscale finite element method (MsFEM) based on modified basis functions obtained from the finescale equations [89] (see [70] for a review), Babuška, Matache and Schwab developed the two-scale FEM [108],[109], Viet Ha Hoang and Schwab proposed the high dimensional FEM [88]. We also mention the huge literature in the structural mechanics and engineering communities concerned with micro-macro methods based on representative volume elements (RVEs). Such methods have been proposed for various type of problems, however, often without convergence analysis. We mention the methods of Terada, Kikuchi and co-workers [135], Miehe and co-workers [110] and Geers, Kouznetsova and Brekelmans [80].

4.1 General methodology

We start by explaining the methodology of the FE-HMM. We consider a general multiscale problem of the form

$$L^{\varepsilon}(u^{\varepsilon}, a^{\varepsilon}) = f \quad \text{in } \Omega,$$

where L^{ε} is a differential operator, a^{ε} denotes the data of the problem, Ω is an open bounded subset of $\mathbb{R}^d$, $f : \Omega \to \mathbb{R}$ is given and $u^{\varepsilon} : \bar{\Omega} \to \mathbb{R}$ is the solution of the above problem for which appropriate boundary conditions are specified. To emphasize the multiscale nature of the data of the above problem, we put a superscript ε (representing the typical size of a small scale in the considered problem) on a, L and u. Here for simplicity, in view of numerical discretization, we assume that $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ is a polygonal domain. The effective problem, assumed to exist, is of the form

$$L^{0}(u^{0}, a^{0}) = f \quad \text{in } \Omega,$$

with $u^0 : \bar{\Omega} \to \mathbb{R}$, the effective solution. One can think of u^0 as being the limit (in an appropriate sense) of the solution u^{ε} as $\varepsilon \to 0$. The weak solutions of any of the above problems are assumed to belong to an appropriate Sobolev space, denoted by $H(\Omega)$. The FE-HMM can be summarized by the following steps.

- Macroscale solver.
  - Macro triangulation: define a macroscopic partition of Ω, $\bigcup_{K \in \mathcal{T}_H} K = \Omega$ and a macroscopic finite dimensional subspace of $H(\Omega)$ denoted by $S(\Omega, \mathcal{T}_H)$.
  - Macro method: define $u_H \in S(\Omega, \mathcal{T}_H)$ solution of $L_H(u_H) = f$, where L_H is an unknown approximation of the effective differential operator L^0.
- Microscale solver.
  - Constrained micro simulations: for every $K \in \mathcal{T}_H$ consider a suitable *quadrature formula* $\{x_{K_j}, \omega_{K_j}\}_{j=1}^{J}$, and *sampling domains* $K_{\delta_j} = x_{K_j} + \delta I,\ I = (-1/2, 1/2)^d$ $(\delta \geq \varepsilon)$. Define a micro triangulation $\bigcup_{T \in \mathcal{T}_h} T = K_{\delta_j}$ and a micro finite dimensional space

$S(K_{\delta_j},\mathcal{T}_h)$. Compute micro solutions $u_h \in S(K_{\delta_j},\mathcal{T}_h)$ constrained by the macro variable using the original fine scale operator.

- Data processing: recover locally $L_H(u_H)|_K$ by suitable average of u_h in K_{δ_j}.

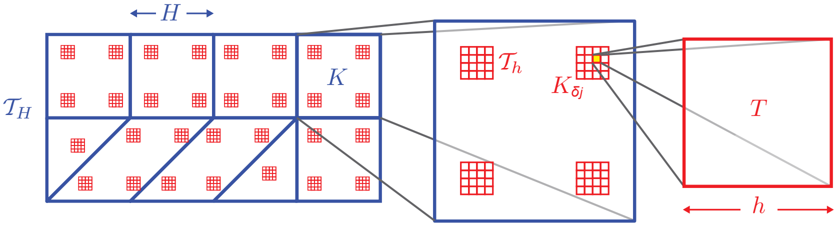


Figure 7: Sketch of the FE-HMM algorithm: macro triangulation $\mathcal{T}_H$ (quadrilateral and simplicial elements) with sampling domains within the macro elements. A zoom on the sampling domains shows the micro triangulation $\mathcal{T}_h$ chosen here to be quadrilaterals.

We notice that the method described here could be used to solve problems with a right-hand side also depending on ε (e.g., f^ε) with minor changes (see [16]).

4.2 Model problem

To describe the method, we consider $L^\varepsilon(u^\varepsilon, a^\varepsilon) = -\nabla \cdot (a^\varepsilon \nabla u^\varepsilon)$, i.e., the linear elliptic homogenization problem already seen in Section 2

$$-\nabla \cdot (a^\varepsilon \nabla u^\varepsilon) = f \text{ in } \Omega, \quad u^\varepsilon = 0 \text{ on } \partial\Omega, \tag{4.1}$$

where the family of tensors $a^\varepsilon(x) \in (L^\infty(\Omega))^{d\times d}$ (indexed by ε) are assumed to be uniformly elliptic and bounded, i.e.,

$$\exists \lambda, \Lambda > 0 \text{ such that } \lambda|\xi|^2 \leq a^\varepsilon(x)\xi\cdot\xi,\ |a^\varepsilon(x)\xi| \leq \Lambda|\xi|,\ \forall\xi \in \mathbb{R}^d, \forall\varepsilon,\ \text{a.e. } x \in \Omega, \tag{4.2}$$

where ε is a microscopic scale that characterizes the multiscale nature of the tensor $a^\varepsilon(x)$. Here we take zero Dirichlet boundary conditions for simplicity. [5] We also assume that $f \in L^2(\Omega)$ ($f \in H^{-1}(\Omega)$ would also be possible). By the Lax-Milgram theorem, owing to the uniform ellipticity and boundedness of the tensor a^ε, the weak form of (4.1) possesses for each ε a unique solution and we can thus consider a family of solutions $\{u^\varepsilon\}$ (indexed by ε) which are bounded in $H^1_0(\Omega)$ by the same constant.

Homogenization results. The goal of homogenization theory is to find an "averaged equation" corresponding to (4.1). Without further assumptions on the heterogeneities of the tensor $a^\varepsilon(x)$ it is possible to show, using H convergence theory [113] (or G convergence theory in the

[5] The algorithm and results stated in this chapter are valid for other boundary conditions (non-zero Dirichlet, Neumann, mixed, etc.) with obvious changes.

symmetric case [56]), that there exists a tensor $a^0(x)$ (again uniformly elliptic and bounded) and a subsequence of $\{u^\varepsilon\}$ which weakly converges in $H_0^1(\Omega)$ to a function $u^0 \in H_0^1(\Omega)$, solution of the (effective) equation

$$-\nabla \cdot \left(a^0 \nabla u^0\right) = f \text{ in } \Omega, \quad u^0 = 0 \text{ on } \partial\Omega. \tag{4.3}$$

Going back to the general description of Section 4.1, we see that here, $L^0(u^0, a^0) = -\nabla \cdot (a^0 \nabla u^0)$.

If $a^\varepsilon(x)$ has a more specific structure, for example if $a^\varepsilon(x) = a(x, x/\varepsilon)$ and is periodic in its second argument, then classical results show that the whole sequence $\{u^\varepsilon\}$ weakly converges to $u^0 \in H_0^1(\Omega)$ and the homogenized tensor $a^0(x)$ at $x \in \Omega$ can be characterized as the average of the solutions of "cell problems" (i.e., d boundary value problems on a domain of size ε^d involving the finescale tensor $a^\varepsilon(x) = a(x, x/\varepsilon)$). We refer for example to [35, 92, 51] for details. Notice that even in this fortunate case, one still has in general an infinite number of cell problems to solve, due to the slow variation in the tensor. Classical numerical approaches consist in pre-computing an approximation of the effective tensor $a^0(x_i)$ at predefined sampling points $x_i \in \Omega,\ i = 1, \ldots, p$ and using a standard FEM based on these pre-computed data. Although straightforward, this sequential strategy does not allow for a natural way to derive a priori or a posteriori error control of the overall discretization (as it depends on the accuracy of the numerically pre-computed homogenized tensors). This procedure does also not give a straightforward discretization for non-periodic or nonlinear problems, and makes it difficult to switch locally to a finescale approximation (desirable in some applications).

4.3 FE-HMM: algorithm

In this section we give a detailed description of the FE-HMM. We restrict ourselves to the model problem (4.1). In Section 4.3.3, generalizations to more involved problems (e.g., nonlinear problems) will be described.

4.3.1 Macroscale solver

We denote by $\mathcal{T}_H$ a family of (macro) partitions[6] of Ω in simplicial or rectangular elements K. The diameter of an element $K \in \mathcal{T}_H$ is denoted by H_K and we set $H = \max_{K \in \mathcal{T}_H} H_K$. We then consider a macro FE space

$$S_0^\ell(\Omega, \mathcal{T}_H) = \{v_H \in H_0^1(\Omega);\ v_H|_K \in \mathcal{R}^\ell(K),\ \forall K \in \mathcal{T}_H\}, \tag{4.4}$$

where $\mathcal{R}^\ell(K)$ is the space $\mathcal{P}^\ell(K)$ of polynomials on K of total degree at most ℓ if K is a simplicial FE, or the space $\mathcal{Q}^\ell(K)$ of polynomials on K of degree at most ℓ in each variable if K is a rectangular FE. For every K we consider the quadrature formula $\{x_{K_j}, \omega_{K_j}\}_{j=1}^J$ (described further in Section 4.3.2), and sampling domains K_{δ_j}, defined as $K_{\delta_j} = x_{K_j} + \delta I,\ I = (-1/2, 1/2)^d$ $(\delta \geq$

[6] By macro partition we mean that $H \gg \varepsilon$ is allowed.

ε).

Multiscale method. Find $u_H \in S_0^\ell(\Omega, \mathcal{T}_H)$ such that

$$B_H(u_H, v_H) = \int_\Omega f v_H dx, \quad \forall v_H \in S_0^\ell(\Omega, \mathcal{T}_H), \tag{4.5}$$

where for arbitrary $v_H, w_H \in S_0^\ell(\Omega, \mathcal{T}_H)$ the bilinear form $B_H(\cdot, \cdot)$ is defined by

$$B_H(v_H, w_H) := \sum_{K \in \mathcal{T}_H} \sum_{j=1}^{J} \frac{\omega_{K_j}}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x) \nabla v_{h,K_j} \cdot \nabla w_{h,K_j} dx, \tag{4.6}$$

where $v_{h,K_j} = R_{K_j}(v_H)$, $w_{h,K_j} = R_{K_j}(w_H)$ and $R_{K_j} = R|_{K_j}$ is the reconstruction operator (in the terminology of Section 2) restricted to the sampling domain K_{δ_j}. In the notation of Section 4.1, we have $L_H : \ S_0^\ell(\Omega, \mathcal{T}_H) \longrightarrow L^2(\Omega)$, where

$$(L_H(v_H), w_H) = B_H(v_H, w_H),$$

and $(\cdot, \cdot)$ is the L^2 inner product. In order to assemble the bilinear form $B_H(v_H, w_H)$ one needs to compute the functions $R_{K_j}(v_H), R_{K_j}(w_H)$ as described in the next section.

4.3.2 Microscale solver

The location of the sampling domains K_{δ_j} and the choice of the weights ω_{K_j} $(j = 1, \ldots, J)$ in the definition of $B_H(\cdot, \cdot)$ rely on the definition of appropriate quadrature formulas (QF). We consider $\hat{K}$ the reference element and for every element of the triangulation the mapping F_K (a $\mathcal{C}^1$-diffeomorphism) such that $K = F_K(\hat{K})$. For every K we consider the quadrature formula $x_{K_j} = F_K(\hat{x}_j)$, $\omega_{K_j} = \hat{\omega}_j |\det(\partial F_K)|$, $j = 1, \ldots, J$, where $\{\hat{x}_j, \hat{\omega}_j\}_{j=1}^J$ is a quadrature formula on $\hat{K}$. We then consider a (micro) partition $\mathcal{T}_h$ of each sampling domain K_{δ_j} in simplicial or rectangular elements T and define a micro FE space

$$S^q(K_{\delta_j}, \mathcal{T}_h) = \{z_h \in W(K_{\delta_j}); \ z_h|_T \in \mathcal{R}^q(T), \ \forall \ T \in \mathcal{T}_h\}, \tag{4.7}$$

where $h = \max_{T \in \mathcal{T}_h} h_T$ (h_T is the diameter of the element T) and $W(K_{\delta_j})$ is a given Sobolev space.

Coupling. Various spaces $W(K_{\delta_j})$ can be chosen for the micro numerical method. For example

$$W(K_{\delta_j}) = W^1_{per}(K_{\delta_j}) = \{z \in H^1_{per}(K_{\delta_j}); \int_{K_{\delta_j}} z dx = 0\}, \tag{4.8}$$

for a periodic coupling, or

$$W(K_{\delta_j}) = H^1_0(K_{\delta_j}), \tag{4.9}$$

for a coupling through Dirichlet boundary conditions. Other coupling conditions, constraining the averaged gradient of a microscale solution can be used. We refer to [143] for further discussion.

Micro method. For each $v_H \in S_0^\ell(\Omega, \mathcal{T}_H)$ we define a microfunction v_{h,K_j}, satisfying $(v_{h,K_j} - v_{H,lin}) \in S^q(K_{\delta_j}, \mathcal{T}_h)$ and solution of

$$\int_{K_{\delta_j}} a^\varepsilon(x) \nabla v_{h,K_j} \cdot \nabla z_h dx = 0, \quad \forall z_h \in S^q(K_{\delta_j}, \mathcal{T}_h), \tag{4.10}$$

where $v_{H,lin}|_{K_{\delta_j}} = v_H(x_{K_j}) + (x - x_{K_j}) \cdot \nabla v_H(x_{K_j})$ is the linearisation of v_H at the quadrature point x_{K_j}. As explained in Section 4.4.1, the bilinear form (4.6) can be rewritten as

$$B_H(v_H, w_H) \quad = \quad \sum_{K \in \mathcal{T}_H} \sum_{j=1}^{J} \omega_{K_j} a_K^0(x_{K_j}) \nabla v_H(x_{K_j}) \cdot \nabla w_H(x_{K_j}),$$

where $a_K^0(x_{K_j})$ is the macroscale data (effective tensor) at the quadrature point x_{K_j} recovered by the microscale simulations. This is the data processing step mentioned in Section 4.1 that is implicitly computed by defining the bilinear form through the quantities $\frac{1}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x) \nabla v_{h,K_j} \cdot \nabla w_{h,K_j} dx$.

Remark 1. *Requirements on the quadrature formula are needed to ensure that the optimal convergence rates for elliptic FEM are obtained when using numerical quadrature. We make the following classical assumptions on the quadrature formula $\{\hat{x}_j, \hat{\omega}_j\}_{j=1}^J$ on the reference element $\hat{K}$ (see [50])*

(Q1) *$\hat{\omega}_j > 0,\ j = 1, \ldots, J, \quad \sum_{j=1}^J \hat{\omega}_j |\nabla \hat{p}(\hat{x}_j)|^2 \geq \hat{\lambda} \|\nabla \hat{p}\|^2_{L^2(\hat{K})},\ \forall \hat{p}(\hat{x}) \in \mathcal{R}^\ell(\hat{K}),\ \hat{\lambda} > 0;$*

(Q2) *$\int_{\hat{K}} \hat{p}(\hat{x}) d\hat{x} = \sum_{j=1}^J \hat{\omega}_j \hat{p}(\hat{x}_j),\ \forall \hat{p}(\hat{x}) \in \mathcal{R}^\sigma(\hat{K})$, where $\sigma = \max(2\ell - 2, \ell)$ if $\hat{K}$ is a simplicial FE, or $\sigma = \max(2\ell - 1, \ell + 1)$ if $\hat{K}$ is a rectangular FE.*

4.3.3 Implementation and numerical illustration

The macro-micro methodology described in Section 4.3 allows for an easy implementation and to design a code whose structure follows the classical finite element implementation at the macro level. In particular, elementwise assembly for each macro element K can be computed to find the additive contribution from (4.6) to the macroscopic stiffness matrix. In [13] a short (less than 200 lines) and flexible MATLAB implementation has been proposed, capable of handling 2 and 3 dimensional elliptic and parabolic problems. All the numerical experiments presented in this sections have been made with the code presented in [13] or a variation of it (e.g., [17]). These codes are publicly available at `http://anmc.epfl.ch`. Additional numerical experiments can be found in [5, 111].

Macro-micro meshes refinement. Some care is needed to set appropriately the macroscopic mesh $\mathcal{T}_H$ (a triangulation of the physical domain Ω) and the microscopic mesh $\mathcal{T}_h$ (a triangulation of the sampling domain K_{δ_j}). Taking N_{mic} elements in each space dimension for the discretization of the sampling domain K_{δ_j}, we have we have $h = \delta/N_{mic}$ and thus $\hat{h} := h/\varepsilon = (\delta/\varepsilon) \cdot (1/N_{mic})$. Since δ scales with ε, typically $\delta = C\varepsilon$ (where C is a constant of moderate size), we have $\hat{h} = C/N_{mic}$. We denote by $M_{mic} = \mathcal{O}(\hat{h}^{-d})$ the number of degrees of freedom (DOF) for the micro FEM and by M_{mac}, the number of DOF of the macro FEM. For quasi-uniform macro meshes, the macro meshsize H (sometimes denoted $H = 1/N_{mac}$) and the micro meshsize $\hat{h}$ are related to M_{mac} and M_{mic} as

$$H = \mathcal{O}(M_{mac}^{-1/d}), \qquad \hat{h} = \mathcal{O}(M_{mic}^{-1/d}).$$

A priori error analysis for the fully discrete method (4.5), first given in [2], gives an insight in this meshing problem and reveals that the optimal refinement of the micromesh read

$$\hat{h} := \frac{h}{\varepsilon} \simeq H^{\frac{\ell}{2q}} \quad (H^1 \text{ norm}), \qquad \hat{h} := \frac{h}{\varepsilon} \simeq H^{\frac{\ell+1}{2q}} \quad (L^2 \text{ norm}).$$

The corresponding complexity in term of macro DOF reads

$$\underbrace{H^{-d}}_{M_{mac}} \cdot \underbrace{H^{\frac{-d\ell}{2q}}}_{M_{mic}} \cdot n_s = (M_{mac})^{1+\frac{\ell}{2q}} \cdot n_s \qquad \text{for the } H^1 \text{ norm},$$
$$\underbrace{H^{-d}}_{M_{mac}} \cdot \underbrace{H^{\frac{-d(\ell+1)}{2q}}}_{M_{mic}} \cdot n_s = (M_{mac})^{1+\frac{\ell+1}{2q}} \cdot n_s \qquad \text{for the } L^2 \text{ norm},$$

where n_s denotes the number of sampling domains per macro element $K \in \mathcal{T}_H$.[7] For example, using piecewise linear polynomials on simplicial FEs, assuming quasi-uniform macro and micro meshes, and that the complexity is proportional to the total DOF we obtain a cost of $\mathcal{O}(M_{mac}^{3/2})$ (H^1 norm) and $\mathcal{O}(M_{mac}^2)$ (L^2 norm). These convergence rates are illustrated in Figure 8 on a very simple multiscale problem for which a reference solutions with high precision can easily be computed (see [2]). In contrast, the memory demand is proportional to $M_{mac} + M_{mic}$ only, as the micro problem, being independent of one another, can be solved one at a time. Finally, we note that by using spectral methods or high order FEM for the micro solvers, it is possible to reduce the total cost of the FE-HMM to log-linear complexity in the macro DOF. This was investigated in [10]. Such an approach requires however high regularity of the oscillating tensor a^ε which may not hold for some applications as for example in material science.

Example 1: homogenization problem with non-periodic tensor. We consider the Problem (4.1), with a log-normal conductivity tensor a^ε (taken from [13]). The domain Ω consists of a semi circle and a rectangle, meshed with 1137 nodes using 576 triangles and 784 quadrilaterals, respectively (see Figure 9a). We use the moving ellipse average method [139, Section 4.1] to

[7] Notice that as the micro problems are solved independently, the method is well suited for parallel implementation which can reduce significantly the complexity of the FE-HMM.

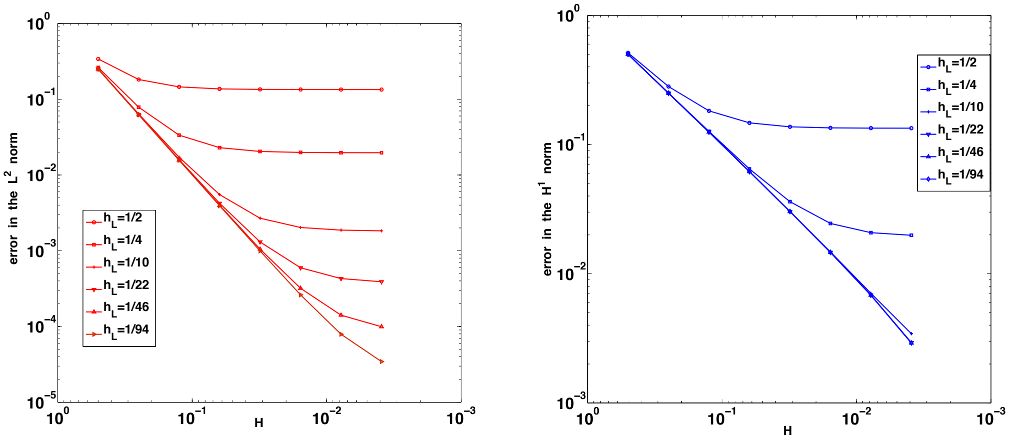


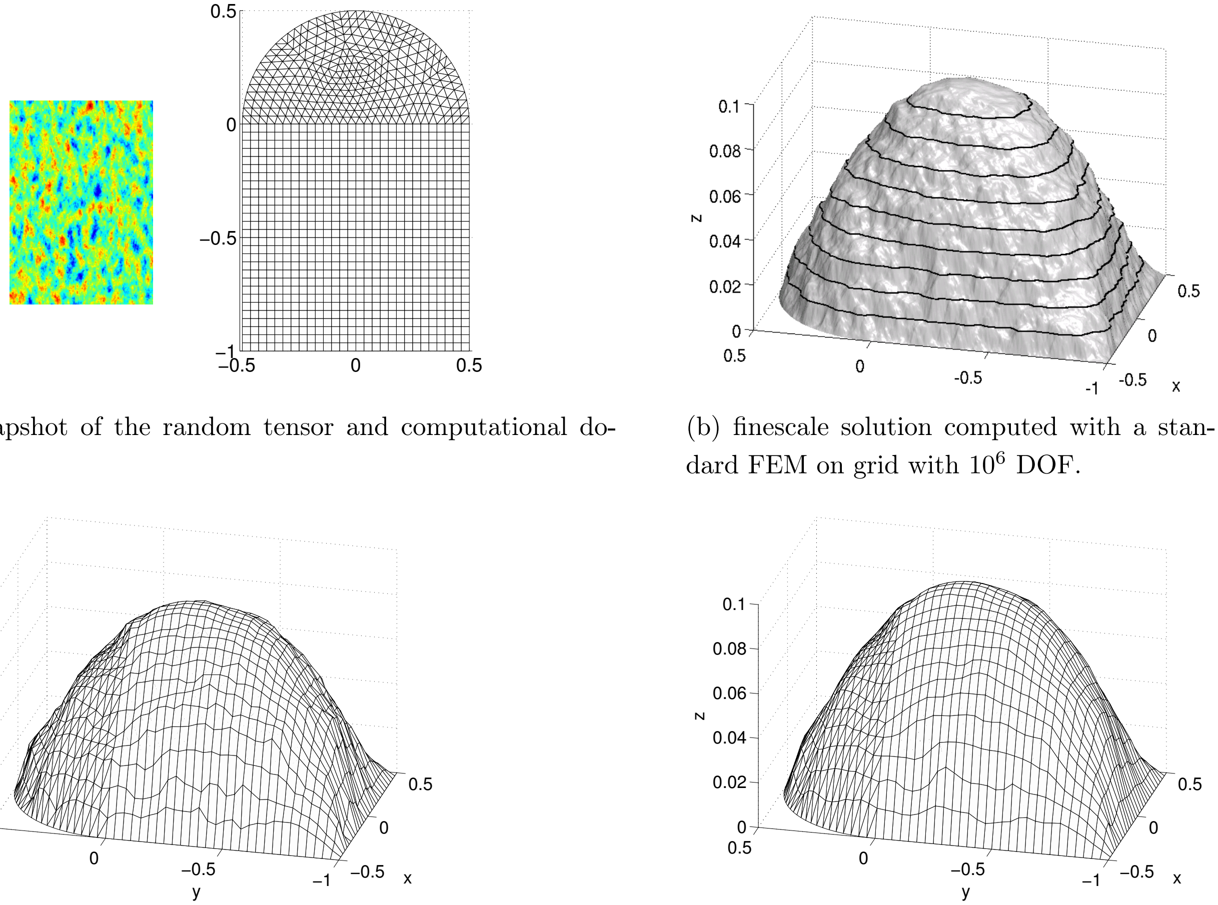
Figure 8: Error with respect to the homogenized solution u^0 for the FE-HMM applied to a 2D periodic problem (L^2 error left Figure, H^1 error right Figure). The horizontal axis represents the macro mesh size (decreasing from left to right). Each curve represents the error obtained by macromesh refinement for fixed micromeshes $h/\varepsilon = 1/2, 1/4, 1/10, 1/22, 1/46, 1/94$. Horizontal curves indicate a dominant micro error.

generate the realization of the log-normal stochastic field with mean zero and variance $\sigma = 1$. The correlation lengths of the stochastic field are set to be $\varepsilon_1 = 0.01$ and $\varepsilon_2 = 0.02$ in the x, y direction, respectively. A snapshot of this tensor is shown in Figure 9a.

N_{mic}	4	8	16	32	64	finescale
$\delta = 0.02$	0.2352	0.2415	0.2439	0.2449	0.2454	0.2583
$\delta = 0.06$	0.2313	0.2454	0.2520	0.2551	0.2567	0.2583

Table 4: Energy norms of the FE-HMM and finescale solutions of the problem described in Example 1 with a random tensor. FE-HMM results are given for various micro sampling-domain sizes $\delta \times \delta$ and DOF N_{mic}^2 of the micro problems ($N_{mic} = 1/\hat{h}$).

As this problem does not have an explicit analytical solution, we compute a finescale solution using a standard FEM with a fine mesh with about 10^6 degrees of freedom (DOF) in order to resolve the micro scale for a given realization (see Figure 9b). We compare this reference solution with the FE-HMM on the coarse macro grid with about 1100 DOF for the same realization. For the FE-HMM we present results for various sizes of the sampling domains ($\delta = 0.02$ and $\delta = 0.06$). We can see in Figures 9c and 9d, that the FE-HMM solution is closer to the reference solution as the sampling domain contains more correlation length of the random field. This observation can also be seen in Table 4 when comparing the energy norm of the various solutions obtained with the FE-HMM to the energy norm of the reference solution.



(a) Snapshot of the random tensor and computational domain.

(b) finescale solution computed with a standard FEM on grid with 10^6 DOF.

(c) FE-HMM with $N_{Mic} = 64$ and micro domain size $\delta = 0.02$ on a macro grid with 1100 DOF.

(d) FE-HMM with $N_{Mic} = 64$ and micro domain size $\delta = 0.06$ on a macro grid with 1100 DOF.

Figure 9: FE-HMM and finescale solution of the random problem described in Example 1. Snapshot of the tensor and comparison of the finescale solution with FE-HMM solutions with different size of sampling domains.

Example 2: a nonlinear problem, the Richards equation. We consider the steady-state Richards equation that describes the fluid pressure $u(x,t)$ in an unsaturated porous medium

$$-\nabla \cdot (a^{\varepsilon}(u^{\varepsilon}(x))\nabla(u^{\varepsilon}(x) - x_2)) = f(x) \text{ in } \Omega = (0,1)^2, \tag{4.11}$$

where x_2 is the vertical coordinate, and f corresponds to possible sources or sinks. Here a^{ε} is a multiscale permeability tensor that depends on the pressure u^{ε}, hence Equation (4.11) is a nonlinear nonmonotone problem. The application of the FE-HMM to such problems has first been considered in [68]. A complete analysis has recently been given in [18, 17] and is summarized in Section 4.4.2. The FE-HMM for such problem reads: find $u_H \in S_0^{\ell}(\Omega, \mathcal{T}_H)$ such that

$$B_H(u_H; u_H, w_H) = F_H(w_H), \ \forall w_H \in S_0^{\ell}(\Omega, \mathcal{T}_H), \tag{4.12}$$

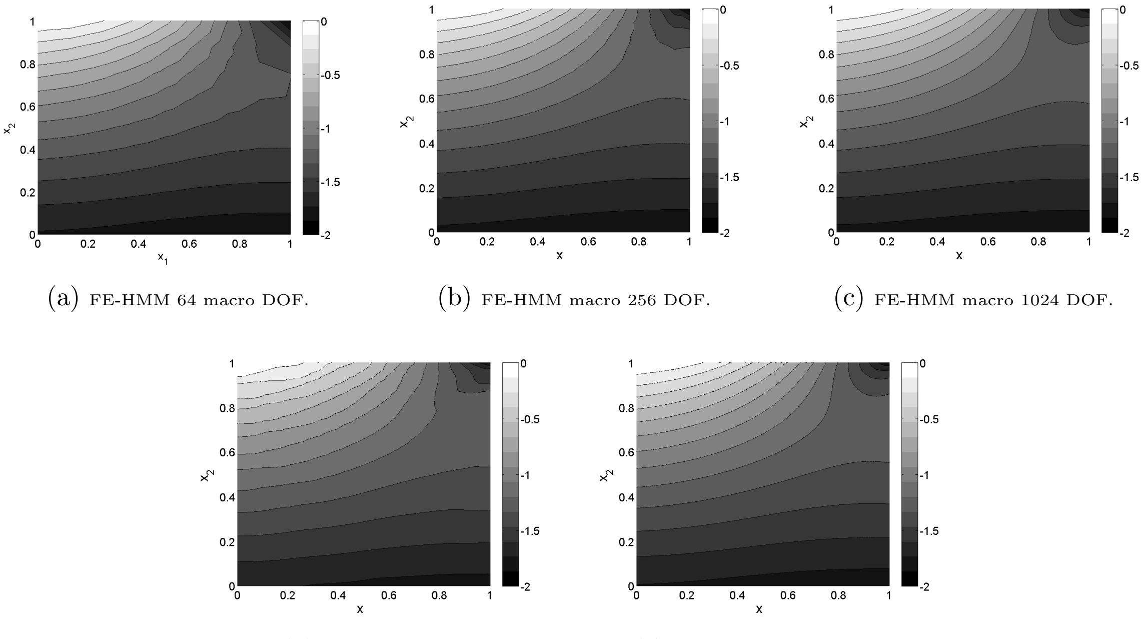
where

$$B_H(u_H; v_H, w_H) := \sum_{K\in\mathcal{T}_H} \sum_{j=1}^{J} \frac{\omega_{K_j}}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x, u_H(x_{K_j})) \nabla v_{h,K_j}^{u_H(x_{K_j})}(x) \cdot \nabla w_{h,K_j}^{u_H(x_{K_j})}(x) dx, \quad (4.13)$$

and $v_{h,K_j}^{u_H(x_{K_j})}$ (and similarly $w_{h,K_j}^{u_H(x_{K_j})}$) is a solution of a linear micro problem similar to (4.10). For the parameter $s = u_H(x_{K_j})$ it reads: find v_{h,K_j}^s such that $(v_{h,K_j}^s - v_{H,lin}) \in S^q(K_{\delta_j}, \mathcal{T}_h)$ and

$$\int_{K_{\delta_j}} a^\varepsilon(x, s) \nabla v_{h,K_j}^s(x) \cdot \nabla z_h(x) dx = 0, \quad \forall z_h \in S^q(K_{\delta_j}, \mathcal{T}_h). \quad (4.14)$$

We observe that while the macro problem (4.12) is nonlinear, the micro problems (4.14) are linear. Indeed, in the implementation $s = u_H(x_{K_j})$ is given by the current (available) state of the macro solution (e.g., using a Newton method for solving the macro problem). For the numerical simulation of (4.11) we set $f(x) \equiv 0$ for simplicity and consider the following boundary conditions



(a) FE-HMM 64 macro DOF.

(b) FE-HMM macro 256 DOF.

(c) FE-HMM macro 1024 DOF.

(d) FEM 1024 DOF.

(e) Resolved FEM $1.5{\cdot}10^6$DOF.

Figure 10: Richards problem (4.11) with mixte boundary conditions (Dirichlet and Neumann). Top pictures: level curves of the FE-HMM solutions with $N_{mic} = N_{mac}$. The macro DOF are indicated. The picture (d) and (e) represent the level curves of a standard FEM with respectively unresolved (1024 DOF) and resolved ($1.5 \cdot 10^6$ DOF) meshes for the given micro scale.

$$\begin{aligned}
&u^\varepsilon(x) = -1.9x_1^2 \ \text{ on } \partial\Omega_D = [0,1] \times \{1\}, \\
&n \cdot (a^\varepsilon(u^\varepsilon(x)) \nabla(u^\varepsilon(x) - x_2)) = 0 \ \text{ on } \partial\Omega_N = \{0,1\} \times [0,1] \cup [0,1] \times \{0\},
\end{aligned}$$

and an exponential model for the permeability tensor a^ε similar to the one in [48, Sect. 5.1],

$$a^\varepsilon(x,s) = \alpha^\varepsilon(x) e^{\alpha^\varepsilon(x)s} \quad \text{where } \alpha^\varepsilon(x) = \frac{1/117.4}{2 + 1.8\sin(2\pi(2x_2/\varepsilon - x_1/\varepsilon))}. \tag{4.15}$$

Numerical solutions are reported in Figure 10. Figures 10a-10c are solutions of the FE-HMM with $\mathcal{P}^1$-triangular FEs and a decreasing size of macromeshes (uniform macromeshes with $H = 1/8, 1/16, 1/32$). The meshes of the micro solver are refined according to the "optimal L^2 refinement strategy" $h/\varepsilon \simeq H$. The numerical results obtained with the FE-HMM are compared to a reference solution of the problem (4.11) (obtained by a resolved standard FEM) plotted in Figure 10e (for $\varepsilon = 10^{-2}$, $\sim 10^6$ degrees of freedom are used). We also plot for comparison in Figure 10d, the result obtained by a standard FEM on a coarse 32×32 mesh that does not resolve the fine oscillations. We observe that this unresolved FEM does not give a correct qualitative result. In contrast, the FE-HMM captures the correct behavior of the problem at a much lower computational cost.

Example 3: a crack problem and adaptive FE-HMM.

In our next experiment we consider a crack problem in a heterogeneous medium, characterized by a highly oscillating conductivity tensor

$$\begin{aligned} -\nabla \cdot (a^\varepsilon(x) \nabla u^\varepsilon) &= 1 \quad \text{in } \Omega, & (4.16)\\ u^\varepsilon &= g_D \quad \text{on } \Gamma_D = \partial\Omega, & (4.17) \end{aligned}$$

on a domain $\Omega = \{|x| + |y| < 1\} \setminus \{0 \le x \le 1,\ y = 0\}$ with a crack along the positive x-axis (see Figure 11). As the solution of the homogenized problem is not in $H^2(\Omega)$, it is well-known that

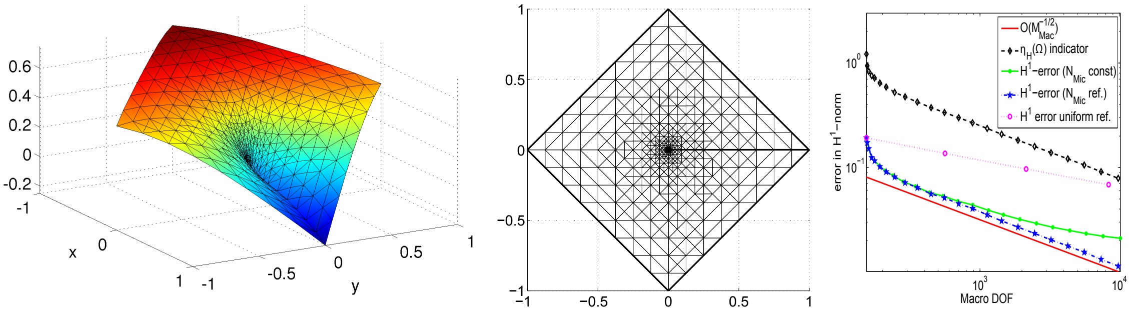


Figure 11: FE-HMM solution and mesh after 10 iterations for the crack-problem described in Example 3 using highly oscillating locally periodic coefficients (left and middle Figs.) Rate of decay for various errors and for the error indicator in the H^1 norm (right Fig.).

a standard piecewise linear FEM for the effective problem will give sub-optimal convergence rates. In this situation, adaptive mesh refinement is needed. By using local error indicators, one identifies, marks and refines those elements that contribute the most to the global error in order to better equidistribute the error in the FE mesh. Overall the procedure consists in the following cycle:

$$\text{SOLVE} \longrightarrow \text{ESTIMATE} \longrightarrow \text{MARK} \longrightarrow \text{REFINE}.$$

An energy based a posteriori error analysis for a standard FEM relies usually on local error indicators η_h that comprise two components: the jump residual that measures the discontinuity of the normal flux across elements interfaces and the element residual, a measure of the accuracy of the solution inside each element. Both quantities depend on the (conductivity) tensor of the elliptic problem. Residual based adaptive FE-HMM for the energy norm have been proposed and analyzed in [14, 12] and also in [120], where an a posteriori error estimate in a “two-scale norm" has been derived. For the FE-HMM, data involved in the usual error indicator are not available beforehand but computed during the integration process. An important ingredient for adaptive FE-HMM is an effective jump based on “multiscale fluxes" first defined in [4, 7]. Let $\mathcal{T}_H$ denote a conformal mesh and let $\mathcal{E}_H$ be the set of interfaces. Two elements sharing an interface $e \in \mathcal{E}_H$ are labeled K^+ and K^-. Consider the micro functions u_{h,K^+} and u_{h,K^-} solutions of (4.10) in the two sampling domains K_δ^+ and K_δ^- of the elements K^+ and K^-, respectively, constraint by the macro solution u_H of (4.5). The jump of multiscale fluxes is then defined as (see Figure 12)

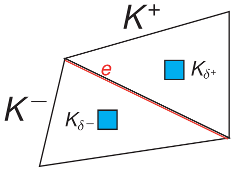


Figure 12: Sketch of the ingredients for the jump of multiscale fluxes defined in (4.18).

$$[\![\overline{a^\varepsilon(x)\nabla u_h}]\!]_e := \left(\frac{1}{|K_\delta^+|}\int_{K_\delta^+} a^\varepsilon(x)\nabla u_{h,K^+}\,dx - \frac{1}{|K_\delta^-|}\int_{K_\delta^-} a^\varepsilon(x)\nabla u_{h,K^-}\,dx\right)\cdot n_e, \qquad (4.18)$$

where the unit outward normal n_e is chosen to be $n_e = n^+$. We omit the index K_δ for the micro solutions u_h in $[\![\overline{a^\varepsilon(x)\nabla u_h}]\!]_e$ as the jump over e involves two sampling domains in adjacent elements. We also assume for simplicity that (4.5) is solved using a piecewise linear macro FE space (see [15] and [117] for a generalization of the “the jump of multiscale fluxes" and the a posterior error analysis for higher order macro FEs). Then, the local error indicator $\eta_H(K)$ on an element K is defined by [14, 12]

$$\eta_H(K)^2 := H_K^2\,\|f_H\|_{L^2(K)}^2 + \frac{1}{2}\sum_{e\subset\partial K} H_e \left\| [\![\overline{a^\varepsilon \nabla u_h}]\!]_e \right\|_{L^2(e)}^2, \qquad (4.19)$$

where f_H is a piecewise constant approximation of f. The quantity $\eta_H(K)^2$ is a measure of the local error $\|u^0 - u_H\|_{H^1(K)}^2$ and is used to identify the elements that contributes most to the error. These elements are then marked for refinement. In order to prove the reliability and the efficiency of an a posteriori error estimate, rigorous upper and lower bounds of the error in terms of the residual have been derived in [14, 12]. For such bounds, one needs to introduce the data approximation error $\xi_H(K)$ on an element K by

$$\xi_H(K)^2 := H_K^2 \|f_H - f\|_{L^2(K)}^2 + \left\|\left(a_K^0 - a^0(x)\right)\nabla u_H\right\|_{L^2(K)}^2, \tag{4.20}$$

where $a^0(x)$ is the unknown homogenized tensor of problem (4.3) and a_K^0 is the effective macro tensor (in the macro element K) that is recovered by the FE-HMM (see Lemma 3 of section 4.4). In order to control the data approximation error, the quantity $\|(a_K^0 - a^0(x))\nabla u_H\|_{L^2(K)}^2$ needs to be quantified. Under suitable assumptions [14, 12] (e.g., (**H1**), (**H2**) of Section 4.4) one can prove that

$$\sup_{x\in K}\left\|\left(a_K^0 - a^0(x)\right)\nabla u_H\right\|_{L^2(K)} \le C\left(H_K + \left(\frac{h}{\varepsilon}\right)^2\right) + e_{MOD}, \tag{4.21}$$

where e_{MOD} is a quantity independent of H, h. This shows that the micro-macro mesh refinement described at the beginning of Section 4.3.3 should now be performed locally. In turn, this local refinement has an important consequence on the complexity of the FE-HMM. While for uniform refinement one needs to refine every sampling domain of each macro element, in an adaptive mesh refinement strategy, the micromesh will be refined only in the sampling domains of a macro element marked for refinement (at the above rate $\frac{h}{\varepsilon} = \sqrt{H_K}$). All other computations in the sampling domains, i.e., all the microfunctions computed in a previous iteration can be re-used in the next iteration in unrefined macro elements. The saving in computational complexity is illustrated in Table 5, where we observe that only a small fraction of the micro problems computed via uniform refinement needs to be computed using adaptive and local refinement as described above.

$\hat{h}$	1/8	1/16	1/24	1/32	1/40	1/48
adaptive FE-HMM, 10th iteration	278	218	60	24	28	40
uniform FE-HMM, 4th iteration	-	-	-	16384	-	-

Table 5: Number of micro problems with various $\hat{h} = h/\varepsilon$ resolution to obtain an error of $\left\|e^H\right\|_{H^1(\Omega)} \le 0.07$ for the crack problem described in Example 3 (computation with a periodic tensor).

We present in Figure 11 the decay of the error after several iterations of the adaptive FE-HMM applied to the crack problem of Example 3. In order to have an exact homogenized solution for comparison purpose, we choose a periodic tensor in our computation (details can be found in [14, Sect. 6.2]). Example with random tensors can also be found in [14, 117]. The rate of convergence of the error and the error indicator, reported in Figure 11, confirm the theoretical rate of $\mathcal{O}\left(M_{mac}^{-1/d}\right)$, where we recall that M_{mac} denotes the macro degrees of freedom. We also see that the local error indicator η_H decays with the right slope and the effectivity index $\eta_H(\Omega)/\|u^0 - u_H\|_{H^1(\Omega)}$ is comparable to the effectivity index for single scale residual based adaptive FEM [74]. We also illustrate in Figure 11 the use of uniform refinement and adaptive FE-HMM without micro refinement. Both strategies give the wrong convergence rate for this problem.

Finally, we briefly mention another type of adaptivity, often more interesting for engineers, namely adaptivity in quantities of interest. Suppose that we are interested in a quantity of interest of the exact solution $J\left(u^0\right)$, where J denotes a linear bounded functional and u^0 is the homogenized solution of a given multiscale problem (e.g., (4.1)). The question is now: is it possible to refine the numerical solution of a multiscale method, say the FE-HMM, to have an approximation of the quantity of interest $J\left(u_H\right) \simeq J\left(u^0\right)$?

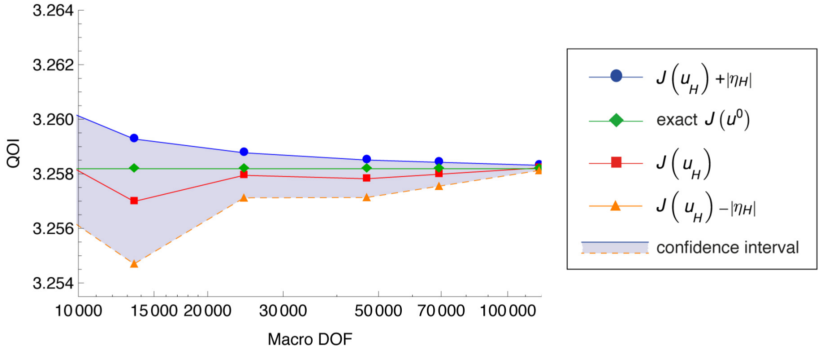


Figure 13: Illustration of the goal-oriented FE-HMM that allows to specify a confidence interval, the shaded area, where the exact quantity of interest (here a regularized pointwise derivative) is expected to be.

For single scale problems, such “goal-oriented adaptivity" has been studied in [123, 118, 34, 116, 22]. We mention also the related work of Oden and co-worker on adaptive control of model [119]. In the context of numerical homogenization, the numerical analysis literature seems rather scarce on the topic. In [15] a posteriori error estimates in quantities of interest for the FE-HMM are discussed. The quantity $J(u^0 - u_H)$ is shown to have an exact representation in terms of the local error estimator and the data approximation error. Provided that certain higher order approximation term can be neglected (see [116, 22] for a discussion on this issue), it is possible to find a “confidence interval" for the estimation of the error in the given quantity of interest. This is illustrated in Figure 13, where the quantity of interest is a regularized pointwise derivative of an effective solution corresponding to a multiscale elliptic problem. The solution u_H is obtained from the FE-HMM with goal-oriented adaptive refinement. We refer to [15] for details.

Example 4: three-dimensional problems. In our last example, we consider the heat distribution in a microprocessor. The model considered in our computations depicted in Figure 14 consist of a silicon IC chip, a lead frame, that acts as a heat spreader, and a mold resin encapsulation (the packaging covering the IC chip for protection). Due to the increasing packaging density in such devices and the resulting temperature growth, the use of composite materials with high thermal conductivity is crucial. The microstructure of the composite materials, whose representative size is denoted by ε, is responsible for the multiscale behavior of the associated heat conductivity problem. The application of the FE-HMM to this example has first been considered in [13], where additional 3D examples are also discussed (see also [117]).

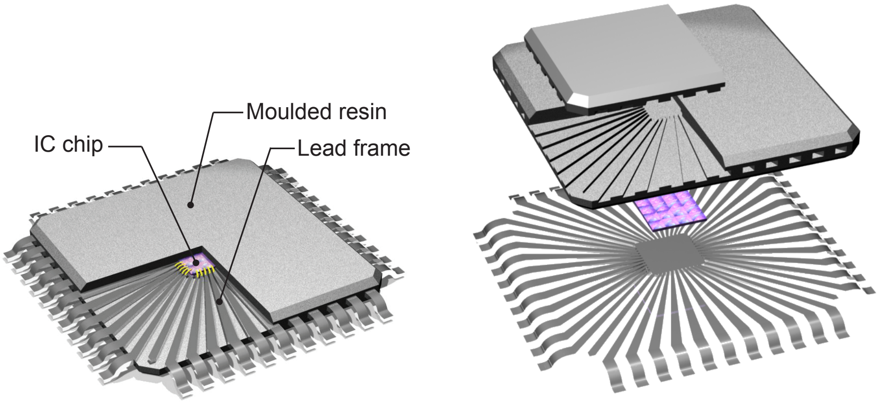


Figure 14: Model of microprocessor used for the numerical computation of Example 4

Geometry and mesh. The size of the full processor model is $12.2 \times 12.2 \times 1$ mm^3. We use a macromesh generated by CUBIT [129] that consists of 81,000 grid points with a maximum tetrahedron volume of $1.4 \cdot 10^{-3}$mm^3 (see Figure 15). Scale resolution for a composite with $\varepsilon = 10^{-6}$m as used below, with about 10 grid points per oscillation (in each spatial direction) would result in a mesh with 10^{12} grid points and thus computationally unfeasible with a standard FEM. In contrast, numerical experiments with realistic values of ε can be computed without difficulties with the FE-HMM. For numerical comparisons, we also present numerical experiments with a relatively large value of ε, $\varepsilon = 5 \cdot 10^{-4}$m. For this value we compute a reference finescale solution that involves 3.9 million grid points and about 22 million tetrahedra. The heat transfer by conduction is modeled by Problem (4.1). Convective heat transfer with the surrounding air is modeled by the Robin and Neumann boundary conditions

$$n \cdot (a^{\varepsilon} \nabla u^{\varepsilon}) + \alpha u^{\varepsilon} = \quad g_R \quad \text{on } \partial\Omega_R, \tag{4.22}$$

$$n \cdot (a^{\varepsilon} \nabla u^{\varepsilon}) = \quad g_N \quad \text{on } \partial\Omega_N, \tag{4.23}$$

where Ω is the domain of the considered object and $\partial\Omega_R$ and $\partial\Omega_N$ are the surfaces of the three dimensional object with Robin and Neumann boundary conditions, respectively. The right hand side of the Robin boundary condition is given by $g_R = q_0 + \alpha u_{amb}$.

In practice, conductivity tensors for realistic materials could be obtained via imaging techniques (our computational strategy could accommodate such data). Here we use simplified

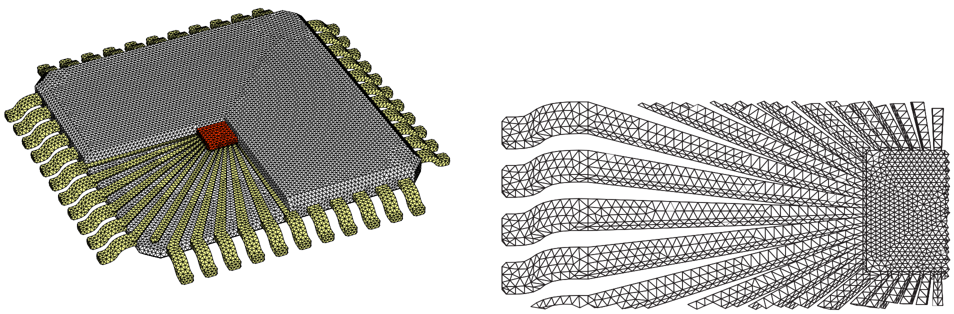


Figure 15: macromesh of the 3D microprocessor problem of Example 4 81,000 grid points and 430,000 tetrahedra (left Figure) with zoom on the macromesh used for the lead frame (right Figure).

tensors (one for each material). The tensor corresponding to the lead frame $a^{\varepsilon}_{\text{leadframe}}$ is chosen as to have a non-periodic slow variation to mimic a change in the material structure from the center of the lead frame to the periphery. For the resin, we chose $a^{\varepsilon}_{\text{resin}}$ to be oscillating, anisotropic with a larger conductivity in the z-direction, while the tensor corresponding to the chip $a^{\varepsilon}_{\text{chip}}$ is assumed to be constant (see [13] for details). Other data are defined as follows: the power of the chip is set to $P_{chip} = 0.125\,\text{W}$ and the chip size is chosen to be $V = 2 \times 2 \times 0.2$ mm^3. This yields an external heat flux f in (4.1) of $f = \frac{P}{V} = 1.875 \cdot 10^8 \frac{\text{W}}{\text{m}^3}$. The room temperature (in Kelvin) is set to $u_{amb} = 293.15$ K and the heat transfer coefficient to $\alpha = 20\frac{\text{W}}{\text{m}^2\text{K}}$, a rough estimate for the effect of air cooling.

		$N_{mic} = 4$	$N_{mic} = 8$	$N_{mic} = 16$	finescale	average
$\varepsilon = 5 \cdot 10^{-4}$	$\|\|u\|\|_A$	0.3906	0.4961	0.5514	0.5189	0.3122
$\varepsilon = 10^{-6}$	$\|\|u\|\|_A$	0.3894	0.4963	0.5512	-	0.3122

Table 6: Energy norm of various solutions of the heat transfer problem in a microprocessor (FE-HMM solution, resolved solution, solution obtained using arithmetic average for the various tensors). Here micro DOF are on the sampling domains are given by N^d_{mic}.

In Figure 16 we compare (for $\varepsilon = 5 \cdot 10^{-4}$m) the results obtained by the FE-HMM with a finescale (resolved) solution. For the FE-HMM two different resolutions of the micro FEM are used. A good qualitative agreement between the FE-HMM and the finescale solutions is observed. For comparison we also plot a solution obtained on the same macro grid as the FE-HMM but with a standard FEM using a naive averaging procedure for the microstructure (here the arithmetic average). We observe that the conductivity is overestimated in this situation, leading to a wrong qualitative behavior of the heat distribution in the microprocessor. To get a rough estimate of the quality of the various experiments performed in this section, we provide

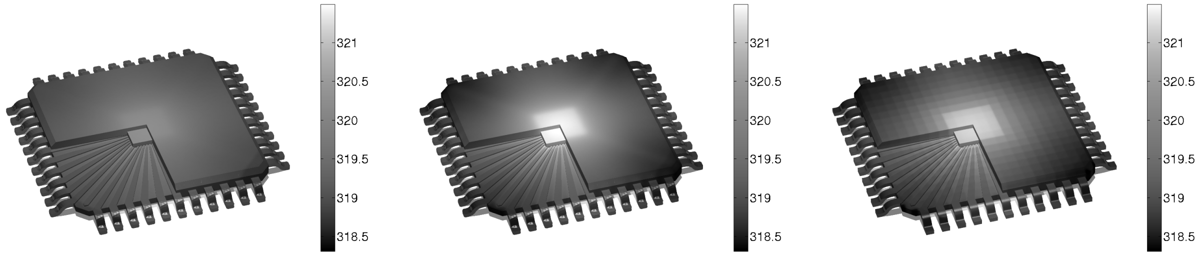


Figure 16: Steady state heat distribution in the 3D microprocessor problem of Example 4 with $\varepsilon = 5 \cdot 10^{-4}$m, 81000 DOF in the macromesh. Comparison of various solutions. Left Figure averaged tensor (arithmetic mean), 81000 DOF, middle Figure FE-HMM solution with 81000 macro DOF, right Figure finescale solution $3.9 \cdot 10^{6}$ DOF. The color bar represents the temperature in Kelvin [K].

in Table 6 the value of the energy norm of the FE-HMM calculation (for various resolutions of the microstructure) and the energy norm of the resolved solution.

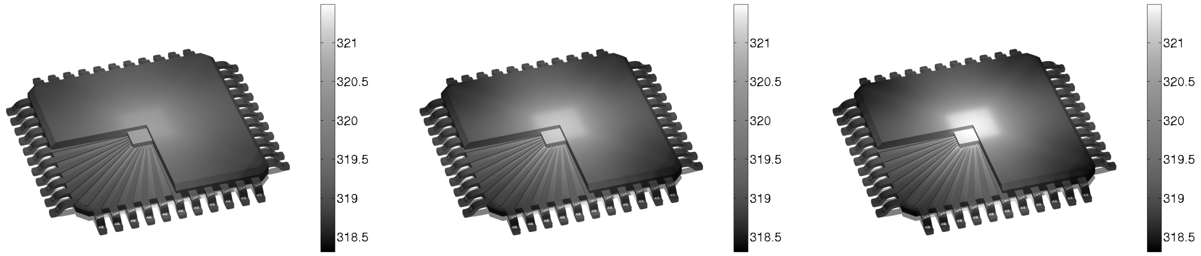


Figure 17: Steady state heat distribution in the 3D microprocessor problem of Example 4, with $\varepsilon = 10^{-6}$m, 81000 DOF in the macromesh. Comparison of FE-HMM solutions with different resolutions of the micro problems ($N_{mic} = 4, 8, 16$ from left to right). The color bar represents the temperature in Kelvin [K].

Finally, we present in Figure 17 computations with a realistic value of ε, ($\varepsilon = 10^{-6}$m). We notice that, a computation with this value of ε is not more expensive than previous computations with a coarser value of ε (recall that the FE-HMM capture the effective solution). The computational cost however depends on the resolution of the microstructure in the sampling domains, i.e., the number of points per wavelengths. Indeed, as can be seen in Figure 17, the effect of the micro error at the macro scale is not negligible (see again [2] and Section 4.4 for an analysis of this behavior).

4.3.4 More sophisticated coupling

In all of the above examples, we used standard FEM at the micro and the macro levels. The structure of the FE-HMM algorithm that couples the macro and micro solvers only trough constraints, allows to use other type of methods.

For many problems, local conservation properties in the numerical approximation and flexibility in meshing (e.g., hanging nodes, local refinements) are desirable. For single scale problems, these requirements have triggered the development of discontinuous Galerkin (DG) FEMs. Such methods have been extensively studied for hyperbolic problems, advection-diffusion and diffusion problems (see [28] for a review). While a large body of literature is available for DG methods applied to single scale problems, the construction and analysis of multiscale DG methods have rarely been addressed. In the context of the HMM, a DG methods for hyperbolic and parabolic one-dimensional scalar problems has been proposed in [47]. For multiscale elliptic problems, a DG method based on HMM has been proposed in [4, 7].

Another issue, as seen in Section 4.3.3, is the cost of the repeated micro solutions in sampling domains in the FE-HMM. If the microstructure is regular enough, one can take advantage of fast micro solvers (e.g., based on spectral methods) to considerably speed-up the computation time. The simultaneous refinement of macro and micromeshes can be avoided in this situation (see [10]).

If in contrast there is only low regularity of the micro solutions, one can try yo avoid repeated micro problems by suitable interpolation of the micro solutions in a few representative sampling domains. Provided that there is some smoothness in the micro solution with respect to the macro variable, this strategy can be successful. The use of reduced basis techniques for numerical homogenization was first explored in [38] and recently integrated in the HMM methodology in [8], where a reduced basis FE-HMM has been proposed and analyzed.

4.4 Error estimates

In this section we discuss a priori and a posteriori error analysis for the FE-HMM. We present the main steps of the a priori error analysis for linear elliptic problems in Section 4.4.1. We only discuss the analysis for the FE-HMM based on macro and micro FEMs, and we refer to [10] for an analysis using micro pseudo-spectral methods, to [7] for an analysis using discontinuous Galerkin macro methods and to [8] for an analysis incorporating reduced basis methods at the micro level. In Section 4.4.2 we briefly mention how the results obtained in the linear case can be generalized for a class of nonlinear problems. For details on the analysis of such problems, we refer to [18, 17]. Finally we also discuss a posteriori error analysis for the FE-HMM in Section 4.4.3, following [14, 12]. Unless otherwise specified, we shall assume that the macroscopic triangulation $\mathcal{T}_H$ that is conformal and shape regular.

4.4.1 A priori error analysis

We collect here a few preliminary results that are useful for the analysis of the FE-HMM.

Energy equivalence and coercivity (see [2, 68, 5]). As an application of the following lemma, we deduce the coercivity of the bilinear form (4.6), hence the existence and uniqueness of the problem (4.5).

Lemma 2. *Let $v_H \in S_0^\ell(\Omega, \mathcal{T}_H)$ and let v_{h,K_j} be the corresponding solution of (4.10) in $S^q(K_{\delta_j}, \mathcal{T}_h)$ with boundary conditions given by (4.8) or (4.9). Assume that* (4.2) *hold. Then we have*

$$\|\nabla v_{H,lin}\|_{L^2(K_{\delta_j})} \le \|\nabla v_{h,K_j}\|_{L^2(K_{\delta_j})} \le C\|\nabla v_{H,lin}\|_{L^2(K_{\delta_j})}, \tag{4.24}$$

where $v_{H,lin}$ is defined in Section 4.3.2.

Reformulation of the FE-HMM (see [5, 7, 68]). The bilinear form (4.6) can be reformulated in the following form.

$$B_H(v_H, w_H) = \sum_{K\in\mathcal{T}_H} \sum_{j=1}^{J} \omega_{K_j} a_K^0(x_{K_j}) \nabla v_H(x_{K_j}) \cdot \nabla w_H(x_{K_j}). \tag{4.25}$$

In order for the expression (4.6) to be equal to the expression (4.25), we see that the following identity must hold

$$\frac{1}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x) \nabla v_{h,K_j} \cdot \nabla w_{h,K_j} dx = \frac{1}{|K|} \int_K a_K^0(x_{K_j}) \nabla v_{H,lin} \cdot \nabla w_{H,lin} dx, \tag{4.26}$$

for all $v_H, w_H \in S_0^\ell(\Omega, \mathcal{T}_H)$ and for all v_{h,K_j} (resp. w_{h,K_j}) solutions of problem (4.10). The following lemma gives an appropriate definition for $a_K^0(x_{K_j})$ so that the above equality is valid (see [7]).

Lemma 3. *For $\mathbf{e_i}$, $i = 1, \ldots, d$ (the canonical basis of $\mathbb{R}^d$) consider the following problem: find $\psi_{h,K_j}^i \in S^q(K_{\delta_j}, \mathcal{T}_h)$ such that*

$$\int_{K_{\delta_j}} a^\varepsilon(x) \nabla \psi_{h,K_j}^i \cdot \nabla z_h dx = -\int_{K_{\delta_j}} a^\varepsilon(x) \mathbf{e_i} \cdot \nabla z_h dx, \ \forall z_h \in S^q(K_{\delta_j}, \mathcal{T}_h), \tag{4.27}$$

where $S^q(K_{\delta_j}, \mathcal{T}_h)$ is defined in (4.7) *with either periodic or Dirichlet boundary conditions. If $a_K^0(x_{K_j})$ is defined as*

$$a_K^0(x_{K_j}) = \frac{1}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x) \left(I + J_{\psi_{h,K_j}}^T(x)\right) dx, \tag{4.28}$$

where $J_{\psi_{h,K_j}}(x)$ is a $d \times d$ matrix with entries $\left(J_{\psi_{h,K_j}}(x)\right)_{i\ell} = \frac{\partial \psi_{h,K_j}^i}{\partial x_\ell}(x)$, then the equality (4.26) *holds true.*

Remark 4. *We observe that a standard FEM with numerical quadrature for the effective problem* (4.3) *reads (see [50]): find $u_{0,H} \in S_0^\ell(\Omega, \mathcal{T}_H)$ such that*

$$B_{0,H}(u_{0,H}, v_H) = \int_\Omega f v_H dx, \quad \forall v_H \in S_0^\ell(\Omega, \mathcal{T}_H), \tag{4.29}$$

where

$$B_{0,H}(v_H, w_H) = \sum_{K\in\mathcal{T}_H} \sum_{j=1}^{J} \omega_{j,K} a^0(x_{K_j}) \nabla v_H(x_{K_j}) \cdot \nabla w_H(x_{K_j}). \tag{4.30}$$

Thus, (4.6) *can be seen as a FEM with numerical quadrature for a modified effective problem. We expect* (4.25) *to be close to* (4.30)*, hence u_H to be close to $u_{0,H}$, if the procedure to recover the macro data from the microscale computation is accurate enough. This will be seen in the error analysis presented below.*

The first theorem gives the macroscopic error of the FE-HMM (see [1, 6, 68]). It does not rely on structure assumption (e.g., periodicity) of the multiscale problem (4.1), but requires sufficient smoothness of the data and the solution of the effective problem (4.3). For a given natural number ℓ and $\mu = 0$ or 1, we assume that the solution u^0 and the tensor a^0 of (4.3) satisfy

(**H1**) $u^0 \in H^{\ell+1}(\Omega)$, $\quad a_{ij}^0 \in W^{\ell+\mu,\infty}(\Omega)$ for $i, j = 1, \dots, d$.
We notice that hypothesis (**H1**) is the usual assumption in order for a FEM with numerical quadrature to converge with optimal rate H^ℓ and $H^{\ell+1}$ in the H^1 and L^2 norms, respectively.

Theorem 5 (**Macro error).** *Let u^0, u_H be the solutions of problems* (4.3),(4.5)*, respectively. Suppose that* (4.2)*, (**Q1**), (**Q2**) and (**H1**) hold.* [8] *Then we have the following estimates*

$$\|u^0 - u_H\|_{H^1(\Omega)} \leq C\left(H^l + e(HMM)\right), \tag{4.31}$$

$$\|u^0 - u_H\|_{L^2(\Omega)} \leq C\left(H^{l+1} + e(HMM)\right), \tag{4.32}$$

where C is independent of H and h and

$$e(HMM) = \sup_{K\in\mathcal{T}_H, x_{K_j}\in K} \|a^0(x_{K_j}) - a_K^0(x_{K_j})\|_F, \tag{4.33}$$

where $a_K^0(x_{K_j})$ is defined in (4.28) *and $\|\cdot\|_F$ is the Frobenius norm.*

Proof. Let $u_{0,H}$ be the solution of (4.29). It is proved in [50] that provided (4.2), (**Q1**), (**Q2**) and (**H1**) hold, we have

$$\|u^0 - u_{0,H}\|_{H^1(\Omega)} \leq CH^l, \tag{4.34}$$

$$\|u^0 - u_{0,H}\|_{L^2(\Omega)} \leq CH^{l+1}, \tag{4.35}$$

[8] In hypothesis (**H1**), $\mu = 0$ for the estimate (4.31) and $\mu = 1$ for the estimate (4.32).

where C is independent of H. Using the coercivity of the bilinear form (4.30) (which follows from (**Q1**)), we obtain

$$\|u_{0,H} - u_H\|_{H^1(\Omega)} \leq C \sup_{w_H \in S_0^\ell(\Omega,\mathcal{T}_H)} \frac{|B_{0,H}(u_H, w_H) - B_H(u_H, w_H)|}{\|w_H\|_{H^1(\Omega)}}.$$

In view of (4.30),(4.25) and (**Q2**), using the Cauchy-Schwarz inequality we see that

$$|B_{0,H}(u_H, w_H) - B_H(u_H, w_H)| \leq C \sup_{K \in \mathcal{T}_H, x_{K_j} \in K} \|a^0(x_{K_j}) - a_K^0(x_{K_j})\|_F \|u_H\|_{H^1(\Omega)} \|w_H\|_{H^1(\Omega)}.$$

The above estimates together with (4.34) and (4.35) give the claimed result. □

The next task is to estimate the quantity $e(HMM)$. It is convenient to further decompose this quantity. We therefore consider for each macro element $K \in \mathcal{T}_H$ and each sampling domain $K_{\delta_j} \subset K$

$$\bar{a}_K^0(x_{j,K}) = \frac{1}{|K_{\delta_j}|} \int_{K_{\delta_j}} a^\varepsilon(x) \left(I + J^T_{\psi_{K_j}}(x) \right) dx, \tag{4.36}$$

a tensor defined similarly as in (4.36) but here $\left(J_{\psi_{K_j}}(x)\right)_{i\ell} = \frac{\partial \psi^i_{K_j}}{\partial x_\ell}(x)$ and $\psi^i_{K_j}$ is the (exact) solution of (4.27) in (4.8) or (4.9). Then obviously,

$$e(HMM) \leq e_{MIC} + e_{MOD},$$

where

$$\begin{aligned} e_{MIC} &:= \sup_{K \in \mathcal{T}_H, x_{K_j} \in K} \|\bar{a}_K^0(x_{K_j}) - a_K^0(x_{K_j})\|_F, && (4.37) \\ e_{MOD} &:= \sup_{K \in \mathcal{T}_H, x_{K_j} \in K} \|a^0(x_{K_j}) - \bar{a}_K^0(x_{K_j})\|_F. && (4.38) \end{aligned}$$

For symmetric tensors, the quantity e_{MIC} has first been analyzed in [1] and the quantity e_{MOD} in [68]. To analyze e_{MIC} appropriate regularity of the oscillating tensor is required. As we use standard a priori results of FEMs in the sampling domain, we also need appropriate regularity of $\psi^i_{K_j}$. However, the coefficient a^ε is allowed to be discontinuous (at the macroscopic level) through smooth interfaces. We therefore make the following assumption

(**H2**) for $q \in \mathbb{N}$ we assume that $|\psi^i_{K_j}|_{H^{q+1}(K_{\delta_j})} \leq C\ \varepsilon^{-q} \sqrt{|K_{\delta_j}|}$, with C independent of ε, of the quadrature points x_{K_j} and the domain K_{δ_j}.

Remark 6. *If one assumes $a^\varepsilon \in W^{1,\infty}(\Omega)$ and that $|a^\varepsilon_{ij}|_{W^{1,\infty}(\Omega)} \leq C\varepsilon^{-1}$ for $i, j = 1, \ldots, d$, then (***H2***) can be proved for $q = 1$ and $W(K_{\delta_j}) = H^1_0(K_{\delta_j})$ following classical H^2 regularity results [98, Chap. 2.6] (notice that in fact we only need local regularity for a^ε_{ij}). If $a^\varepsilon = a(x, x/\varepsilon) = a(x, y)$ Y-periodic in y, $\delta/\varepsilon \in \mathbb{N}$, and $W(K_{\delta_j}) = W^1_{per}(K_{\delta_j})$, then (***H2***) can be established for a given q, provided that the tensor a^ε is smooth enough (this follows classical regularity results for solutions of periodic boundary value problems, e.g., [36, Chap. 3]).*

The following theorem has first been obtained in [1] for elliptic problem and in [3] for problem in elasticity (see also [5, 7] for generalizations). In [59] the estimate has been extended to non-symmetric tensors (see also [17, Lemma 4.6] for a short alternative proof).

Theorem 7 **(Micro error).** *Assume that* (4.2) *and (***H2***) hold. Then*

$$e_{MIC} \leq C \left(\frac{h}{\varepsilon}\right)^{2q}, \tag{4.39}$$

where C is independent of H, h and ε.

As first observed in [1, 5] we see that a fixed number of grid points per wavelength of the smallest oscillation of the problem does not ensure a robust convergence of the FE-HMM. Indeed, the above theorem shows that for fixed micromeshes, the error due to the micro solver are dominant for fine enough macromeshes. Comparing the rate of convergence of the macro solver (Theorem 5) with the rate of convergence of the micro solver (see Theorem 7) gives a criterion to obtain optimal (macro) convergence rate with the minimal computational cost. Indeed, if the macro problem (4.5) is solved in $S_0^{\ell}(\Omega, \mathcal{T}_H)$ with micro problems (4.10) solved in $S^q(K_{\delta_j}, \mathcal{T}_h)$, then, assuming uniform macro and micromeshes, we have with $\hat{h} = \frac{h}{\varepsilon}$

$$\begin{aligned} \hat{h} &\simeq H^{\frac{\ell}{2q}} \text{ (optimal refinement in } H^1 \text{ norm)}, && (4.40)\\ \hat{h} &\simeq H^{\frac{\ell+1}{2q}} \text{ (optimal refinement in } L^2 \text{ norm)}. && (4.41)\end{aligned}$$

We recall here that $\hat{h} = h/\varepsilon$ is independent of ε (see Section 4.3.3).

The last task is to estimate the modeling error. Here structure assumptions on the tensor such as local periodicity or random stationarity are required. For deterministic tensor we assume

(H3) $\quad a^{\varepsilon}(x) = a(x, x/\varepsilon) = a(x, y)$ is Y-periodic in y,
$a_{ij}(x, y) \in \mathcal{C}\left(\bar{\Omega}; W^{1,\infty}_{per}(Y)\right)$, for all $i, j = 1, \ldots, d$,

where we set $Y = (0, 1)^d$ for simplicity. For such a tensor the variables x and y are usually referred to as slow and fast variables, respectively. We first mention the following result obtained in [16].

Lemma 8. *Assume (***H3***) and micro problems* (4.10) *solved in $S^q(K_{\delta_j}, \mathcal{T}_h) \subset W^1_{per}(K_{\delta_j})$ with $\delta/\varepsilon \in \mathbb{N}$. Assume further that the slow variable tensor $a(x, x/\varepsilon)$ is collocated at the quadrature points x_{K_j}, i.e., $a(x_{K_j}, x/\varepsilon)$ is used in the problem* (4.10) *and in the bilinear form* (4.6)*. Then,* $e_{MOD} = 0$.

A consequence of this result is that the FE-HMM converges with a robust rate (i.e. independent of ε) to the homogenized solution In a more general situation, when for example the size of the period is unknown, a modeling error due to a mismatch of the sampling domain size and ε (e.g., $\delta/\varepsilon \notin \mathbb{N}$) and to artificial boundary conditions arise. This error is often called cell resonance in the literature (see e.g., [89]). The following result has been obtained in [68] (see also [143]).

Theorem 9 (**Modeling error**)**.** *Assume (***H3***) and that the micro problems* (4.10) *are solved in $S^q(K_{\delta_j},\mathcal{T}_h) \subset H_0^1(K_{\delta_j})$ with $\delta > \varepsilon$. Then*

$$e_{MOD} \leq C\left(\frac{\varepsilon}{\delta} + \delta\right),$$

where C is independent of H, h and ε.

We conclude this section by mentioning that recently, a new approach to cell resonance has been proposed in [82]. By modifying the cell problem (4.10) and adding a zeroth order term, one gets a micro problem with better decay of the error due to artificial boundary conditions (due to a faster decaying Green function associated with the problem). The bias introduced by modifying the cell problem can be controlled by tuning the constant associated with the zeroth order term. A rate of $\left(\frac{\varepsilon}{\delta}\right)^p$, $p < 4$ is reported in [82]. Numerical experiments however show that this asymptotic rate is only obtained for $\delta \gg \varepsilon$. Nevertheless, this constitutes a promising approach for the issue of cell resonance error.

Elliptic problems with random coefficients. The FE-HMM for elliptic problems with random conductivity tensors has been investigated in [68], where the modeling error e_{MOD} has been analyzed under a stationarity assumption. Convergence rates have been derived for dimensions $d = 1$ and $d = 3$ under the additional assumption that the random tensors satisfy a uniform mixing condition [144].

4.4.2 Example of a priori analysis for nonlinear problems

We consider a class of nonlinear nonmonotone multiscale problems described in Example 2

$$-\nabla \cdot (a^\varepsilon(x, u^\varepsilon(x)))\nabla u^\varepsilon(x)) = f(x) \text{ in } \Omega, \qquad u^\varepsilon(x) = 0 \text{ on } \partial\Omega, \tag{4.42}$$

with a $d \times d$ tensor $a^\varepsilon(x,s)$ satisfying (4.2) with entries $a^\varepsilon_{ij}(x,s)$ that are continuous functions on $\overline{\Omega} \times \mathbb{R}$ and uniformly Lipschitz continuous with respect to s and ε, i.e.,

$$\begin{aligned} \exists \Lambda_1 > 0, \quad |a^\varepsilon_{ij}(x,s_1) - a^\varepsilon_{ij}(x,s_2)| \leq \Lambda_1 |s_1 - s_2|, \\ \forall \varepsilon, \forall x \in \overline{\Omega}, \forall s_1, s_2 \in \mathbb{R}, \forall\ 1 \leq i,j \leq d. \end{aligned} \tag{4.43}$$

For such problems, it is known (see e.g., [37]) that there exists a subsequence of $\{a^\varepsilon(\cdot, s)\}$ (again indexed by ε) such that the corresponding sequence of solutions $\{u^\varepsilon\}$ converges weakly to u^0 in $H^1(\Omega)$, where u^0 is solution of the so-called homogenized problem

$$-\nabla \cdot \left(a^0(x, u^0(x))\nabla u^0(x)\right) = f(x) \text{ in } \Omega, \qquad u^0(x) = 0 \text{ on } \partial\Omega, \tag{4.44}$$

with a homogenized tensor $a^0(x,s)$ which can be shown to have similar properties as assumed for $a^\varepsilon(x,s)$. The numerical method for computing an effective solution of (4.42) has been described in (4.12). The coerciveness of the bilinear form $B_H(z^H;\cdot,\cdot)$ can be established similarly as in the linear case. Then, the existence of a solution of problem (4.12) relies on the Brouwer fixed point theorem applied to the nonlinear map $S_H : S_0^\ell(\Omega,\mathcal{T}_H) \to S_0^\ell(\Omega,\mathcal{T}_H)$ defined for $z_H \in S_0^\ell(\Omega,\mathcal{T}_H)$ as the solution $S_H z_H$ of the linear problem

$$B_H(z_H; S_H z_H, w_H) = \int_\Omega f w_H dx,\ \forall w_H \in S_0^\ell(\Omega,\mathcal{T}_H). \tag{4.45}$$

We refer for example to [91] for details. In contrast the uniqueness is much more involved (see below). For deriving a priori error estimates and rates of convergence for the error $\|u^0 - u_H\|_{H^1(\Omega)}$, we follow the methodology described in the linear case. There is however one main difference: a priori error analysis for FEMs with numerical quadrature such as the results in [50] used in the linear case need to be derived. Such results, that are quite technical in the nonlinear nonmonotone case, have been derived in [18, 20]. Using the results in [20] it is then possible to show (see [20] for a proof)

Theorem 10. *Let u^0, u_H be the solutions of problems* (4.44) *and* (4.12)*, respectively. Let $\ell \geq 1$. Let $\mu = 0$ or 1. Assume (*$\mathbf{Q1}$*), (*$\mathbf{Q2}$*) and (*$\mathbf{H1}$*)* [9] *hold. In addition, assume that $u^0 \in W^{1,\infty}(\Omega)$, $\partial_u a^0_{mn} \in W^{1,\infty}(\Omega \times \mathbb{R})$, and that the coefficients $a^0_{mn}(x,s)$ are twice differentiable with respect to s, with the first and second order derivatives continuous and bounded on $\overline{\Omega} \times \mathbb{R}$, for all $m,n = 1\ldots d$.*

Then, there exist $r_0 > 0$ and $H_0 > 0$ such that, provided

$$H \leq H_0 \quad \textit{and} \quad e(HMM) \leq r_0, \tag{4.46}$$

any solution u_H of (4.12) *satisfies*

$$\begin{aligned} \|u^0 - u_H\|_{H^1(\Omega)} &\leq C(H^\ell + e(HMM)) \quad \textit{if } \mu = 0,1, & (4.47)\\ \|u^0 - u_H\|_{L^2(\Omega)} &\leq C(H^{\ell+1} + e(HMM)) \quad \textit{if } \mu = 1 \textit{ holds}, & (4.48)\end{aligned}$$

where $e(HMM)$ is given in (4.49)*. Here, the constants C are independent of H and $e(HMM)$.*

Remark 11. *Notice that we assume that the macroscopic family of triangulations $\{\mathcal{T}_H\}$ satisfies the inverse assumption $\frac{H}{H_K} \leq C$ for all $K \in \mathcal{T}_H$ and all $\mathcal{T}_H$ in the above theorem (such a condition is often used for the FE analysis of nonlinear problems [141]). We also assume (for simplicity) the convexity of the domain Ω for the L^2 bound. Such a condition on the domain allows to infer suitable regularity to use an Aubin-Nitsche type duality argument. A weaker condition on the adjoint of the linearized problem associated to* (4.44) *(see [18, Equ. (5)]) is sufficient. This condition is usually assumed for L^2 estimates of linear FEM with numerical quadrature (see [125, Equ. (2.11)]).*

[9] Here $a^0_{ij} \in W^{\ell+\mu,\infty}(\Omega \times \mathbb{R})$ for $i,j = 1,\ldots,d$.

The proof of Theorem 10 is more involved than the similar theorem in the linear case. We note that the additional assumptions compared to the linear case (Theorem 5), i.e., the $W^{1,\infty}$ assumption on u^0, the smoothness of $s \mapsto a(x,s)$, are already used for the analysis of FEM for single scale nonlinear problems of the type (4.44) (see for example [91]). The inverse assumption is also used in [91] for the proof of the uniqueness of a numerical solution. The quantity $e(HMM)$ is defined by

$$e(HMM) := \sup_{K\in\mathcal{T}_H, x_{K_j}\in K, s\in\mathbb{R}} \|a^0(x_{K_j},s) - a^0_{K_j}(s)\|_F, \tag{4.49}$$

where a^0 is the homogenized tensor of (4.44) and $a^0_{K_j}$ is the tensor defined similarly as (4.28) by replacing $a^\varepsilon(x)$ with the nonlinear tensor $a^\varepsilon(x,s)$ (s playing here the role of a parameter), ψ^i_{h,K_j} with $\psi^{i,s}_{h,K_j}$ and $J_{\psi_{h,K_j}}(x)$ with $J_{\psi^s_{h,K_j}}(x)$. A nice feature of the analysis in [17]) is that micro and modeling errors can be analyzed, thank to Theorem 10, following the analysis obtained for the linear case (estimates with similar rate of convergence can be derived, see [17, Thms. 3.6,3.7].) The proof of the uniqueness of a numerical solution of Problem (4.12) is quite involved but can be derived without any structure assumption on the oscillation provided

$$s\in\mathbb{R}\mapsto a^\varepsilon(\cdot,s)\in(W^{1,\infty}(\Omega))^{d\times d} \text{ is of class } C^2 \text{ and } |\partial^k_u a^\varepsilon(\cdot,s)|_{W^{1,\infty}(\Omega)} \leq C\varepsilon^{-1},\ k\leq 2, \tag{4.50}$$

where C is independent of s and ε (see [17, Thm. 3.3]). Here we present a simpler uniqueness result that can be stated solely in terms of the size of the macro and micromeshes (see [17, Corollary 3.4])

Theorem 12. *In addition to the hypothesis of Theorem 10 assume (***H2***) and (***H3'***) and* (4.50). *Assume $W(K_{\delta_j}) = W^1_{per}(K_{\delta_j})$ (periodic coupling conditions), $\delta/\varepsilon\in\mathbb{N}$ and that the slow variable of the tensor $a(x,x/\varepsilon,s)$ is collocated at the quadrature points x_{K_j} in the problem* (4.14) *and in the form* (4.13). *Then, there exists a positive constant H_0 such that for all*

$$(h/\varepsilon)^{2q} \leq H \leq H_0,$$

the solution u_H of (4.12) *is unique.*

Here (**H3'**) is defined similarly as (**H3**) for the nonlinear tensor $a^\varepsilon_{ij}(x,s) = a_{ij}(x,x/\varepsilon,s)$, where $a_{ij}(x,y,s)$ is Y-periodic in y, and the map $(x,s)\mapsto a_{ij}(x,\cdot,s)$ is Lipschitz continuous and bounded from $\overline{\Omega}\times\mathbb{R}$ into $W^{1,\infty}_{per}(Y)$.

4.4.3 A posteriori error analysis

The goal of the residual based a posteriori analysis is to give upper and lower bounds of the error $e_H := u^0 - u_H$ in the H^1 norm for a given solution u_H of the problem (4.5) in terms of residual and data approximations, $\eta_H(\Omega), \xi_H(\Omega)$ defined in (4.19) and (4.20), respectively. As mentioned in Section 4.3.3 (see Example 3), we will concentrate here on piecewise linear simplicial macro FEs. Compared to classical residual based a posteriori error analysis, we have here two additional difficulties

- the data of the effective problem (4.3) (entering in the residual η_H) are not available beforehand;
- "variational crimes" (see [39]) are committed by replacing the true bilinear form by the bilinear form of the FE-HMM.

The first issue can be dealt with by introducing the jump of multiscale fluxes for the modeling of (4.19). For the second issue, the following representation formula derived in [14, 12] is crucial. Let $B_0(\cdot,\cdot)$ be the bilinear form corresponding to the variational formulation of (4.3). Then, we can relate the bilinear form $B_0(\cdot,\cdot)$ evaluated at e_H to the actual numerical solution u_H, the multiscale flux and data approximation as follows. Here for simplicity we consider piecewise linear macroscopic FE functions.

Lemma 13 (Error representation formula)**.** *For all $v \in H_0^1(\Omega)$, we have*

$$B_0(e_H,v) = \int_\Omega fvdx - \sum_{e\in\mathcal{E}_H} \int_e \overline{[\![a^\varepsilon\left(x\right)\nabla u_h]\!]}_e v ds + \sum_{K\in\mathcal{T}_H}\int_K \left(a_K^0(x_K) - a^0\left(x\right)\right)\nabla u_H \cdot \nabla v dx, \quad (4.51)$$

where $\mathcal{E}_H$ denotes the set of interfaces of the conformal mesh $\mathcal{T}_H$, u_H is the solution of (4.5) *and where the jump of multiscale fluxes $\overline{[\![a^\varepsilon\left(x\right)\nabla u_h]\!]}_e$ is defined in (4.18).*

Using the above formula we obtain for $v = u^0 - u_H$

$$\begin{aligned} B_0\left(e_H,e_H\right) &= \int_\Omega f_H\left(e_H - I_H e_H\right)dx + \int_\Omega \left(f - f_H\right)\left(e_H - I_H e_H\right)dx \\ &- \sum_{e\in\mathcal{E}_H}\int_e \overline{[\![a^\varepsilon\left(x\right)\nabla u_h]\!]}_e \left(e_H - I_H e_H\right)ds \\ &+ \sum_{K\in\mathcal{T}_H}\int_K \left(a_K^0(x_K) - a^0\left(x\right)\right)\nabla u_H\cdot\nabla e_H\,dx, \end{aligned}$$

where $a_K^0(x_K)$ is defined in (4.28), I_H denotes the Clément interpolation operator (see [52]). It is a linear operator $I_H : H^1\left(\Omega\right) \to S(\Omega,\mathcal{T}_H)$ having the property that for all $v \in H^1\left(\Omega\right)$ and $K \in \mathcal{T}_H$

$$\left\|v - I_H v\right\|_{L^2(K)} \le C H_K \left\|\nabla v\right\|_{L^2(N(K))} \quad (4.52)$$

and

$$\left\|\nabla\left(v - I_H v\right)\right\|_{L^2(K)} \le C\left\|\nabla v\right\|_{L^2(N(K))}, \quad (4.53)$$

where $N\left(K\right)$ is a neighborhood of K that consists of all elements of $\mathcal{T}_H$ which have a non-empty intersection with K. Using the properties of the Clément interpolation operator, Cauchy-Schwarz and Poincaré inequalities, the coercivity of $B_0(\cdot,\cdot)$ and the finite overlapping property of the neighborhoods $N\left(K\right)$, we can derive (see [14] for a complete proof)

Theorem 14 (Upper bound). *Let u^0, u_H be the solutions of problems* (4.3) *and* (4.5)*, respectively. Then*

$$\left\|u^0 - u_H\right\|^2_{H^1(\Omega)} \leq C\left(\eta_H\left(\Omega\right)^2 + \xi_H\left(\Omega\right)^2\right),$$

where C only depends on the shape regularity constant, the coercivity and continuity bounds (4.2), the dimension d and the constant of the Poincaré inequality.

To derive a lower bound, bubble functions can be used as in the classical theory [138]. We recall the construction of such functions. Let $\widetilde{\mathcal{T}}_H$ be a refinement of $\mathcal{T}_H$ such that every $K \in \mathcal{T}_H$ has an interior node $\tilde{x}_K \in K\backslash\partial K$ in $\widetilde{\mathcal{T}}_H$, and every edge e of $\mathcal{T}_H$ not on the boundary $\partial\Omega$ has an interior node $\tilde{x}_e \in e\backslash\partial e$ in $\widetilde{\mathcal{T}}_H$. Based on the triangulation $\widetilde{\mathcal{T}}_H$ we define a piecewise linear FE space $S^1_0(\Omega, \widetilde{\mathcal{T}}_H) \supset S^1_0(\Omega, \mathcal{T}_H)$.

We start by estimating the so-called interior residual. For any $K \in \mathcal{T}_H$ consider an interior bubble function, i.e., a function $\psi_K \in S^1_0(\Omega, \widetilde{\mathcal{T}}_H)$ such that $0 \leq \psi_K \leq 1$, $\psi_K\left(\tilde{x}_K\right) = 1$ and $\psi_K \equiv 0$ on $\Omega\backslash K$. Using the representation formula (4.51) with a test function given by $v := \psi_K f_H \in H^1_0\left(\Omega\right)$ one can estimate the so-called interior residual

$$\begin{aligned} H^2_K \left\|f_H\right\|^2_{L^2(K)} \leq & C\left(\left\|\nabla e_H\right\|^2_{L^2(K)} + H^2_K \left\|f - f_H\right\|^2_{L^2(K)}\right. \\ & \left. + \left\|\left(a^0_K(x_K) - a^0\left(x\right)\right)\nabla u_H\right\|^2_{L^2(K)}\right). \end{aligned} \tag{4.54}$$

Next, we estimate the so-called jump residual. Let $e \in \mathcal{E}_H$ be an interior interface and let $K_1 \in \mathcal{T}_H$ and $K_2 \in \mathcal{T}_H$ be such that $K_1 \cap K_2 = e$. Furthermore, let $\tilde{x}_e \in e$ be an interior node and $\psi_e \in S^1_0(\Omega, \widetilde{\mathcal{T}}_H)$ a bubble function such that $\psi_e\left(\tilde{x}_e\right) = 1$ and $\psi_e \equiv 0$ on $\Omega\backslash(K_1 \cup K_2)$. Using the representation formula (4.51) with $v := \psi_e$ we find

$$\begin{aligned} H_e \left\|[\![\overline{a^\varepsilon\left(x\right)\nabla u_h}]\!]_e\right\|^2_{L^2(e)} \leq C & \left(\left\|\nabla e_H\right\|^2_{L^2(\omega_e)} + H^2_{\omega_e}\left\|f - f_H\right\|^2_{L^2(\omega_e)}\right. \\ & \left. + \left\|\left(a^0_K(x_K) - a^0\left(x\right)\right)\nabla u_H\right\|^2_{L^2(\omega_e)}\right), \end{aligned} \tag{4.55}$$

where $H_{\omega_e} = \max_{i=1,2} H_i$, $\omega_e = K_1 \cup K_2$. Using the estimates (4.54),(4.55) on the interior and jump residuals, respectively, allows to derive the following lower bound (see [14] for a complete proof)

Theorem 15 (Lower bound). *Let u^0, u_H be the solutions of problems* (4.3) *and* (4.5)*, respectively. Denote by ω_K the domain which consist of all elements sharing at least one side with K. Then*

$$\eta_H\left(K\right)^2 \leq C\left(\left\|u^0 - u_H\right\|^2_{H^1(\omega_K)} + \xi_H\left(\omega_K\right)^2\right),$$

where C only depends on the shape regularity constant, the coercivity and continuity bounds (4.2), the dimension d and the constant of the Poincaré inequality.

5 Finite volume methods

5.1 The algorithm

In this section we discuss the application of HMM in the setting of finite volume methods. Finite volume methods are generally preferred when the macroscopic models are in the form of conservation laws. In fluid and solid mechanics, the macroscale models are often derived from the conservation laws of mass, momentum and energy:

$$\left\{ \begin{array}{rcl} \partial_t \rho + \nabla \cdot q & = & 0, \\ \partial_t q + \nabla \cdot \tau & = & 0, \\ \partial_t E + \nabla \cdot J & = & 0. \end{array} \right. \tag{5.1}$$

Here ρ is the mass density of the system, $q = \rho v$ is the momentum density with v being the velocity, and E is the total energy density. τ and J are the momentum and energy fluxes respectively. In conventional continuum models, empirical relations are sought to express these fluxes in terms of the conserved densities. In HMM, the fluxes are computed directly from some underlying microscopic models.

An example of the microscopic model that one can consider is that of molecular dynamics, i.e. Newton's equations of motion for the constituting atoms,

$$\left\{ \begin{array}{rcl} \dot{x}_i & = & v_i, \\ m_i \dot{v}_i & = & \displaystyle\sum_{j \neq i} f\big(x_i(t) - x_j(t)\big). \end{array} \right. \tag{5.2}$$

Here m_i is the mass of the i-th particle and $f(r) = -\nabla \phi(r)$ is the force exerted on the particles with interatomic potential $\phi(r)$. Here we have assumed that the particles interact via a pair-wise potential even though the algorithms discussed here can be easily adapted to the situation when general inter-atomic potentials are used.

To see how the microscopic model can be linked with the macroscopic model, observe that for (5.2), we can also write down the analogs of the equations (5.1) by defining,

$$\left\{ \begin{array}{rcl} \widetilde{\rho}(x,t) & = & \displaystyle\sum_i m_i \delta\big(x - x_i(t)\big), \\ \widetilde{q}(x,t) & = & \displaystyle\sum_i m_i v_i(t) \delta\big(x - x_i(t)\big), \\ \widetilde{E}(x,t) & = & \displaystyle\frac{1}{2}\sum_i \Big[m_i |v_i|^2 + \sum_{j \neq i} \phi\big(x_i(t) - x_j(t)\big)\Big] \delta\big(x - x_i(t)\big), \end{array} \right. \tag{5.3}$$

and

$$\begin{aligned}\widetilde{\tau}(x,t) = & \sum_i m_i v_i \otimes v_i \delta\big(x - x_i(t)\big) \\ & + \frac{1}{2}\sum_i \sum_{j\neq i} f\big(x_i - x_j\big) \otimes (x_i - x_j) \\ & \times \int_0^1 \delta\Big(x - \big(x_j + \lambda(x_i - x_j)\big)\Big) d\lambda \end{aligned} \tag{5.4a}$$

$$\begin{aligned}\widetilde{J}(x,t) = & \sum_i v_i \Big[\frac{1}{2} m_i |v_i|^2 + \frac{1}{2}\sum_{j\neq i} \phi\big(x_i(t) - x_j(t)\big)\Big] \delta\big(x - x_i(t)\big) \\ & + \frac{1}{4}\sum_{j\neq i} \big(v_j(t) + v_i(t)\big) \cdot f\big(x_j(t) - x_i(t)\big)(x_i - x_j) \\ & \times \int_0^1 \delta\Big(x - \big(x_j + \lambda(x_i - x_j)\big)\Big) d\lambda. \end{aligned} \tag{5.4b}$$

One can easily verify that, from (5.2), we have

$$\left\{ \begin{array}{rcl} \partial_t \widetilde{\rho} + \nabla_{\mathbf{x}} \cdot \widetilde{q} & = & 0, \\ \partial_t \widetilde{q} + \nabla_{\mathbf{x}} \cdot \widetilde{\tau} & = & 0, \\ \partial_t \widetilde{E} + \nabla_{\mathbf{x}} \cdot \widetilde{J} & = & 0. \end{array} \right. \tag{5.5}$$

An important factor to be considered is the scale separation: microscopic processes usually take place at the length scale of nanometers (10^{-9}m) and time scale of femto- or pico-seconds on which one can neglect the variation of the macroscale variables. Locally, one can then think of the atomistic system as been constrained by the local macroscopic quantities, namely the local mass, momentum and energy densities. In addition, the atomistic system will stay close to local equilibrium since there is sufficient time for relaxation to take place. Consequently, one may view the models (5.1) as being the ensemble averages of (5.2). This is the ideal situation for HMM to be used.

A HMM strategy for this setting has been developed in [103, 126]. Our presentation here follows that of [103].

Macroscale solver. Since the macroscopic model (5.1) is in the form of conservation laws, it is natural to choose as the macroscale solver a finite volume method. Although there are many different finite volume methods that are available for conservation laws e.g. [100, 83], many involve the computation of the Jacobian for the flux functions. These are less suited for the present problem since the flux function is not explicitly given to us. An exception is the central scheme of Lax-Friedrichs type, such as [114], which is formulated over a staggered-grid. As it turns out, this method can be easily coupled with molecular dynamics.

To be more specific, we first rewrite the conservation laws in a generic form,

$$\partial_t u + \partial_x f = 0, \tag{5.6}$$

We will confine our discussion to macroscopically one dimensional problems since the extension to higher dimension is straightforward. The first order central scheme represents the solutions by piece-wise constants, which are the average values over each cell:

$$u_k^n = \frac{1}{\Delta x} \int_{x_{k-1/2}}^{x_{k+1/2}} u(x, t^n) dx.$$

Time integration over $[x_j,\ x_{j+1}] \times [t^n,\ t^{n+1})$ leads to the following scheme,

$$u_{k+1/2}^{n+1} = \frac{u_k^n + u_{k+1}^n}{2} - \frac{\Delta t}{\Delta x}\big(f_{k+1}^n - f_k^n\big), \tag{5.7}$$

with numerical flux,

$$f_k^n = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} f(x_k, t) dt,$$

which is taken to be $f_k^n = f(x_k, t^n)$ in central scheme.

(5.1) is still incomplete since we still do not know the fluxes. Next we describe how this information can be extracted from MD or other atomistic models.

Microscale solver. We first discuss the case when the microscopic model is MD. At each point where numerical fluxes are needed, we perform a local MD simulation to estimate the fluxes. The atomistic simulation will be constrained by the local macroscopic variables, which are the local density, momentum and energy. To initialize the MD simulation, we first arrange a trial configuration of the atoms according to the local density. From the local total energy, momentum and potential energy V of the trial configuration, we can compute the thermal energy K_θ by

$$K_\theta = E - V - \frac{1}{2}\rho v^2,$$

If K_θ is negative, the trial configuration is rejected and another trial configuration with the same number of atoms is generated. Otherwise it is accepted, and a local initial temperature is computed from K_θ. The velocities of the atoms are then given by

$$v_i = v + v_i',$$

where v_i' is thermal velocity given by the Maxwell-Boltzmann distribution with the given local temperature.

The set of ODEs (5.2) can be solved by standard finite difference scheme such as Verlet's method. For the present analysis, however, we will assume that the ODEs are solved exactly to avoid unnecessary complication. After the MD system equilibrates, we obtain the needed quantities by averaging. Specifically let $\tilde{j}$ be the spatial average over the simulation box of a local flux whose expression was given in (5.4),

$$\tilde{J} = \tilde{J}(X), \quad X = (x_1, x_2, \cdots, x_N, v_1, v_2, \cdots, v_N), \tag{5.8}$$

then we obtain the corresponding macroscale quantity by time averaging,

$$J = \frac{1}{T}\int_{\tau}^{\tau+T} \tilde{J}\big(X(t)\big)dt, \tag{5.9}$$

where τ is the starting point when the time averaging begins, and T is the duration of the time averaging. An additional ensemble averaging can also be used if desired. For detailed discussion on the setup of the MD as well as boundary conditions, see [103].

The overall numerical procedure is shown schematically in Figure 18. At each time step, the scheme (5.7) requires as input the fluxes at grid point x_j to complete the time integration. These flux values are obtained by performing local MD simulations that are consistent with the local macroscale state (ρ, q, E). Once these values are computed, one can advance to the next macro time step using (5.7).

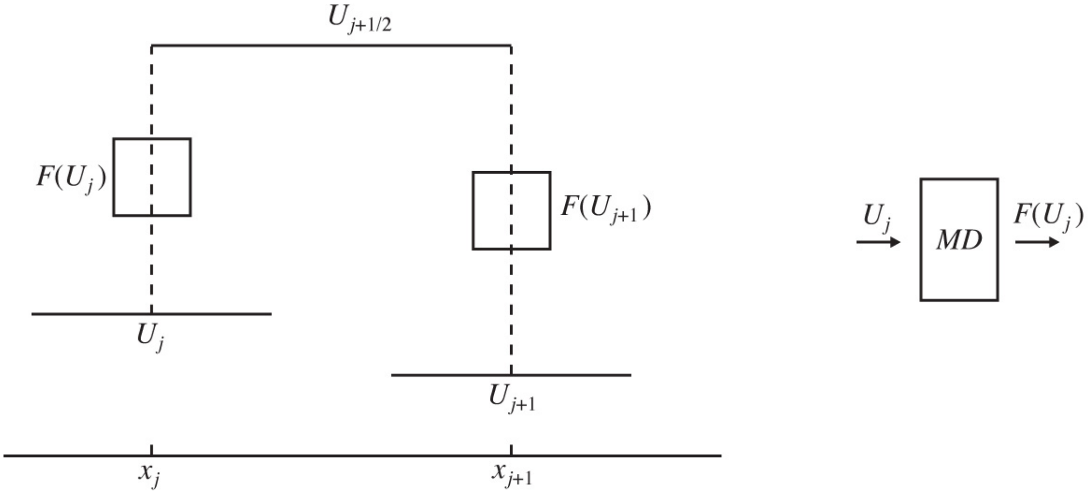


Figure 18: A schematic illustration of the numerical procedure: starting from piece-wise constant solutions $\mathbf{u}_k^n$, one integrates 5.6 in time and in the cell x_k, x_{k+1}. The time step Δt is chosen in such a way that the waves coming from $x_{k+1/2}$ will not reach x_k, and thus for $t \in [t^n, t^{n+1}]$, $\mathbf{u}(x_k, t) = u_k^n$. If $f(u)$ at x_k is found to be unknown, we perform a MD simulation using u_k^n to invoke and restrict the microscopic process. The needed flux is then extracted from the simulation and the integration is completed. Analogously one can embed the MD simulation to higher order macro-schemes or higher dimensions.

One result from such a method is shown in Figure 19. Here the setup for the macroscale model is a Riemann problem for one-dimensional wave propagation in solids. The result of HMM is compared with that of a direct MD simulation. The microscale model is MD with Lennard-Jones potential.

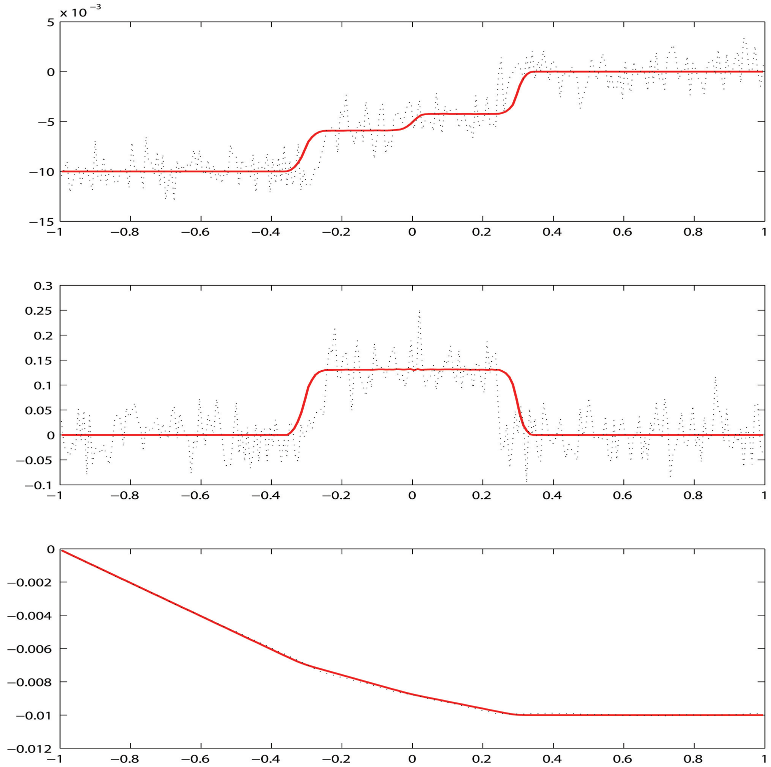


Figure 19: Numerical test on shock formation and propagation. 200 macro-grid points are used and each local MD simulation consists of 40×10 atoms and 10^4 steps of time integration. The solution is displayed after 40 steps of integration over macro time steps. Solid line: computed solution; dashed line: full atom simulation (one realization). Top: strain, middle: velocity, bottom: displacement.

5.2 Error analysis

As we have seen before, error estimates for HMM proceed in two steps. The first is to estimate the "HMM error", i.e. the the error in the estimated data. The second step is to obtain the overall error estimate.

We will take as the exact solution the solution for the case when the fluxes are given by the averaged quantities for the equilibrium distribution with parameters given by the local values of the macroscale variables (density, momentum, energy).

First, let us look at the HMM error. Let

$$e(\text{HMM}) = E\left[|J - \bar{J}|\right], \tag{5.10}$$

where J denotes the value of the flux computed by HMM, $\bar{J}$ is the exact value for the flux:

$$\bar{J} = \int \tilde{J}(X) d\mu_\infty, \tag{5.11}$$

with μ_∞ being the local equilibrium distribution.

Let L be the size of the microscale simulation box and d be the dimension of the problem. Denote by μ_L^t the particle distribution for the MD system at time t, and let μ_L be the equilibrium particle distribution for a system of size L. In addition, define two quantities $\alpha(L)$ and $\tau_0(L)$ by the following

$$\left|\int f(X)d(\mu_L - \mu_L^t)\right| \leq C\mathrm{e}^{-\frac{t}{\tau_0(L)}}\|f\|_{w,\infty}, \tag{A_1}$$

and

$$\left|\int f(X)d(\mu_L - \mu_\infty)\right| \leq C\alpha(L)\|f\|_{w,\infty}, \tag{A_2}$$

for any continuous function f that satisfies,

$$\|f\|_{w,\infty} \leq \infty,$$

where,

$$\|f\|_{w,\infty} = \sup_X \left(1 + |X|^2\right)^{-p/2}|f(X)|,$$

for $p \geq 2$. $\tau_0(L)$ can be regarded as the relaxation time for system of size L, and $\alpha(L)$ measures the error due to the finite size effect.

Now let

$$J_L^t = \int \tilde{J}(X)d\mu_L^t, \tag{5.12}$$

$$A(t,s) = E\Big[\big(\tilde{J}(t) - J_L^t\big)\big(\tilde{J}(s) - J_L^s\big)\Big], \tag{5.13}$$

where the expectation is w.r.t. μ_L^0, the initial distribution. We further assume that,

$$\int_0^{+\infty} A(t,s)ds \leq C_0,\ \forall t \geq 0. \tag{A_3}$$

This amounts to assuming that the correlation decays in time sufficiently fast.

With (A_1), (A_2), we have,

$$\left|\bar{J} - \frac{1}{T}\int_\tau^{\tau+T} J_L^t dt\right| \leq C(\alpha(L) + T^{-1}\mathrm{e}^{-\frac{t}{\tau_0(L)}}). \tag{5.14}$$

Combined with (A_3), we have,

$$\begin{aligned} E\left[\left(J - \frac{1}{T}\int_\tau^{\tau+T} J_L^t dt\right)^2\right] &= \frac{1}{T^2}E\left[\int_\tau^{\tau+T}\int_\tau^{\tau+T}(\tilde{J}(t) - J_L^t)(\tilde{J}(s) - J_L^s)dtds\right], \\ &\leq \frac{C_0}{T}, \end{aligned}$$

Hence, we have

$$e(\text{HMM}) \leq C\left(\alpha(L) + T^{-1}\mathrm{e}^{-\frac{\tau}{\tau_0(L)}} + \frac{1}{\sqrt{T}}\right). \tag{5.15}$$

If ensemble averaging is also used, for example, with M independent copies, the above estimate becomes,

$$e(\mathrm{HMM}) \leq C\left(\alpha(L) + T^{-1}\mathrm{e}^{-\frac{\tau}{\tau_0(L)}} + \frac{1}{\sqrt{TM}}\right). \tag{5.16}$$

We clearly see that $e(\mathrm{HMM})$ is controlled by the behavior of $\alpha(L)$ and $\tau_0(L)$. In general one expects that

$$\alpha(L) \sim L^{-d/2}, \tag{5.17}$$

from a central limit theorem type of argument, and

$$\tau_0(L) \sim L^r, \tag{5.18}$$

with $r > 0$.

From (5.17) and (5.18), we see the conflliciting effects of choosing the size of the MD system: when L is increased, the finite size error is reduced while the relaxation time τ_0 is increased. Similarly when L is decreased, τ_0 decreases but $\alpha(L)$ will increase.

The macroscopic models considered in this paper have the following properties: They are hyperbolic and they have a strictly convex entropy function. To avoid irrelevant complictions, we will further assume that the numerical solutions at the macroscale level are bounded and the exact solution is piecewise Lipschitz continuous.

Assume that (5.6) is solved by the three-point conservative scheme,

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\big(f_{j+1/2}^n - f_{j-1/2}^n\big), \tag{5.19}$$

where $f_{j+1/2} = F(u_j, u_{j+1})$ and we have omitted the superscript. For Lax-Friedrichs type of schemes, which are the ones considered in the following analysis, the numerical flux $f_{j+1/2}$ is given by,

$$F(u_j, u_{j+1}) = \frac{1}{2}\big(f(u_j) + f(u_{j+1})\big) - \frac{a}{2}\big(u_{j+1} - u_j\big). \tag{5.20}$$

The constant a is chosen to be bigger than the eigenvalues of ∇f

$$a \geq \max|\lambda(\nabla f)|,$$

to ensure stability. Since the time step and the grid size are always proportional in the scheme, we will use Δx to indicate the discretization error.

The HMM solution can be written as:

$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{\Delta t}{\Delta x}\big(g_{j+1/2} - g_{j-1/2}\big), && (5.21)\\ g_{j+1/2} &= G(u_j, u_{j+1}), && (5.22) \end{aligned}$$

with,

$$G(u_j, u_{j+1}) = \frac{1}{2}\big(g(u_j) + g(u_{j+1})\big) - \frac{a}{2}\big(u_{j+1} - u_j\big). \tag{5.23}$$

Define the intermediate variable:

$$w_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\big(h_{j+1/2} - h_{j-1/2}\big), \tag{5.24}$$

$$h_{j+1/2} = F(u_j, u_{j+1}) \tag{5.25}$$

then,

$$\|w^{n+1} - u^{n+1}\| \le Ce(\text{HMM}), \tag{5.26}$$

with some constant C. The distance between $\{w_j^{n+1}\}$ and $\{v_j^n\}$, however, is directly related to the stability of the numerical methods.

Theorem. Let v be the exact solution to the PDEs (5.6) and $u_{\text{HMM}} = \{u_j^n\}$, then we have

$$\big\|v - u_{\text{HMM}}\big\|_{L_2} \le C\Delta x + \sqrt{C\Delta x + C\frac{e(\text{HMM})}{\Delta x}}. \tag{5.27}$$

5.3 Application to spall fracture

Spall fractures, which, in its simplest form, occurs when two strong shock waves under strain condition interact to produce a region of tension in the interior of a material body. The interacting waves arise, for example, when the system is subject to high velocity impacts. The shock waves lead to deformation of material at the highest possible rate, and the inertial effects become extremely important.

Experimental studies on the fracture initiation due to rapid loadings have been well documented [23, 54]. However, computational approach remains a challenge due to the processes occurring on multiple physical scales.

This problem was studied by Xiantao Li using HMM, by integrating conventional methods for continuum models of solids, with an atomic level description – the molecular dynamics model. The coupling is shown in figure 20. This method treats the crack tip area by explicitly incorporating the atomic interactions, to describe the interaction with the shock waves. Meanwhile, the integration with continuum models allows one to capture shock waves, and simulate materials of realistic size.

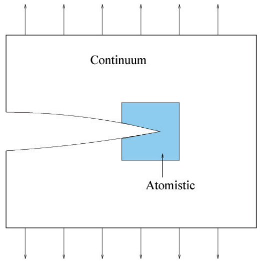


Figure 20: A coupled model.

In the continuum region, one solves the conventional computational model of the elastodynamics equation,

$$\rho_0 \frac{\partial u}{\partial t} = \nabla \cdot \sigma. \tag{5.28}$$

Here ρ_0 is the initial density, σ is the stress tensor.

The model is often combined with Griffith type of criteria to simulate crack growth, e.g. see [43, 142]. However, many of the fundamental properties of cracks are determined by the detailed atomic interactions. Therefore, one introduces an atomistic region around the crack, where the system is modeled by molecular dynamics (MD).

In [101]. Roe's scheme was used as the macro-solver away from the crack tip. Around the crack tip, numerical fluxes were evaluated via the help of MD. When the finite volume method is applied to the microscale model, the average momentum in a cell is determined by the total traction along the cell edges, which can be written as,

$$t = \sum_{i,j} f_{ij}. \tag{5.29}$$

Here f_{ij} is the force between two atoms that lie on two sides of the interface. Such a coupling method maintains the continuity of the fluxes across the interface. The coupling scheme is illustrated in the Fig. 20.

Another critical issue is the boundary condition for the atomistic region. This issue has been extensively studied in [102, 105, 104].

Numerical results. As the first application, Xiantao Li studied cracks in pure iron material at low temperature. The interatomic potential is the EAM model [55]. The system is a rectangular sample with size $0.78\mu m \times 0.78\mu m$. The condition plain strain was assumed. For the MD model, this is enforced by a periodic boundary condition in the third direction with period equal to one atomic spacing. In the first experiment, tensile stress is rapidly applied from the top and bottom boundary of the specimen. This generates two shock waves, which later propagate to the center of the system to interact with a pre-existing crack. Below certain threshold, no crack growth is observed after the two shocks collide. Instead of shock collisions, reflections off the crack faces are observed behind the crack tip. Above certain threshold, one begins to observe crack growth, followed by a sequence of high frequency lattice waves, the first two of which are of elliptical and diamond shapes. Experimental observation shows that the spallation does not occur instantaneously when the tension exceeds the spallation strength. It occurs after a brief incubation period. This was also observed in our numerical experiments. A sequence of snapshots are shown in Fig. 21.

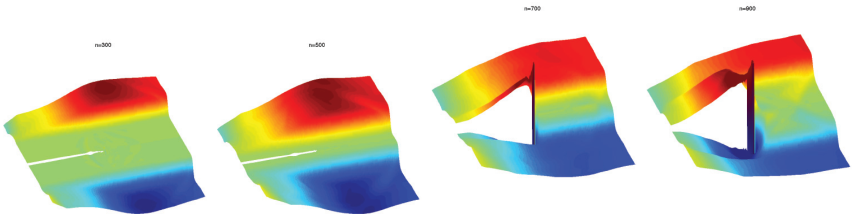


Figure 21: An open crack under mode I shock loading.

In the second experiment, Li studied the response of a crack to loadings along the crack faces. These results are shown in Fig. 22. In this case, the shocks are no longer of longitudinal type, and they propagate with much slow speed. The magnitude required to produce fractures is much higher.

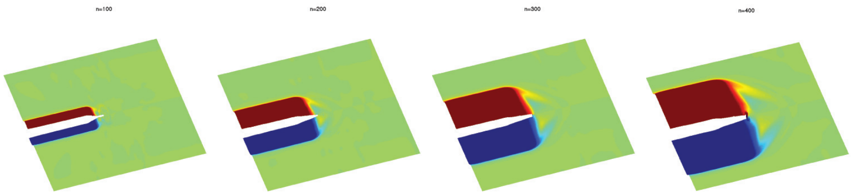


Figure 22: An open crack subject to face loading.

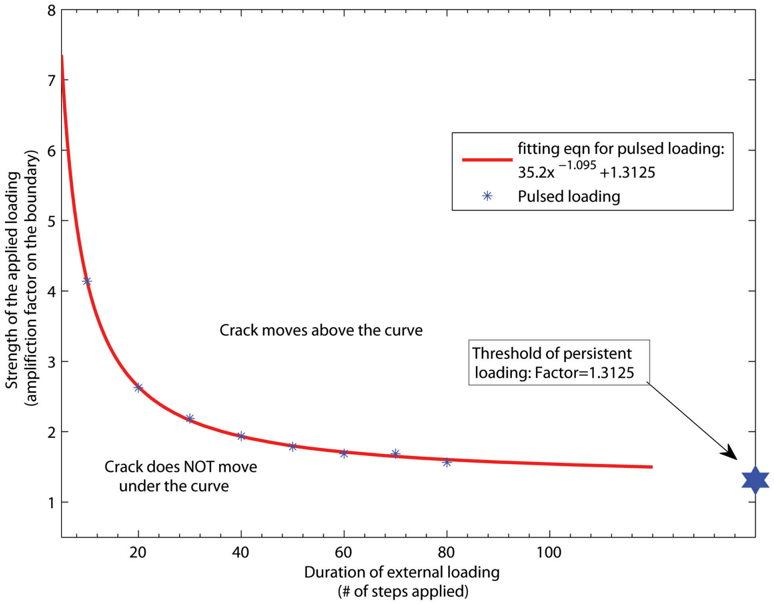


Figure 23: The onset of crack initiation.

Figure 23 summarizes the results under the impact of shocks of different magnitude. The goal is to investigate the strength of the material under various type of extreme loading conditions. The curve in the figure indicates the onset of the crack growth. In particular, for a given impact duration, the point on the curve indicates the magnitude for which the shock impact leads to fracture. For instantaneous shock loads, only very strong shocks are able to produce a growing crack. For continuing (slow) impacts, the threshold is much lower, and the curve approaches to a plateau, which agrees with linear elasticity analysis. The results can be compared to the classical experimental results [124].

6 Conclusion

When designing multiscale methods, there are two important factors to consider:

1. From a modeling viewpoint, one would like to eliminate ad hoc modeling assumptions as much as possible.

2. From a numerical viewpoint, the algorithms have to be practical, and satisfy the standard requirements for numerical algorithms such as stability and accuracy.

These two issues are not always consistent with each other. Therefore compromises have to be made. This is one lesson that we have learned in the past decade in multiscale modeling. HMM provides a general framework that allows us to make such compromises and make progress.

Although HMM has shown a lot of promises, much work remains to be done in order to fully realize its potential. One has to note though that serious obstacles remain in order to apply HMM to more challenging problems. For type A problems, i.e. problems involving local defects or singularities, the macro states around the defects or singularities have to be inhomogeneous, and there are difficulities in formulating constrained microscopic models in order for them to be consistent with inhomogeneous macro states. In addition, near defects or singularities, the local macroscale solutions are often singular also. Therefore for numerical efficiency, local mesh refinements are needed. This is a technical problem that need to be addressed. For type B problems, i.e. problems for which microscopic models are needed to supplement constitutive relations, the main obstacle is still the cost associated in running the microscopic models.

A major issue for multiscale methods is the effect of fluctuations. This is manifested in several ways. Fluctuation effects are almost always present in microscopic models such as molecular dynamics or Monte Carlo models. They are also present in detailed models of random media. However, they are absent in most effective macroscale models. In cases when the fluctuation effects are important, these effective macroscale models have to be remedied to take into account the fluctuations [57]. How to do this is still very much of an open question. In particular, in HMM, in order to save cost, one would like to perform the microscopic simulations on as small domains as possible. But central limit theorem tells us that fluctuations are inversely proportional to the square root of the volume. So the computational savings might come at the cost of artificially enlarge the size of the fluctuations [33]. These are all issues that will need to be carefully addressed.

Acknowledgement: We are grateful to Xiantao Li for his help in writing this article. The work of Abdulle is supported in part by the Swiss National Science Foundation under Grant 200021 134716/1. The work of E has been supported in part by ONR grant N00014-01-1-0674. The work of Engquist has been supported in part by NSF grant DMS1016577. The work of Vanden-Eijnden is supported in part by NSF grant DMS-0708140 and ONR grant N00014-11-1-0345.

References

[1] A. Abdulle. Multiscale methods for advection-diffusion problems. *Discrete Contin. Dyn. Syst.*, suppl.:11–21, 2005.

[2] A. Abdulle. On a priori error analysis of fully discrete heterogeneous multiscale FEM. *SIAM, Multiscale Model. Simul.*, 4(2):447–459, 2005.

[3] A. Abdulle. Analysis of a heterogeneous multiscale FEM for problems in elasticity. *Math. Models Methods Appl. Sci.*, 16(4):615–635, 2006.

[4] A. Abdulle. Multiscale method based on discontinous galerkin methods for homogenization problems. *C. R. Math. Acad. Sci. Paris*, 346(1-2):97–102, 2008.

[5] A. Abdulle. The finite element heterogeneous multiscale method: a computational strategy for multiscale pdes. *GAKUTO Int. Ser. Math. Sci. Appl.*, 31:135–184, 2009.

[6] A. Abdulle. A priori and a posteriori error analysis for numerical homogenization: a unified framework. *Ser. Contemp. Appl. Math. CAM*, 16:280–305, 2011.

[7] A. Abdulle. Discontinuous Galerkin finite element heterogeneous multiscale method for elliptic problems with multiple scales. *Math. Comp.*, 81(278):687–713, 2012.

[8] A. Abdulle and Y. Bai. Reduced basis finite element heterogeneous multiscale method for high-order discretizations of elliptic homogenization problems. *To appear in J. Comput. Phys.*, 2011.

[9] A. Abdulle and W. E. Finite difference heterogeneous multi-scale method for homogenization problems. *J. Comp. Phys.*, 191:18–39, 2003.

[10] A. Abdulle and B. Engquist. Finite element heterogeneous multiscale methods with near optimal computational complexity. *SIAM, Multiscale Model. Simul.*, 6(4):1059–1084, 2007.

[11] A. Abdulle and M. Grote. Finite element heterogeneous multiscale method for the wave equation. *SIAM, Multiscale Model. Simul.*, 9(2):766–792, 2011.

[12] A. Abdulle and A. Nonnenmacher. A posteriori error analysis of the heterogeneous multiscale method for homogenization problems. *C. R. Math. Acad. Sci. Paris*, 347(17-18):1081–1086, 2009.

[13] A. Abdulle and A. Nonnenmacher. A short and versatile finite element multiscale code for homogenization problem. *Comput. Methods Appl. Mech. Engrg*, 198(37-40):2839–2859, 2009.

[14] A. Abdulle and A. Nonnenmacher. Adaptive finite element heterogeneous multiscale method for homogenization problems. *Comput. Methods Appl. Mech. Engrg.*, 200(37-40):2710–2726, 2011.

[15] A. Abdulle and A. Nonnenmacher. A posteriori error estimate in quantities of interest for the finite element heterogeneous multiscale method. *Preprint*, submitted for publication, 2011.

[16] A. Abdulle and C. Schwab. Heterogeneous multiscale fem for diffusion problems on rough surfaces. *SIAM, Multiscale Model. Simul.*, 3(1):195–220, 2005.

[17] A. Abdulle and G. Vilmart. Analysis of the finite element heterogeneous multiscale method for nonmonotone elliptic homogenization problems. *Preprint*, submitted for publication, 2011.

[18] A. Abdulle and G. Vilmart. The effect of numerical integration in the finite element method for nonmonotone nonlinear elliptic problems with application to numerical homogenization methods. *C. R. Acad. Sci. Paris, Ser. I*, 349(19-20):1041–1046, 2011.

[19] A. Abdulle and G. Vilmart. Coupling heterogeneous multiscale FEM with Runge-Kutta methods for parabolic homogenization problems: a fully discrete space-time analysis. *To appear in Math. Models Methods Appl. Sci.*, 2012.

[20] A. Abdulle and G. Vilmart. A priori error estimates for finite element methods with numerical quadrature for nonmonotone nonlinear elliptic problems. *To appear in Num. Math.*, 2012.

[21] C.F. Abrams and E. Vanden-Eijnden. Large-scale conformational sampling of proteins using temperature-accelerated molecular dynamics. *Proceedings of the National Academy of Sciences*, 107(11):4961, 2010.

[22] M. Ainsworth and R. Rankin. Guaranteed computable bounds on quantities of interest in finite element computations. *Internat. J. Numer. Methods Engrg*, 89:1605–1634, 2012.

[23] T. Antoun, L. Seaman, D. R. Curran, G. I. Kanel, R. V. Razorenov, and A. V. Utkin. *Spall Fracture*. Springer, 2003.

[24] G. Ariel, B. Engquist, H.-O. Kreiss, and R. Tsai. Multiscale computations for highly oscillatory problems. In *Multiscale modeling and simulation in science*, volume 66 of *Lect. Notes Comput. Sci. Eng.*, pages 237–287. Springer, Berlin, 2009.

[25] G. Ariel, B. Engquist, and R. Tsai. A multiscale method for highly oscillatory ordinary differential equations with resonance. *Math. Comp.*, 78:929–956, 2009.

[26] G. Ariel, B. Engquist, and R. Tsai. Numerical multiscale methods for coupled oscillators. *Multi. Mod. Simul.*, 7:1387–1404, 2009.

[27] G. Ariel, J.M. Sanz-Serna, , and R. Tsai. A multiscale technique for finding slow manifolds of stiff mechanical systems. *Under review*, 2012.

[28] D. Arnold, F. Brezzi, B. Cockburn, and D. Marini. Unified analysis of discontinuous galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779, 2001/2002.

[29] Z. Artstein, I. G. Kevrekidis, M. Slemrod, and E. S. Titi. Slow observables of singularly perturbed differential equations. *Nonlinearity*, 20(11):2463–2481, 2007.

[30] Z Artstein, J. Linshiz, and E. S. Titi. Young measure approach to computing slowly advancing fast oscillations. *Multiscale Model. Simul.*, 6(4):1085–1097, 2007.

[31] I. Babuska. Homogenization and its application. Mathematical and computational problems. *Numerical solution of partial differential equations, III (Proc. Third Sympos. (SYNSPADE), Univ. Maryland, College Park, Md., 1975)*, pages 89–116, 1976.

[32] I. Babuska and J.E. Osborn. Generalized finite element methods: their performance and their relation to mixed methods. *SIAM J. Numer. Anal.*, 20:510–536, 1983.

[33] G. Bal and W. Jing. Corrector theory for msfem and hmm in random media. *Multiscale Model. Simul.*, 9:1549–1587, 2011.

[34] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta numerica*, 10:1–102, 2001.

[35] A. Bensoussan, J.-L. Lions, and G.C. Papanicolaou. *Asymptotic analysis for periodic structures.* North-Holland Publishing Co., Amsterdam, 1978.

[36] L. Bers, F. John, and M. Schechter. *Partial differential equations*, volume Proceedings of the summer seminar of *Lectures in Applied Mathematics, Proceeding of the Summer Seminar.* Boulder, CO, 1957.

[37] L. Boccardo and F. Murat. Homogénéisation de problèmes quasi-linéaires. *Publ. IRMA, Lille*, 3(7):13–51, 1981.

[38] S. Boyaval. Reduced-basis approach for homogenization beyond the periodic setting. *Multiscale Model. Simul.*, 7(1):466–494, 2008.

[39] D. Braess. *Finite Elements: theory, fast solvers, and applications in elasticity theory.* Third edition. Cambridge University Press, Cambridge, 2007.

[40] A. Brandt. Multiscale scientific computation: Review 2001. *Multiscale and Multiresolution Methods: Theory and Applications*, 20:3–96, 2002.

[41] M. P. Calvo and J. M. Sanz-Serna. Instabilities and inaccuracies in the integration of highly oscillatory problems. *SIAM J. Sci. Comput.*, 31(3):1653–1677, 2009.

[42] M. P. Calvo and J. M. Sanz-Serna. Heterogeneous multiscale methods for mechanical systems with vibrations. *SIAM J. Sci. Comput.*, 32(4):2029–2046, 2010.

[43] G. T. Camacho and M. Ortiz. Computational modelling of impact damage in brittle materials. *Int. J. Solids Sturct.*, 33:2899–2938, 1996.

[44] Y. Cao, D. Gillespie, and L. Petzold. Multiscale stochastic simulation algorithm with stochastic partial equilibrium assumption for chemically reacting systems. *J. Comput. Phys.*, 206:395–411, 2005.

[45] R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55:2471–2474, 1985.

[46] P. Chartier, A. Murua, and J. M. Sanz-Serna. Higher-order averaging, formal series and numerical integration I: B-series. *Found. Comput. Math.*, 10(6):695–727, 2010.

[47] S. Chen, W. E, and C.-W. Shu. The heterogeneous multiscale method based on the discontinuous galerkin method for hyperbolic and parabolic problems. *SIAM Multiscale Model. Simul.*, 3(4):871–894, 2005.

[48] Z. Chen, W. Deng, and H. Ye. Upscaling of a class of nonlinear parabolic equations for the flow transport in heterogeneous porous media. *Commun. Math. Sci.*, 3(4):493–515, 2005.

[49] A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys.*, 2:12–26, 1967.

[50] Ph. Ciarlet and P.A. Raviart. The combined effect of curved boundaries and numerical integration in isoparametric finite element methods. *Math. Foundation of the FEM with Applications to PDE*, pages 409–474, 1972.

[51] D. Cioranescu and P. Donato. *An introduction to homogenization.*, volume 17 of *Oxford Lecture Series in Mathematics and its Applications.* Oxford University Press, New York, 1999.

[52] Ph. Clément. Approximation by finite element functions using local regularization. *Rev. Fran. Automat. Informat. Recherche Opérationnelle Sér. RAIRO Analyse Numérique*, 9(R2):77–84, 1975.

[53] D. Cohen, T. Jahnke, K. Lorenz, and C. Lubich. Numerical integrators for highly oscillatory hamiltonian systems: A review. In *Analysis, Modeling and Simulation of Multiscale Problems*, pages 553–576. Springer Berlin Heidelberg, 2006.

[54] L. Davison. *Fundamentals of Shock Wave Propagation in Solids*. Springer, 2008.

[55] M. S. Daw and M. I. Baskes. Embedded-atom method: derivation and application to impurities, surfaces, and other defects in metals. *PRB*, 29:6443, 1984.

[56] E. De Giorgi and S. Spagnolo. Sulla convergenza degli integrali dell'energia per operatori ellittici del secondo ordine. *Boll. Un. Mat. Ital.*, 4(8):391–411, 1973.

[57] A. Donev, J. B. Bell, A. L. Garcia, and B. J. Alder. A hybrid particle-continuum method for hydrodynamics of complex fluids. *Multiscale Modeling and Simulation*, 8:871–911, 2010.

[58] M. Dorobantu and B. Engquist. Wavelet-based numerical homogenization. *SIAM J. Numer. Anal.*, 35(2):540–559 (electronic), 1998.

[59] R. Du and P.B. Ming. Heterogeneous multiscale finite element method with novel numerical integration schemes. *Commm. Math. Sci.*, 8(4):863–885, 2010.

[60] W. E. Analysis of the heterogeneous multiscale method for ordinary differential equations. *Commun. Math. Sci.*, 1(3):423–436, 2003.

[61] W. E. *Principles of Multiscale Modeling*. Cambridge University Press, 2011.

[62] W. E and B. Engquist. The heterogeneous multi-scale methods. *UCLA CAM Report*, pages 02–15, 2002.

[63] W. E and B. Engquist. The heterogeneous multi-scale methods. *Comm. Math. Sci.*, 1:87–133, 2003.

[64] W. E, B. Engquist, W. Ren X. Li, and E. Vanden-Eijnden. Heterogeneous multiscale methods: A review. *Comm. Comput. Phys.*, 3:367–450, 2007.

[65] W. E, D. Liu, and E. Vanden-Eijnden. Nested stochastic simulation algorithm for chemical kinetic systems with disparate rates. *J. of Chem. Phys.*, 123:194107–194107–8, 2005.

[66] W. E, D. Liu, and E. Vanden-Eijnden. Nested stochastic simulation algorithms for chemical kinetic systems with multiple time scales. *J. Comput. Phys.*, 221(1):158–180, 2007.

[67] W. E and J. Lu. Seamless multiscale modeling via dynamics on fiber bundles. *Commun. Math. Sci.*, 5(3):649–663, 2007.

[68] W. E, P.B. Ming, and P.W. Zhang. Analysis of the heterogeneous multiscale method for elliptic homogenization problems. *J. Amer. Math. Soc.*, 18(1):121–156, 2005.

[69] W. E, W. Ren, and E. Vanden-Eijnden. A general strategy for designing seamless multiscale methods. *J. Comput. Phys.*, 228(15):5437–5453, 2009.

[70] Y. Efendiev and T. Hou. *Multiscale finite element methods. Theory and applications*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer, New york, 2009.

[71] B. Engquist. Computation of oscillatory solutions to partial differential equations. *Nonlinear hyperbolic problems (St. Etienne, 1986)*, 1270:10–22, 1987.

[72] B. Engquist and O. Runborg. Wavelet-based numerical homogenization with applications. *Multiscale and multiresolution methods*, 20:97–148, 2002.

[73] B. Engquist and Y.-H. Tsai. Heterogeneous multiscale methods for stiff ordinary differential equations. *Math. Comp.*, 74(252):1707–1742 (electronic), 2005.

[74] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. In *Acta numerica, 1995*, Acta Numer., pages 105–158. Cambridge Univ. Press, Cambridge, 1995.

[75] I. Fatkullin and E. Vanden-Eijnden. A computational strategy for multiscale chaotic systems with applications to Lorenz 96 model. *J. Comp. Phys.*, 200:605–638, 2004.

[76] E. Fermi, J. Pasta, and S. Ulam. Studies of the nonlinear problems, i. *Los Alamos Report LA-1940*, 1955. Later published in *Collected Papers of Enrico Fermi*, ed. E. Segre, Vol. II (University of Chicago Press, 1965) p.978.

[77] W. Gautschi. Numerical integration of ordinary differential equations based on trigonometric polynomials. *Numerische Mathematik*, 3:381–397, 1961.

[78] C. W. Gear and I. G. Kevrekidis. Constraint-defined manifolds: a legacy code approach to low-dimensional computation. *J. Sci. Comput.*, 25(1-2):17–28, 2005.

[79] C.W. Gear and K.A. Gallivan. Automatic methods for highly oscillatory ordinary differential equations. In *Numerical analysis (Dundee, 1981)*, volume 912 of *Lecture Notes in Math.*, pages 115–124. Springer, 1982.

[80] M.G.D. Geers, V.G. Kouznetsova, and W.A.M. Brekelmans. Multi-scale computational homogenization: Trends and challenges. *J. Comput. Appl. Math.*, 234:2175–2182, 2010.

[81] D. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.

[82] A. Gloria. Reduction of the resonance error. part 1: Approximation of homogenized coefficients. *to appear in Math. Models Methods Appl. Sci.*, 21(8):1601–1630, 2011.

[83] E. Godlewski and P.A. Raviart. *Numerical Approximation of hyperbolic systems of conservation laws.* Springer-Verlag, 1996.

[84] W. Grabowski. Coupling cloud processes with the large-scale dynamics using the cloud-resolving convection parameterization (crcp). *J. Atmos. Sci.*, 58:978–997, 2001.

[85] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2002. Structure-preserving algorithms for ordinary differential equations.

[86] E. L. Haseltine and J. B. Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic kinetics. *J. Chem. Phys.*, 117:6959–6969, 2002.

[87] P. Henning and M. Ohlberger. The heterogeneous multiscale finite element method for advection-diffusion problems with rapidly oscillating coefficients and large expected drift. *Netw. Heterog. Media*, 5(4):711–744, 2010.

[88] V.H. Hoang and C. Schwab. High-dimensional finite elements for elliptic problems with multiple scales. *Multiscale Model. Simul*, 3(1):168–194, 2005.

[89] T. Hou, X.H. Wu, and Z. Cai. Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comp.*, 68(227):913–943, 1999.

[90] J. H. Irving and J. G. Kirkwood. The statistical mechanical theory of transport processes iv. *J. Chem. Phys.*, 18:817–829, 1950.

[91] Jr. J. Douglas and T. Dupont. A Galerkin method for a nonlinear Dirichlet problem. *Math. Comp.*, 29(131):689–696, 1975.

[92] V. Jikov, S. Kozlov, and O.A. Oleinik. *Homogenization of differential operators and integral functionals.* Springer-Verlag, Berlin, Heidelberg, 1994.

[93] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, and C. Theodoropoulos. Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis. *Comm. Math. Sci.*, 1:715–762, 2003.

[94] H.-O. Kreiss. Problems with different time scales for ordinary differential equations. *SIAM J. Numer. Anal.*, 16(6):980–998, 1979.

[95] H.-O. Kreiss. Problems with different time scales. In *Acta numerica, 1992*, pages 101–139. Cambridge Univ. Press, 1992.

[96] H.-O. Kreiss and J. Lorenz. Manifolds of slow solutions for highly oscillatory problems. *Indiana Univ. Math. J.*, 42(4):1169–1191, 1993.

[97] T. G. Kurtz. A limit theorem for perturbed operator semigroups with applications for random evolutions. *J. Functional Analysis*, 12:55–67, 1973.

[98] O.A. Ladyzhenskaya. *The boundary value problems of mathematical physics*, volume 49 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1985.

[99] B. Leimkuhler and S. Reich. *Simulating Hamiltonian dynamics*, volume 14 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, 2004.

[100] R. LeVeque. *Numerical Methods for Conservation Laws.* Birkhauser, 1990.

[101] X. Li. private communication.

[102] X. Li. Variational boundary conditions for molecular dynamic in solids: treatment of the loading conditions. *J. Comp. Phys.*, 227:10078–10093, 2008.

[103] X. Li and W. E. Multiscale modeling for dynamics of solids at finite temperature. *J. Mech. Phys. Solids*, 56:1650 – 1685, 2005.

[104] X. Li and W. E. Variational boundary conditions for molecular dynamics simulations of solids at low temperature. *Comm. Comp. Phys.*, 1:136 – 176, 2006.

[105] X. Li and W. E. Boundary conditions for molecular dynamics simulations at finite temperature: Treatment of the heat bath. *Phys. Rev. B*, 76:104107, 2007.

[106] L. Maragliano and E. Vanden-Eijnden. A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations. *Chemical physics letters*, 426(1-3):168–175, 2006.

[107] L. Maragliano and E. Vanden-Eijnden. On-the-fly string method for minimum free energy paths calculation. *Chemical Physics Letters*, 446(1-3):182–190, 2007.

[108] A.-M. Matache, I. Babuska, and C. Schwab. Generalized p-fem in homogenization. *Numer. Math.*, 86(2):319–375, 2000.

[109] A.-M. Matache and C. Schwab. Two-scale fem for homogenization problems. *M2AN Math. Model. Numer. Anal*, 36(4):537–572, 2002.

[110] C. Miehe, J. Schröder, and C. Bayreuther. On the homogenization analysis of composite materials based on discretized fluctuations on the micro-structure. *Acta Mechanica*, 135:1–16, 2002.

[111] P.B. Ming and X. Yue. Numerical methods for multiscale elliptic problems. *J. Comput. Phys.*, 214(1):421–445, 2006.

[112] P.B. Ming and P.W. Zhang. Analysis of the heterogeneous multiscale method for parabolic homogenization problems. *Math. Comp.*, 76(257):153–177 (electronic), 2007.

[113] F. Murat and L. Tartar. H-convergence, topics in the mathematical modeling of composite materials. *Progr. Nonlinear Differential Equations Appl.*, 31:21–43, 1997.

[114] H. Nessyahu and E. Tadmor. Nonoscillatory central differencing for hyperbolic conservation laws. *J. Comp. Phys.*, 87:408–463, 1990.

[115] N. Neuss, W. Jäger, and G. Wittum. Homogenization and multigrid. *Computing*, 66(1):1–26, 2001.

[116] R.H. Nochetto, A. Veeser, and M. Verani. A safeguarded dual weighted residual method. *IMA journal of Numerical Analysis*, 29(1):126–140, 2009.

[117] A. Nonnenmacher. *Adaptive Finite Element Methods for Multiscale Partial Differential Equations, Thèse No 5097.* École Polytechnique Fédérale de Lausanne, 2011.

[118] J.T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5-6):735 – 756, 2001.

[119] J.T. Oden, S. Prudhomme, A. Romkes, and P. Bauman. Multiscale modeling of physical phenomena: adaptive control of models. *SIAM J. Sci. Comput.*, 28(6):2359–2389, 2006.

[120] M. Ohlberger. A posteriori error estimates for the heterogeneous multiscale finite element method for elliptic homogenization problems. *Multiscale Model. Simul*, 4(1):88–114, 2005.

[121] G. C. Papanicolaou. Introduction to the asymptotic analysis of stochastic equations. In *Modern modeling of continuum phenomena (Ninth Summer Sem. Appl. Math., Rensselaer Polytech. Inst., Troy, N.Y., 1975)*, pages 109–147. Lectures in Appl. Math., Vol. 16. Amer. Math. Soc., Providence, R.I., 1977.

[122] L. Petzold, O.J. Laurent, and Y. Jeng. Numerical solution of highly oscillatory ordinary differential equations. *Acta Numerica*, 6:437–483, 1997.

[123] S. Prudhomme and J.T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):313–331, 1999.

[124] K. Ravi-Chandar and W. G. Knauss. An experimental investigation into dynamic fracture: I. crack initiation and arrest. *Int. J. Fracture*, 43:247, 1984.

[125] P.A. Raviart. The use of numerical integration in finite element methods for solving parabolic equations. *Miller, J. J. H. (ed.), Topics in Numerical Analysis, Academic Press*, pages 233–264, 1973.

[126] W. Ren and W. E. Heterogeneous multiscale method for the modeling of complex fluids and microfluidics. *J. Comput. Phys.*, 204:1–26, 2005.

[127] H. Salis and Y. Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *J. Chem. Phys.*, 122:054103–054103–13, 2005.

[128] A. Samant and D. G. Vlachos. Overcoming stiffness in stochastic simulation stemming from partial equilibrium: A multiscale monte carlo algorithm. *J. Chem. Phys.*, 123:144114–144114–8, 2005.

[129] Sandia National Lab. http://cubit.sandia.gov, 1997-2010.

[130] J. M. Sanz-Serna. Modulated Fourier expansions and heterogeneous multiscale methods. *IMA J. Numer. Anal.*, 29(3):595–605, 2009.

[131] R.E. Scheid. The accurate numerical solution of highly oscillatory ordinary differential equations. *Mathematics of Computation*, 41(164):487–509, 1983.

[132] R. Srivastava, L. You, J. Summers, and J. Yin. Stochastic vs. deterministic modeling of intracellular viral kinetics. *J. Theor. Biol.*, 218:309–321, 2002.

[133] E. B. Tadmor, M. Ortiz, and R. Phillips. Quasicontinuum analysis of defects in crystals. *Phil. Mag. A*, 73:1529 – 1563, 1996.

[134] M. Tao, H. Owhadi, and J. Marsden. Nonintrusive and structure preserving multiscale integration of stiff ODEs, SDEs, and Hamiltonian systems with hidden slow dynamics via flow averaging. *Multiscale Model. Simul.*, 8(4):1269–1324, 2010.

[135] K. Terada and N. Kikuchi. A class of general algorithms for multi-scale analyses of heterogeneous media. *Comput. Methods Appl. Mech. Engrg.*, 190(40-41):5427–5464, 2001.

[136] E. Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Comm. Math. Sci.*, 1:385–391, 2003.

[137] E. Vanden-Eijnden. On HMM-like integrators and projective integration methods for systems with multiple time scales. *Communications in Mathematical Sciences*, 5(2):495–505, 2007.

[138] R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques.* Wiley-Teubner, New-York, 1996.

[139] T.C. Wallstrom, S. Hou, M. A. Christie, L. J. Durlofsky, and D. H. Sharp. Accurate scale up of two phase flow using renormalization and nonuniform coarsening. *Comput. Geosci.*, 3(1):67–87, 1999.

[140] Y. Xing, W. Grabowski, and A. J. Majda. New efficient sparse space-time algorithms for superparamterization on mesoscales. *Mon. Wea. Rev.*, 137:4307–4324, 2009.

[141] J. Xu. Two-grid discretization techniques for linear and nonlinear pde. *SIAM J. Numer. Anal.*, 33(5):1759–1777, 1996.

[142] X.-P. Xu and A. Needleman. Numerical simulations of fast crack growth in brittle solids. *J. Mech. Phys. Solids*, 42:1397–1434, 1994.

[143] X. Yue and W. E. The local microscale problem in the multiscale modeling of strongly heterogeneous media: effects of boundary conditions and cell size. *J. Comput. Phys.*, 222(2):556–572, 2007.

[144] V. V. Yurinskiĭ. Averaging of symmetric diffusion in a random medium. *Sibirsk. Mat. Zh.*, 27(4):167–180, 1986.

Recent publications :

MATHEMATICS INSTITUTE OF COMPUTATIONAL SCIENCE AND ENGINEERING
Section of Mathematics
Ecole Polytechnique Fédérale
CH-1015 Lausanne

04.2012 D. KRESSNER, C. TOBLER:
htucker – A Matlab toolbox for tensors in hierarchical Tucker format

05.2012 A. ABDULLE, G. VILLMART, KONSTANTINOS C. ZYGALAKIS:
Second weak order explicit stabilized methods for stiff stochastic differential equations

06.2012 A. CABOUSSAT, S. BOYAVAL, A. MASSEREY:
Three-dimensional simulation of dam break flows

07.2012 J BONNEMAIN, S. DEPARIS, A. QUARTERONI:
Connecting ventricular assist devices to the aorta: a numerical model

08.2012 J BONNEMAIN, ELENA FAGGIANO, A. QUARTERONI, S. DEPARIS:
A framework for the analysis of the haemodynamics in patient with ventricular assist device

09.2012 T. LASSILA, A. MANZONI, G. ROZZA:
Reduction strategies for shape dependent inverse problems in haemodynamics

10.2012 C. MALOSSI, P. BLANCO, P. CROSETTO, S. DEPARIS, A. QUARTERONI:
Implicit coupling of one-dimensional and three-dimensional blood flow models with compliant vessels

11.2012 S. FLOTRON J. RAPPAZ:
Conservation schemes for convection-diffusion equations with Robin's boundary conditions

12.2012 A. UMSCHMAJEW, B. VANDEREYCKEN:
The geometry of algorithms using hierarchical tensors

13.2012 D. KRESSNER, B. VANDEREYCKEN:
Subspace methods for computing the pseudospectral abscissa and the stability radius

14.2012 B. JEURIS, R. VANDEBRIL, B. VANDEREYCKEN:
A survey and comparison of contemporary algorithms for computing the matrix geometric mean

15.2012 A. MANZONI, A. QUARTERONI, G. ROZZA:
Computational reduction for parametrized PDEs: strategies and applications

16.2012 A.C.I. MALOSSI, J. BONNEMAIN:
Numerical comparison and calibration of geometrical multiscale models for the simulation of arterial flows

17.2012 A. ABDULLE, W. E, B. ENGQUIST, E. VANDEN-EIJNDEN:
The heterogeneous multiscale method