A reversible multiscale integration method [*]

Gil Ariel [†] Bjorn Engquist [‡] Richard Tsai [§]

April 13, 2009

Abstract

A multiscale, time reversible method for computing the effective slow behavior of systems of highly oscillatory ordinary differential equations is presented. The proposed method relies on correctly tracking a set of slow variables that is sufficient to approximate any variable and functional that are slow under the dynamics of the system. The algorithm follows the framework of the heterogeneous multiscale method. The notion of time reversibility in the multiple time-scale setting is discussed. The algorithm requires nontrivial matching between the microscopic state variables and the macroscopic slow ones. Numerical examples show the efficiency of the multiscale method and the advantages of time reversibility.

1 Introduction

The dynamics of Hamiltonian systems are known to have special geometrical symmetries. Let $\mathcal{H}(\mathbf{q},\mathbf{p})$ denote the Hamiltonian of a system with position coordinates $\mathbf{q}$ and momentum $\mathbf{p}$. An important property of the Hamiltonian dynamics is that if $(\mathbf{q}(t),\mathbf{p}(t))$ is a solution then $(\mathbf{q}(-t),-\mathbf{p}(-t))$ is also a solution. Time reversible and symplectic schemes have proven to be highly valuable tools for integrating systems of ordinary differential equations (ODEs) [18, 24] whose solutions possess similar symmetries. These methods are particularly useful for integration over long time segments. However, for general systems involving two or more time scales, most such schemes require a step size that is of the order of the fastest scale, typically due to stability and accuracy considerations. As a result, application of conventional schemes becomes prohibitively expensive and inefficient. Accordingly, devising a multiscale time reversible algorithm seems desirable particularly if it can inherit the benefits of both the multiscale and the time reversible approaches.

Many challenges of multiscale numerical integration have been addressed by several different approaches. Stiff problems with fast transients can be optimally solved by implicit schemes [8, 19, 21]. The Chebyshev methods [1, 23] as well as the projective integrator approach [16] provide stable and explicit computational strategies for this class of problems in general. For near harmonic oscillatory problems, traditional numerical approaches attempt to either filter out or fit fast oscillations to some known functions in order to reduce the complexity, e.g. [15, 22, 30], or use some notion of Poincaré map to determine slow changes in the orbital structure [17, 27].

A general class of approaches aiming at Hamiltonian systems are geometric integration schemes that preserve a discrete version of certain invariances. We refer the readers to [18] and [24] for an extensive

[*]

[†] Department of Mathematics, The University of Texas at Austin, Austin, TX, 78712, USA, (ariel@math.utexas.edu).

[‡] Department of Mathematics, The University of Texas at Austin, Austin, TX, 78712, USA, (engquist@math.utexas.edu).

[§] Department of Mathematics, The University of Texas at Austin, Austin, TX, 78712, USA, (ytsai@math.utexas.edu).

list of literature. Many schemes specialized for finite dimensional mechanical systems can be conveniently derived from the view point of variational integrators [26]. In certain applications, special considerations are given to the expensive cost of evaluating non-local potentials in large systems, see e.g. the impulse method and its derivatives [24]. For a recent review on numerical methods for highly oscillatory systems see [7].

In this paper we propose a multiscale numerical scheme that approximates the slowly varying effective dynamics of highly oscillatory ODE systems. The numerical method is time reversible in the sense that, upon time reversal, the algorithm traces back to the initial condition with no truncation error. To be more precise, let $h>0$ denote the step size used in the algorithm. Then, taking a step of size h followed by a step of size $-h$, the algorithm goes back to the initial state. The only error is in round-off. We follow the framework of the Heterogeneous Multiscale Methods (HMM) [9, 10, 11, 32], and the general strategy proposed in [13]. We point out here that it is not entirely clear if for general highly oscillatory systems which are time-reversible in the Hamiltonian sense, the effective slow dynamics is also reversible in the same sense. *Nevertheless, it is reasonable to ask if the multiscale scheme can update the fast scale variables in a (numerically) reversible way, without fully resolving all the fast oscillations at all time while remaining consistent with the slow dynamics. Furthermore, it is interesting to see if this additional symmetry brings any benefit for the overall multiscale approximation.* To this end we give two numerical examples whose main purpose is to demonstrate that reversible methods may in fact be superior to non-reversible ones. In the first example of the inverted pendulum, we find that the reversible method allows bigger macro steps than a non-reversible method of the same order. In the Fermi-Pasta-Ulam (FPU) example [14], we show that energy drift is much smaller with the reversible method. The energy can stay constant over long time segments and does not increase linearly in time as with non-reversible schemes. In this respect, the purpose of the examples is more than just a proof of concept. They also serve to demonstrate the advantages of a reversible macro-solver.

Recently, Calvo and Sanz-Serna suggested an HMM scheme that is both time reversible and symplectic [6]. Their method can be applied to some types of Hamiltonian systems, for example, mechanical systems that are driven by a single external fast oscillation. These ideas were further developed for stiff mechanical systems with constraints [20].

Here, we consider general ODE systems of the form

$$\dot{\mathbf{x}} = \epsilon^{-1} f(\mathbf{x}) + g(\mathbf{x}), \qquad \mathbf{x}(0) = \mathbf{x}_0,\ t \in [0,T], \tag{1.1}$$

where $0<\epsilon\leq\epsilon_0$, $\mathbf{x}=(x_1,\ldots,x_d)\in\mathbb{R}^d$ and $0<T<\infty$ is independent of ϵ. It is assumed that the solution of (1.1) remains in a domain $\mathcal{D}_0\subset\mathbb{R}^d$ which is bounded independent of ϵ for all $t\in[0,T]$. For fixed ϵ and initial condition $\mathbf{x}_0$, the solution of (1.1) is denoted $\mathbf{x}(t;\epsilon,\mathbf{x}_0)$. For brevity we will write $\mathbf{x}(t)$ when the dependence on ϵ and $\mathbf{x}_0$ is not directly relevant to the discussion. Furthermore, in this paper we only consider the case in which the fast dynamics is oscillatory in nature rather than dissipative. In particular, we assume that the unperturbed system, $g(\mathbf{x})=0$, has a continuous family of periodic solutions or an asymptotically stable periodic limit cycle.

The general approach previously developed in [2, 3, 4, 12] and [13], is to identify a set of functions in the state space whose values change slowly along the oscillatory trajectories. The time evolution of these slow variables is used to guide the slow time-scale dynamics. The ODE (1.1) is then integrated following the HMM framework: a Macro-solver integrates the effective, but generally unknown evolution equation for the slow variables under the dynamics of (1.1). The rates of change for these slow variables are computed on-the-fly by a micro-solver that integrates the full ODE (1.1) for short time segments. For a recent review see [11].

In the typical HMM setting, the macroscopic system evolves a set of slowly changing quantities (slow variables in our case), denoted here by ξ

$$\frac{d}{dt}\xi = F(\xi,t), \tag{1.2}$$

where the right hand side $F(\xi,t)$ is evaluated by the appropriate averaging of the solutions of (1.1) with suitable initial conditions. See Section 2 for details. Hence, one can formally apply any desirable scheme to discretize (1.2). One possibility includes time reversible schemes such as the leap-frog scheme

$$\xi_{n+1} = \xi_{n-1} + 2\Delta t \cdot F(\xi_n,t), \tag{1.3}$$

where ξ_n denotes the numerical approximation for ξ at the n'th macroscopic step. However, this is not the complete story at the level of actual numerical discretization which involves nontrivial coupling of the numerical schemes for the macro- and micro-scales, e.g. the evaluation of $F(\xi_n,t)$ via suitable solutions of (1.1). Additional challenges emerge if we want to update the microscopic variables $\mathbf{x}_n$ in a reversible way while remaining consistent with the underlying effective dynamics regulated by the macroscopic slow variables.

Despite the fact that in general leap-frog is second order accurate, the slow variables need to be related to the original fast state variable. In section 3 we show that a naive implementation may lead to a reduction in accuracy and describe a way to regain the second order accuracy of leap-frog. All these couplings make time reversibility at both macroscopic and microscopic level in an HMM scheme non-trivial.

The organization of the paper is as follows. Section 2 reviews the main results and algorithms developed in [2], [3] and [12] and examines the notion of time reversibility in our multiple time-scale setting. Section 3 describes particular implementations of the method which are time reversible and analyses their accuracy. A few examples are presented in Section 4. We end with concluding remarks in Section 5.

2 The HMM scheme

In order to study the long time properties of (1.1) we need to distinguish between the fast and slow constituents of the dynamics. We say that a real valued smooth function (variable) $\alpha(\mathbf{x})$ is slow with respect to (1.1) in an open connected set $\mathcal{A}$ if

$$\max_{\mathbf{x}_0\in\mathcal{A},t\in[0,T]} \left| \frac{d}{dt}\alpha(\mathbf{x}(t;\epsilon,\mathbf{x}_0)) \right| \le C_0, \tag{2.1}$$

where C_0 is a constant that is independent of ϵ. Otherwise, $\alpha(\mathbf{x})$ is said to be fast. Similarly, we say that a quantity or constant is of order one if it is bounded independent of ϵ in $\mathcal{D}_0$ or $[0,T]$.

Of course, any function of slow variables is also slow. Therefore, it is reasonable to look for variables which are functionally independent, i.e., a vector of slow variables $\xi = (\xi^{(1)}(\mathbf{x}),\ldots,\xi^{(r)}(\mathbf{x}))$ such that $\nabla\xi^{(1)}(\mathbf{x}),\ldots,\nabla\xi^{(r)}(\mathbf{x})$ are linearly independent in $\mathcal{A}$. Since r is bounded by the dimension, d, it is useful to look at a set with a maximal number of functionally independent slow variables. Augmenting the slow variables with $d-r$ fast ones $z=(z_1,\ldots,z_{d-r})$ such that $\partial(\xi,z)/\partial\mathbf{x}$ is non-singular in $\mathcal{A}$, one obtains a local coordinate systems, i.e., a chart of the states space. We will refer to a chart in which a maximal number of coordinates is slow as a maximal slow chart for $\mathcal{A}$ with respect to the ODE (1.1). Covering the set $\mathcal{D}_0$ by maximal slow charts we obtain a maximal slow atlas for $\mathcal{D}_0$.

One of the important observations which follow from our definition of a slow chart is that typical slow variables do not appear in conjugate pairs of some generalized position and momentum. Hence, there is no clear notion of macroscopic time reversibility in the Hamiltonian sense. In [20], the slow behavior of a particular class of stiff mechanical systems is studied. It is shown that in the limit of $\epsilon \to 0$ the dynamics can be approximated by an effective equation with Holonomic constraints [5]. As a result, this effective dynamics is time reversible.

We continue by examining the relations between the original state variables, which are typically fast, and the slow coordinates in a maximal slow chart. In particular, we wish to establish the existence of an effective evolution equation for the slow variables $\xi(\mathbf{x}(t))$ under the flow of (1.1). The assumption that the unperturbed dynamics is periodic implies that the only fast coordinate is equivalent to rotation on the unit circle with constant velocity, i.e., $\phi \in \mathbb{S}^1$. This case is quite general since many weakly perturbed integrable systems in resonance fall into this category through the notion of action-angle variables. Then, an averaging principle can be used to prove that for small ϵ, $\xi(\mathbf{x}(t;\epsilon,\mathbf{x}_0))$ is well approximated in $[0,T]$ by an effective equation of the form

$$\dot{\xi} = F(\xi), \quad \xi(0) = \xi(\mathbf{x}_0). \tag{2.2}$$

See [2, 5, 28] for details. The requirement that (ξ,ϕ) is a maximal slow chart is critical for the derivation of (2.2). Without it, there is no guaranty that the right hand side of the averaged equation does not depend on additional slow variables which may be hidden or unknown.

The effective equation (2.2) may not be available as an explicit formula. Instead, the idea behind the HMM algorithm is to evaluate $F(\xi)$ by numerical solutions of the original ODE (1.1) on significantly reduced time intervals. In this way, the HMM algorithm approximates an assumed effective equation whose form is typically unknown. This strategy is advantageous if $F(\xi)$ can be approximated efficiently. The additional requirement of a time reversibility poses constraints on the way the $F(\xi)$ is evaluated. Furthermore, integration of (2.2) should be done while keeping the sequence of fast state variables reversible even across macroscopic steps. The next section describes such an algorithm.

2.1 The algorithm

Suppose $\xi = (\xi^{(1)}(\mathbf{x}),\ldots,\xi^{(r)}(\mathbf{x}))$ are the slow variables in a slow atlas for (1.1). The system is integrated using a two level algorithm, each level corresponding to a different time scale. The first is a Macro-solver which integrates the effective equation (2.2) for the slow variables ξ. The second level is a micro-solver that is invoked whenever the Macro-solver needs an estimate of $F(\xi)$. The micro-solver computes a short time solution of (1.1) using suitable initial data. Then, the time derivative of ξ is approximated by

$$\dot{\xi}(t) \sim \langle \dot{\xi} \rangle_\eta(t) = \int_{-\eta/2}^{\eta/2} \dot{\xi}(\mathbf{x}(t+\tau)) K_\eta(t-\tau) d\tau, \tag{2.3}$$

where, $K_\eta(\cdot)$ denotes a smooth averaging kernel with support on $[-\eta/2,\eta/2]$. Note that $\dot{\xi}$ is not necessarily slow. However, it is bounded independent of ϵ. The properties of averaging with respect to a kernel will be reviewed shortly.

To better explain the algorithm, denote the Macro-solver sample times by $t_0,\ldots,t_N$, $N = T/H$, and its output at corresponding times by $\mathbf{x}_0,\ldots,\mathbf{x}_N$. At the n-th Macro-step, the micro-solver can be implemented using any scheme with step-size h and initial condition $\mathbf{x}(t_n) = \mathbf{x}_n$. It integrates the full ODE both backwards and forward in time to approximate the solution in $[t_n - \eta/2, t_n + \eta/2]$. The structure of the algorithm, depicted in Figure 1, is as follows [10, 13]

1. Initial conditions: $\mathbf{x}(0) = \mathbf{x}_0$, $\xi_0 = \xi(\mathbf{x}_0)$ and $n = 0$.

2. Force estimation:

   (a) micro-simulation: solve (1.1) in $[t_n - \eta/2, t_n + \eta/2]$ with initial conditions $\mathbf{x}(t_n) = \mathbf{x}_n$.
   (b) Averaging: approximate $\dot{\xi}(t_n)$ by $\langle \dot{\xi} \rangle_\eta (t_n)$.

3. Macro-step (forward-Euler example): $\xi_{n+1} = \xi_n + H \langle \dot{\xi} \rangle_\eta (t_n)$.

4. Reconstruction: find $\mathbf{x}_{n+1}$ consistent with ξ_{n+1}. take $\mathbf{x}_{n+1} = \mathbf{x}_n + H\tilde{F}_n$, where $\tilde{F}_n$ is the least squares solution of the linear system
$$\frac{\partial \xi(\mathbf{x}_n)}{\partial \mathbf{x}} \tilde{F}_n = \langle \dot{\xi} \rangle_\eta (t_n)$$

5. $n = n+1$. Repeat steps (2) and (3) to time T.

Here, $\partial \xi / \partial \mathbf{x}$ is a matrix whose k'th row is $\nabla \xi^{(k)}$. The scheme described above can be generalized to Macro-solvers with higher order accuracy.

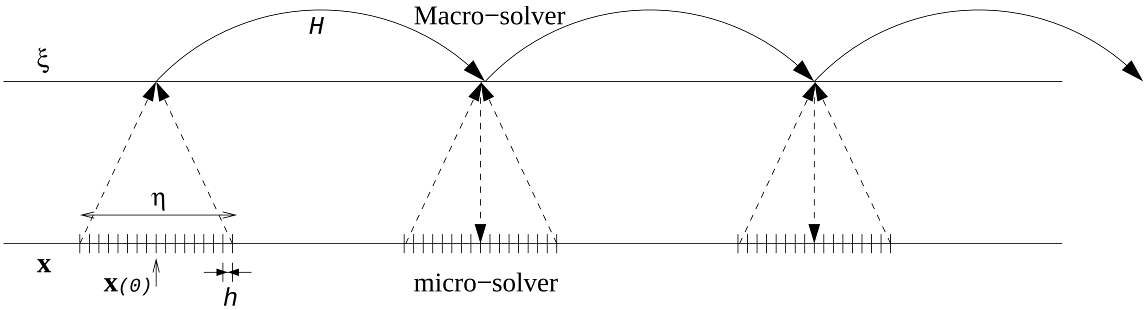

Figure 1: The cartoon depicts the time steps taken by the HMM scheme. At the n-th Macro step, a micro-solver with step size h integrates (1.1) to approximate $\mathbf{x}(t)$ in a time segment $[t_n - \eta/2, t_n + \eta/2]$. This data is used to calculate $\langle \dot{\xi}(\mathbf{x}) \rangle_\eta (t)$. Then, the Macro-solver takes a big step of size $H\tilde{F}_n$, where $\tilde{F}_n$ is consistent with $\langle \dot{\xi}^{(k)} \rangle_\eta$ for all slow variables $\xi^{(k)}$ in the maximal slow chart.

2.2 Updating state variables

At the core of the HMM framework lies the idea that we are actually solving the effective equation $\dot{\xi} = F(\xi)$ at the macroscopic time scale. Accordingly, one has the freedom of using an integrator of choice. For example, applying forward-Euler with step size H yields a single-step rule

$$\xi_{n+1} = \xi_n + HF(\xi_n), \tag{2.4}$$

where ξ_n denotes the approximation for $\xi(t_n)$. Alternatively, one can use the familiar two-step leap-frog method, which is reversible in time

$$\xi_{n+1} = \xi_{n-1} + 2HF(\xi_n). \tag{2.5}$$

As in the algorithm described above, $F(\xi_n)$ is approximated by solving the full system for a very short time window.

Once new values for the slow variables are determined, it is necessary to find a new set of state variables, $\mathbf{x}_{n+1}$, that are consistent with the new slow state, ξ_{n+1}, i.e., $\xi(\mathbf{x}_{n+1}) = \xi_{n+1}$. This stage is referred to as reconstruction. Instead, the algorithm described above bypasses this difficulty by evolving the state variables

$\mathbf{x}$ directly [2, 13]. Hence, step (3) of the algorithm above does not need to be performed. Of course, this has to be done in a way that is consistent with the slow dynamics (to some order in H). With single-step methods, the Macro-step takes the form

$$\mathbf{x}_{n+1} = \mathbf{x}_n + H\tilde{F}(\mathbf{x}_n, \xi_n). \tag{2.6}$$

Comparing with (2.4) we find that, to order H^2, $\tilde{F}(\mathbf{x}_n, \xi_n)$ can be taken to be the least squares solution of the linear system

$$\frac{\partial \xi(\mathbf{x}_n)}{\partial \mathbf{x}} \tilde{F}(\mathbf{x}_n, \xi_n) = F(\xi_n).$$

Higher order schemes following Runke-Kutta methods are developed in [2].

The main goal of this paper is to develop a time reversible leap-frog scheme for the state variables which consistently embeds the leap-frog scheme for ξ, (2.5). For example, a naive attempt for a reversible reconstruction step is to take

$$\mathbf{x}_{n+1} = \mathbf{x}_{n-1} + 2H\tilde{F}_n, \tag{2.7}$$

where $\tilde{F}_n$ is the least squares solution of the linear system

$$\frac{\partial \xi(\mathbf{x}_n)}{\partial \mathbf{x}} \tilde{F}_n = \langle \dot{\xi} \rangle_\eta (t_n).$$

Although this approach is simple, in section 3 it is shown that this leads to a method which is only first order accurate in H. This low order accuracy is often too poor and impractical.

More generally, the scheme can be written implicitly in the form

$$G(\mathbf{x}_n, \xi_n; \mathbf{x}_{n-1}, \mathbf{x}_{n+1}) = 0, \tag{2.8}$$

where we require that $\mathbf{x}_{n-1}$ and $\mathbf{x}_{n+1}$ are consistent with (2.5) to some known power of H. In Section 3, we suggest possible forms for G, which are skew-symmetric in $\mathbf{x}_{n-1}$ and $\mathbf{x}_{n+1}$, i.e.,

$$G(\mathbf{x}_n, \xi_n; \mathbf{x}_{n-1}, \mathbf{x}_{n+1}) = -G(\mathbf{x}_n, \xi_n; \mathbf{x}_{n+1}, \mathbf{x}_{n-1}).$$

As a result, the evolution operator is time reversible. The process of finding a microscopic state $\mathbf{x}_{n+1}$, consistent with the Macroscopic ξ_{n+1} is under-determined since different $\mathbf{x}$ may correspond to the same slow coordinates ξ. Using (2.8) one picks a particular possible solution in such a way that the entire algorithm become reversible in time.

Stability of the new scheme is inherited from that of the leap-from method for ξ, (2.5). Note that the approximation is only consistent for the slow variables ξ and not for the original state variables $\mathbf{x}$, since any information on the fast coordinate is discarded.

2.3 Averaging using kernels

Let $K(\cdot)$ denote a smooth kernel function with support on $[-1,1]$ with unit mass, $\int_{-1}^{1} K(\tau) d\tau = 1$, and vanishing first moment, $\int_{-1}^{1} K(\tau)\tau d\tau = 0$. In this paper we restrict the discussion to kernels which are symmetric with respect to their mid-point. For example, the following smooth exponential kernel was found useful:

$$K(t) = Z^{-1} \exp\left(-\frac{5}{4} \frac{1}{(t-1)(t+1)} \right), \tag{2.9}$$

for $t \in (-1,1)$ and zero otherwise. Here, Z is a normalization constant. For $\eta > 0$ let,

$$K_\eta(\tau) = \frac{2}{\eta} K(\frac{2}{\eta}\tau). \tag{2.10}$$

We will take η to be ϵ dependent such that $0 < \epsilon < \eta \ll 1$. The convolution of a function $a(t)$ with K_η is denoted as (recall (2.3))

$$\langle a \rangle_\eta(t) = \int_{-\eta/2}^{\eta/2} a(t+\tau) K_\eta(t-\tau) d\tau. \tag{2.11}$$

Typically, the fast dynamics in equations such as (1.1) is one of two types (compare to the linear case, $f(\mathbf{x}) = A\mathbf{x}$). The first consists of modes that are attracted to a low dimensional manifold in a time scale of order one. These modes are referred to as transient or dissipative modes and will not be discussed in this paper. The second type consists of oscillators with constant or slowly changing frequencies. Averaging of oscillatory modes filters out high frequency oscillations. The errors introduced by the averaging are estimated in [2] and [12]. For example, for a function $\beta(t)$ with period one and a kernel with q continuous derivatives, we have that

$$\left| K_\eta(\cdot) * \beta(\epsilon^{-1}\cdot) - \bar{\beta} \right| \leq C ||\beta||_\infty ||K||_{W^{1,q}} \left(\frac{\epsilon}{\eta} \right)^q, \tag{2.12}$$

where $\bar{\beta} = \int_0^1 \beta(\tau) d\tau$, $||\cdot||_\infty$ denotes the sup norm in $\mathcal{D}_0$,

$$||\beta||_\infty = \sup_{\mathbf{x} \in \mathcal{D}_0} |\beta(\mathbf{x})|, \tag{2.13}$$

and

$$||K||_{W^{1,q}} = \int_{-1}^{1} |K^{(q)}(t)| dt. \tag{2.14}$$

Here, $K^{(q)}$ denotes the q-th derivative of K.

3 A consistent and reversible multiscale solver

In this Section we describe our strategy that enables time reversibility in the entire HMM algorithm described in Section 2.1 while maintaining the consistency between the macro- and micro states. Let $\mathbf{x}(t)$ denote the exact solution of the full ODE system (1.1) with the initial condition $\mathbf{x}(0) = \mathbf{x}_0$. In addition, let H denote the Macroscopic step size used in the Macro-solver, $\mathbf{x}_n = \mathbf{x}(nH)$ and $\Delta^+ \mathbf{x}_{n-1} = \mathbf{x}_n - \mathbf{x}_{n-1}$.

For illustration purposes, consider the following linear system describing a slowly expanding spiral

$$\begin{cases} \dot{z}_1 & = -\epsilon^{-1} z_2 + z_1 \\ \dot{z}_2 & = \epsilon^{-1} z_1 + z_2, \end{cases} \tag{3.1}$$

with initial conditions $z_1(0) = 1$ and $z_2(0) = 0$. The exact solution of (3.1) is $\mathbf{x}(t) = (z_1(t), z_2(t)) = (e^t \cos \epsilon^{-1} t, e^t \sin \epsilon^{-1} t)$ and $\xi(\mathbf{x}) = z_1^2 + z_2^2$ is a slow variable. Figure 2 depicts a forward-Euler type Macro-step for (3.1).

Let γ denote a smooth curve connecting $\mathbf{x}_0$ and $\mathbf{x}_1 = \mathbf{x}(H)$. The change in $\xi(\mathbf{x})$ along the curve can be expressed as

$$\Delta^+ \xi(\mathbf{x}_0) = \xi(\mathbf{x}_1) - \xi(\mathbf{x}_0) = \int_\gamma \nabla \xi(\mathbf{x}) \cdot d\mathbf{x}. \tag{3.2}$$

We are interested in two particular curves. The first is along the solution of the ODE. Taking $\gamma = \{\mathbf{x}(t)|0 \leq t \leq H\}$, (3.2) yields

$$\Delta^+\xi(\mathbf{x}_0) = \int_0^H \nabla\xi(\mathbf{x}(\tau)) \cdot \dot{\mathbf{x}} d\tau = \int_0^H \dot{\xi}(\mathbf{x}(\tau)) d\tau.$$

For $0 < \epsilon < \eta \ll H$, we obtain an estimate

$$\Delta^+\xi(\mathbf{x}_0) = H \left\langle \dot{\xi}(\mathbf{x}) \right\rangle_\eta (0) + O(H^2 + e), \tag{3.3}$$

where e denotes the error from approximating the average $\dot{\xi}$ using kernels. We assume e is negligible compared to H^2.

An alternative choice of curve follows the straight line in phase space connecting $\mathbf{x}_0$ and $\mathbf{x}_1$. Parameterizing the segment as $s\mathbf{x}_1 + (1-s)\mathbf{x}_0$ with $s \in [0,1]$ yields

$$\Delta^+\xi(\mathbf{x}_0) = \int_0^1 \nabla\xi(\mathbf{x}_0 + s\Delta^+\mathbf{x}_0) \cdot (\mathbf{x}_1 - \mathbf{x}_0) ds = \nabla\xi(\mathbf{x}_0) \cdot \Delta^+\mathbf{x}_0 + O(H^2). \tag{3.4}$$

Comparing (3.3) and (3.4) we deduce that to second order in H,

$$H \langle \dot{\xi}(\mathbf{x}) \rangle_\eta (0) = \nabla\xi(\mathbf{x}_0) \cdot \Delta^+\mathbf{x}_0. \tag{3.5}$$

Thus, solving for $\Delta^+\mathbf{x}_0$, (3.5) yields the forward-Euler Macro-step formula used for the algorithm described in Section 2.1 with a local truncation error that is second order in H. For larger systems with several slow variables (3.5) generalizes to a linear system whose components comes directly from each of the slow variables, and (3.5) can be solved using singular value decomposition.

The derivation above can be generalized to high order Runge-Kutta type methods by improving the approximation in (3.2) and (3.4). This approach is developed in [2]. More attention is needed for multistep methods that uses directly the original variables of the full system at the Macroscopic level. In the following, we discuss our approach for designing such schemes.

Suppose the multiscale algorithm has already produced the first two Macroscopic steps, $\mathbf{x}_0 = \mathbf{x}(0)$ and $\mathbf{x}_1 = \mathbf{x}(H)$. We are looking for the next Macro-step $\mathbf{x}_2 = \mathbf{x}(2H)$. In analogy to the leap frog method, we would like to find the value for $\mathbf{x}_2$ using $\mathbf{x}_0$, $\mathbf{x}_1$ and the derivative at the middle point $\mathbf{x}_1$, $\langle \dot{\xi} \rangle_\eta (H)$. In particular, any reversible explicit scheme cannot use the derivatives at $\mathbf{x}_0$, $\langle \dot{\xi} \rangle_\eta (0)$. Figure 3 depicts a reversible Macroscopic two-step solver.

Following the discussion above, we consider the change in a slow variable ξ between $\mathbf{x}_0$ to $\mathbf{x}_2$

$$\Delta^0\xi = \xi(\mathbf{x}_2) - \xi(\mathbf{x}_0) = \int_\gamma \nabla\xi(\mathbf{x}) \cdot d\mathbf{x}, \tag{3.6}$$

where γ is a smooth curve starting at $\mathbf{x}_0$ and ending at $\mathbf{x}_2$. Integrating along the solution of the ODE, $\mathbf{x}(t)$, yields

$$\Delta^0\xi = \int_0^{2H} \dot{\xi}(\mathbf{x}(\tau)) d\tau = \int_0^{2H} \langle \dot{\xi}(\mathbf{x}) \rangle_\eta (\tau) d\tau + O(He) = 2H \langle \dot{\xi}(\mathbf{x}) \rangle_\eta (H) + O(H^3). \tag{3.7}$$

On the other hand, we expand $\xi(\mathbf{x})$ around the middle point $\mathbf{x}_1$

$$\xi(\mathbf{x}_1 + \delta\mathbf{x}) = \xi(\mathbf{x}_1) + \mathbf{v} \cdot \delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^T A \delta\mathbf{x} + O(|\delta\mathbf{x}|^3),$$

where $\mathbf{v} = \nabla\xi(\mathbf{x}_1)$ and $A = D^2\xi(\mathbf{x}_1)$ is the Hessian of $\xi(\mathbf{x})$ evaluated at $\mathbf{x}_1$. We have

$$\xi(\mathbf{x}_2) = \xi(\mathbf{x}_1) + \mathbf{v}\cdot(\mathbf{x}_2 - \mathbf{x}_1) + \frac{1}{2}(\mathbf{x}_2 - \mathbf{x}_1)^T A(\mathbf{x}_2 - \mathbf{x}_1) + O(|\mathbf{x}_2 - \mathbf{x}_1|^3)$$

and

$$\xi(\mathbf{x}_0) = \xi(\mathbf{x}_1) + \mathbf{v}\cdot(\mathbf{x}_0 - \mathbf{x}_1) + \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_1)^T A(\mathbf{x}_0 - \mathbf{x}_1) + O(|\mathbf{x}_0 - \mathbf{x}_1|^3).$$

Hence,

$$\Delta^0\xi = \xi(\mathbf{x}_2) - \xi(\mathbf{x}_0) = \mathbf{v}\cdot(\mathbf{x}_2 - \mathbf{x}_0) + O(|\mathbf{x}_2 - \mathbf{x}_1|^2 + |\mathbf{x}_0 - \mathbf{x}_1|^2). \tag{3.8}$$

Assuming that $\xi(t)$ is Lipshitz in the domain of interest, $\mathcal{D}_0$, we have that $|\mathbf{x}_2 - \mathbf{x}_1|$ and $|\mathbf{x}_0 - \mathbf{x}_1|$ are both of order H. Note that, in general, the second order term does not cancel. Therefore, comparing (3.7) and (3.8) yields, to order H^2,

$$2H\langle\dot{\xi}(\mathbf{x})\rangle_\eta(H) = \mathbf{v}\cdot(\mathbf{x}_2 - \mathbf{x}_0). \tag{3.9}$$

In order to achieve a local truncation error as small as possible, we look for the solution $\mathbf{x}_2$ such that $|\mathbf{x}_2 - \mathbf{x}_1|$ is minimal. The reversibility of the algorithm hinges on the fact that (3.9) is anti-symmetric with respect to switching $\mathbf{x}_0$ and $\mathbf{x}_2$ and that both $|\mathbf{x}_2 - \mathbf{x}_1|$ and $|\mathbf{x}_0 - \mathbf{x}_1|$ are minimal. Since (3.9) is a linear system, $\mathbf{x}_2$ can be easily obtained using singular value decomposition. This is equivalent to using the naive approach (2.7). The solution of (3.9) is unique since it is a least squares problem. Geometrically, $\mathbf{x}_2 - \mathbf{x}_0$ is the point on the hyperplane with normal direction v, shifted from $\mathbf{x}_0$ by $2H\langle\dot{\xi}(\mathbf{x})\rangle_\eta(H)\mathbf{v}/|\mathbf{v}|^2$ which is closest to $\mathbf{x}_0$.

The numerical scheme is reversible, since, given initial conditions, $\mathbf{x}_0$ and $\mathbf{x}_1$, the micro-solver is used to approximate the force at $\mathbf{x}_1$, $\langle\dot{\xi}(\mathbf{x})\rangle_\eta(H)$, and we solve (3.9) for $\mathbf{x}_2$. The process can be reversed to take $\mathbf{x}_2$ and $\mathbf{x}_1$, approximate the force starting at the $\mathbf{x}_1$, and solve (3.9) for $\mathbf{x}_0$.

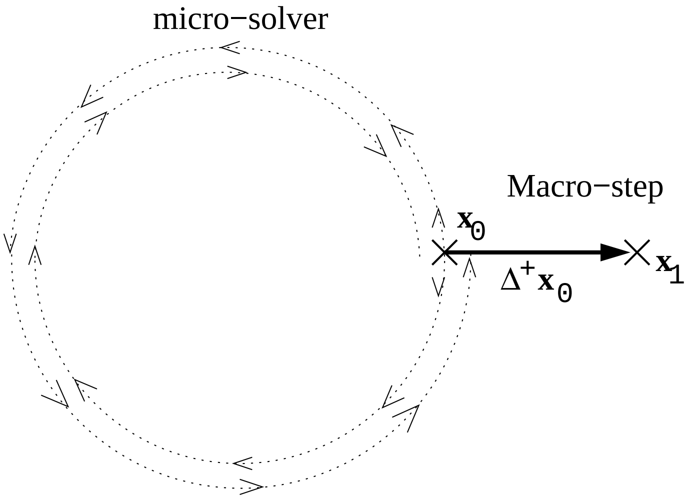

Figure 2: A forward-Euler type Macro-step for the expanding spiral (3.1).

As discussed above, the sequence of microscopic states $\mathbf{x}_0, \ldots, \mathbf{x}_N$ is generated in a reversible manner in the sense that, given $\mathbf{x}_N$ and $\mathbf{x}_{N-1}$, the algorithm can be traced back to obtain $\mathbf{x}_0$ and $\mathbf{x}_1$ up to round off errors. However, the method is not reversible for the corresponding slow variables $\xi_0 = \xi(\mathbf{x}_0), \ldots, \xi_N = \xi(\mathbf{x}_N)$. Starting with macroscopic states, ξ_N and ξ_{N-1}, requires identification of new microscopic states,

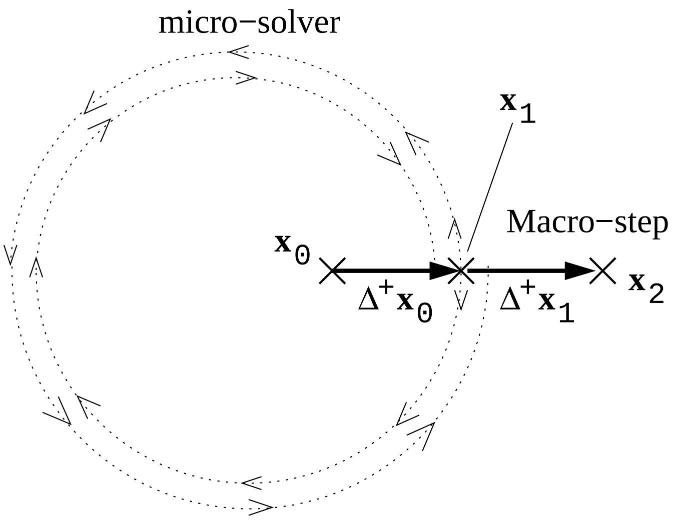

Figure 3: A leap-frog type Macro-step for the expanding spiral (3.1).

$\mathbf{x}_N$ and $\mathbf{x}_{N-1}$, such that $\xi_N = \xi(\mathbf{x}_N)$ and $\xi_{N-1} = \xi(\mathbf{x}_{N-1})$. Tracing back to $t = 0$, the new value for ξ_0 will be different than the initial $\xi(\mathbf{x}_0)$ by an order H. The lack of reversibility for the slow variables is also apparent in the fact that the order of accuracy is even. The second order correction does not vanish, as expected for reversible schemes.

Using (3.9), the error in each Macro step is of order H^2. Hence, the global error of the Macro-solver is of order H. In order to obtain a second order method, we expand the slow variable $\xi(\mathbf{x}_1 + \delta\mathbf{x})$ to second order in $\delta\mathbf{x}$

$$\xi(\mathbf{x}_1 + \delta\mathbf{x}) = \xi(\mathbf{x}_1) + \mathbf{v} \cdot \delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x} \cdot A\delta\mathbf{x} + O(\delta^3\mathbf{x}). \tag{3.10}$$

Using (3.6), (3.7) and (3.10) yields an equation for $\mathbf{x}_2$ that is accurate to order H^3

$$2H\langle\dot{\xi}(\mathbf{x})\rangle_\eta(H) = (\mathbf{x}_2 - \mathbf{x}_0) \cdot (\mathbf{v} - A\mathbf{x}_1) + \frac{1}{2}\mathbf{x}_2 \cdot A\mathbf{x}_2 - \frac{1}{2}\mathbf{x}_0 \cdot A\mathbf{x}_0. \tag{3.11}$$

As before, in order to have a reversible scheme, we look for a solution that is closest to the middle point $\mathbf{x}_1$. With several slow variables $\xi^{(1)}, \ldots, \xi^{(r)}$, $\mathbf{x}_2$ is the minimum of $|\mathbf{x}_2 - \mathbf{x}_1|^2$ under the constraint (3.11) for each of the slow variables. Since (3.11) is a quadratic hypersurface in $\mathbf{x}_2$ and $\mathbf{x}_0$, the sought constrained minimum exists and is unique unless $\mathbf{x}_2$ is on the medial axis of the hypersurface. It can be shown that for $\langle\dot{\xi}(\mathbf{x})\rangle_\eta(H) \neq 0$ this is not possible for small enough values of H. Hence, the scheme is reversible. Using Lagrange multipliers, $\mathbf{x}_2$ satisfies

$$\begin{cases} (\mathbf{x}_2 - \mathbf{x}_0) \cdot (\mathbf{v} - A\mathbf{x}_1) + \frac{1}{2}\mathbf{x}_2 \cdot A\mathbf{x}_2 - \frac{1}{2}\mathbf{x}_0 \cdot A\mathbf{x}_0 = \Delta^0\xi^{(k)}, \ k = 1 \ldots r \\ 2\mathbf{x}_2 - 2\mathbf{x}_1 + \sum_{k=1}^r \lambda_k [\mathbf{v} + A(\mathbf{x}_2 - \mathbf{x}_1)] = 0, \end{cases} \tag{3.12}$$

where $\Delta^0\xi^{(k)} = 2H\langle\dot{\xi}^{(k)}(\mathbf{x})\rangle_\eta(H)$ and $\lambda_1 \ldots \lambda_r$ are the Lagrange multipliers. For systems of coupled oscillators, the slow variables correspond to amplitudes and the relative phase between the oscillators. In [2, 3], we show that for periodic systems there exists $r = d - 1$ functionally independent slow variables. Hence, (3.12) is a quadratic system of $2d-1$ equations and $2d-1$ unknowns. In the examples appearing at the next Section, (3.12) is solved using Newton-Raphson with a required accuracy ϵ. Taking the second order approximation (3.9) and $\lambda_1 = \cdots = \lambda_r = 0$ is a good initial guess. Since the system is quadratic, convergence is usually rapid.

We summarize our method with the following algorithm. Notations are the same as in Section 2.

1. Initial conditions: $\mathbf{x}(0) = \mathbf{x}_0$ and $n = 0$.
2. Force estimation:
   (a) micro-simulation: solve (1.1) in $[t_n - \eta/2, t_n + \eta/2]$ with initial conditions $\mathbf{x}(t_n) = \mathbf{x}_n$.
   (b) Averaging: approximate $\dot{\xi}(t_n)$ by $\langle \dot{\xi} \rangle_\eta (t_n)$.
3. Macro-step: evolving ξ and reconstructing $\mathbf{x}_{n+1}$ (leap frog example)
   solve $G(\mathbf{x}_n, 2H\langle \dot{\xi} \rangle_\eta (t_n); \mathbf{x}_{n-1}, \mathbf{x}_{n+1}) = 0$ for $\mathbf{x}_{n+1}$, where G is given by (3.12).
4. $n = n+1$. Repeat steps (2) and (3) to time T.

Finally, we remark that higher order quadrature methods can be constructed in a similar fashion using two or more steps for approximating (3.7) and additional terms in the Taylor expansion (3.10).

4 Examples

In this Section we apply the reversible HMM algorithm described above to several model systems. The main purpose of the examples is to show the advantages of time reversibility in he multiscale setting.

4.1 The inverted pendulum

The following example considers a pendulum with a rigid arm that is attached at one of its ends to a mechanical motor. The setup is depicted in Figure 4. The motor causes the point of suspension of the arm to vibrate up and down with amplitude ϵ and frequency ϵ^{-1}. Surprisingly, the fast vibrations of the motor can cause the pendulum to oscillate slowly (with a $O(1)$ frequency) around the inverted position, in which its arm is pointing up. Denoting by θ the angle between the pendulum arm and the upward direction, the equation of motion for the system becomes

$$l\ddot{\theta} = \left[g + \epsilon^{-1}\sin(2\pi\epsilon^{-1}t)\right]\sin\theta, \tag{4.1}$$

where θ denotes the angle between the arm and the upward direction, l is the arm's length and g is the gravitational constant [25]. Rewriting (4.1) as a first order autonomous system yields an ODE of the form

$$\begin{cases} \dot{\theta}_1 & = \theta_2 \\ \dot{\theta}_2 & = l^{-1}(g + \epsilon^{-1}\psi_1)\sin\theta_1 \\ \dot{\psi}_1 & = 2\pi\epsilon^{-1}\psi_2 \\ \dot{\psi}_2 & = -2\pi\epsilon^{-1}\psi_1 \end{cases} \tag{4.2}$$

In [2] we describe a variational numerical method for identifying the slow variable for (4.2). The method identifies three slow variables that constitute a slow atlas:

$$\begin{aligned} \xi^{(1)} &= \theta_1 \\ \xi^{(2)} &= \psi_1^2 + \psi_2^2 \\ \xi^{(3)} &= \theta_2 + (2\pi l)^{-1}\psi_2 \sin\theta_1. \end{aligned} \tag{4.3}$$

Indeed, it is easily verified that $(d/dt)\xi^{(k)}(\mathbf{x}(t))$ is bounded independent of ϵ for $k=1,2,3$. Figure 5a depicts the numerical HMM solution for (4.2) with $g=0.1$ and $l=0.05$ using the reversible, second order Macro-solver (3.11). Simulation parameters are $\epsilon=10^{-5}$, $h=\epsilon/25$, $H=0.25$, $\eta=6.2\epsilon$ and the exponential kernel (2.9). The Newton-Raphson algorithm for solving (3.12) never requires more than two iterations. Initial conditions are $\theta_1(0)=0$, $\theta_2(0)=-0.4$, $\psi_1(0)=0$ and $\psi_2(0)=1$. The value for $\xi^{(2)}$ is practically constant with an error that is less than $10^{-8}\%$. Invariance of quadratic constants of motion is a typical advantage of reversible methods. With the above parameters the HMM algorithm runs over 5000 times faster than Verlet. Figure 5b depicts a similar numerical HMM solution for (4.2) using the exact same parameters, but with a Macro-solver applying the mid-point rule. All other parameters are the same. Although both methods are second order accurate, the errors using the midpoint rule re visibly larger. Additional approaches for applying the HMM strategy on this example can be found at [6, 29, 31].

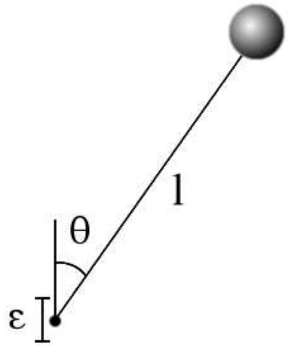

Figure 4: The inverted pendulum has a rigid arm which is attached to a motor that is vibrating fast. The centrifugal force pulls the arm upwards.

4.2 Fermi-Pasta-Ulam

The Fermi-Pasta-Ulam model [14] is a one dimensional system of unit mass particles connected by springs. The springs alternate between stiff linear and soft non-linear ones [18]. The model is derived from the following Hamiltonian

$$H=\frac{1}{2}\sum_{i=1}^{2k}p_i^2+\frac{1}{4}\epsilon^{-2}\sum_{i=1}^{k}(q_{2i}-q_{2i-1})^2+\sum_{i=0}^{k}(q_{2i+1}-q_{2i})^4. \tag{4.4}$$

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

$$x_i=\epsilon^{-1}(q_{2i-1}-q_{2i})/\sqrt{2}\,, \quad v_i=(p_{2i-1}-p_{2i})/\sqrt{2}, \tag{4.5}$$

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

$$y_i=(q_{2i-1}+q_{2i})/\sqrt{2}\,, \quad u_i=(p_{2i-1}+p_{2i})/\sqrt{2}, \tag{4.6}$$

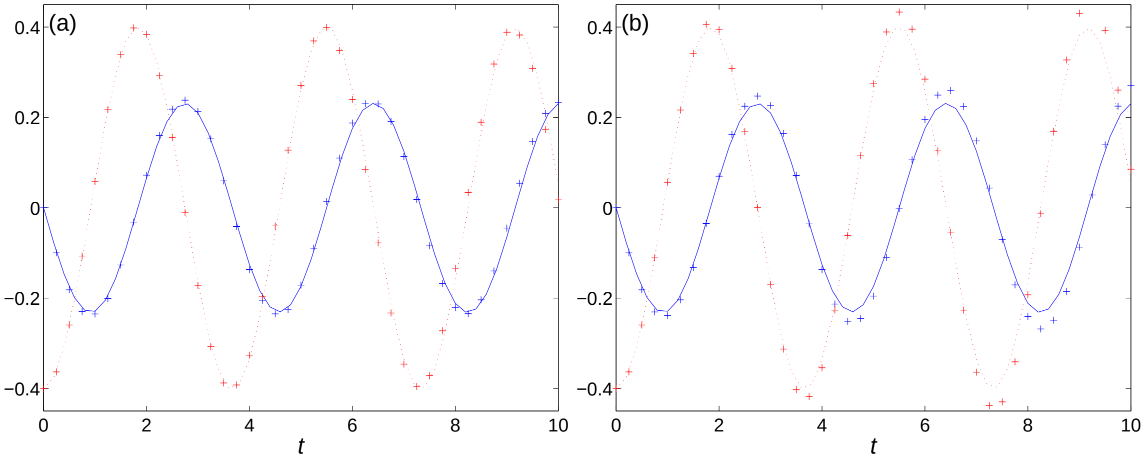

Figure 5: (a) Comparison of the HMM approximation for the solution of (4.2) to the Verlet method with step size of order ϵ. Solid curve: $\xi^{(1)}=\theta_1$, dotted curve: $\xi^{(3)}=\theta_2+(2\pi l)^{-1}\psi_2\sin\theta_1$. $\xi^{(2)}$ is constant. (b) The same system with a Macro-solver applying the mid-point rule. All other parameters are the same. With the mid-point rule errors are visibly larger.

Defining $y_0=x_0=y_{2k+1}=x_{2k+1}=0$, the equations of motion can be written as

$$\begin{cases} \dot{y}_i &= u_i \\ \dot{x}_i &= \epsilon^{-1}v_i \\ \dot{u}_i &= -(y_i-\epsilon x_i-y_{i-1}-\epsilon x_{i-1})^3+(y_{i+1}-\epsilon x_{i+1}-y_i-\epsilon x_i)^3 \\ \dot{v}_i &= -\epsilon^{-1}x_i+(y_i-\epsilon x_i-y_{i-1}-\epsilon x_{i-1})^3+(y_{i+1}-\epsilon x_{i+1}-y_i-\epsilon x_i)^3. \end{cases} \tag{4.7}$$

As discussed in [2], the slow atlas for the system consists of $4k-1$ 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, $I_i=x_i^2+v_i^2$, $i=1\dots k$. Finally, the relative phases between the different stiff springs, $\phi_i=x_1x_i+v_1v_i$, $i=2\dots k$. Any other function $\alpha(\mathbf{x})$ which is slow under the dynamics of (4.7) can be written as a function of the $4k-1$ variables described above.

On the $O(1)$ time scale the energy of the stiff springs and their relative phases are fixed, while the degrees of freedom that correspond to the soft springs oscillate in a complicated, non-harmonic way. On the $O(\epsilon^{-1})$ time scale the dynamics becomes more interesting as the energies I_i begin to change [14, 18]. The purpose of this example is to demonstrate the benefits of the reversible algorithm. Indeed, with non-reversible Macro-solvers the algorithm suffers from relatively high energy dissipation and the method is unpractical for computations on the $O(\epsilon^{-1})$ time scale. The reversible solver greatly improves energy conservation and a posteriori error analysis suggests that it is convergent. While it is possible to construct a method that works on the correct, $O(\epsilon^{-1})$ time scale, this is beyond the scope of the current paper.

Figure 6a depicts our results for a system with three stiff springs, $k=3$. Initial conditions are $x_1=-1$, $y_1=-0.5$, $y_1=u_1=x_2=1$, $v_3=-0.5$ and zero otherwise. Fixing $\epsilon=10^{-3}$, simulation parameters were varied until amplitude values changed by $5-10\%$. It was particularly difficult to get convergence using the standard Verlet method.

HMM parameters are $h=\epsilon/50$, $H=0.1$, $\eta=60\epsilon$ and the exponential kernel (2.9). The second order accurate reversible HMM (3.12) is compared to the Verlet solution with step size $\epsilon/200$. Note the significantly smaller step size required for Verlet. Figure 6b depicts the drift in the total energy of the entire system, which is smaller than 0.4%, even though our method does not guarantee convergence on the ϵ^{-1} time scale.

The Newton-Raphson method used for solving the equations obtained by the Lagrange multipliers, (3.12), becomes inefficient if the partial derivatives matrix of the right hand side of (3.12) is close to singular. In principle, one should then use a different method for solving these equations. Since in this example we are mostly interested in demonstrating energy conservation, we bypass this difficulty by integrating the full system (4.7) using Verlet for a time H whenever this problem occurs. In practice, the Verlet method is used in a few segments whose total length is less than 1% of $[0,T]$ and therefore does not reduce the efficiency considerably. In addition, these segments improve the stability of the algorithm as they smooth out oscillations between even and odd steps. We stress that the difficulties discussed above only occur on the long, $O(\epsilon^{-1})$ time scale.

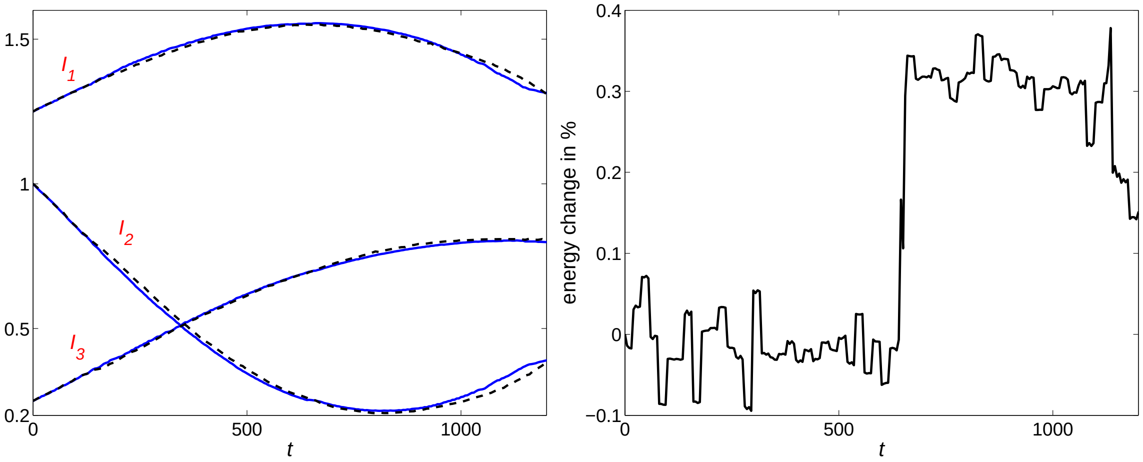

Figure 6: (a) Comparison of the HMM approximation for the solution of the Fermi-Pasta-Ulam equations of motion (4.7) to the one obtained using the Verlet method. (b) Energy dissipation with reversible HMM.

5 Conclusion

Previously, we have proposed an approach for identifying a change of variables that decomposes a vector field into its fast and slow constituents [2, 3]. The decomposition is used in an HMM algorithm that efficiently integrates the slow parts of the dynamics without fully resolving the fast parts over the computed time interval. The algorithm applies a different integrator to each of the time scales in the problem. In this paper we further develop this approach and describe a method in which both the integrators and the feed back between the different scales are implemented in a time reversible way. As a result, we obtain a numerical scheme in which the sequence of microscopic states is reversible. The method is not reversible for the slow variables. A particular set of slow variables does not correspond to a unique microscopic state. As a result, given the final slow states ξ_{N-1} and ξ_N, it is necessary to find $\mathbf{x}_{N-1}$ and $\mathbf{x}_N$ such that $\xi_{N-1}=\xi(\mathbf{x}_{N-1})$ and $\xi_N=\xi(\mathbf{x}_N)$. This reconstruction procedure ca only be preformed approximately. Hence, one cannot trace back to ξ_0 exactly. This is consistent with the fact that the dynamics of the slow variables is not reversible in the sense of Hamiltonian systems.

We demonstrate that the new approach enjoys many of the benefits of traditional time reversible integrators such as low energy dissipation. These properties are critical for integrating Hamiltonian systems over long time periods, as was demonstrated in the Fermi-Pasta-Ulam example.

Acknowledgments

Support from NSF through Grant DMS-0714612 is gratefully acknowledged. Tsai's research is partially supported by an Alfred P. Sloan Fellowship.

References

[1] A. Abdulle. Fourth order Chebyshev methods with recurrence relation. *SIAM J. Sci. Comput.*, 23(6):2041–2054 (electronic), 2002.

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

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

[4] Gil Ariel, Bjorn Engquist, Heinz-Otto Kreiss, and Richard Tsai. Multiscale computations for highly oscillatory problems. In Bjorn Engquist, Per Lötstedt, and Olof Runborg, editors, *LNCSE*, volume 66. Springer, 2008.

[5] V.I. Arnol'd. *Mathematical methods of classical mechanics*. Springer-Verlag, New York, 1989.

[6] P. M. Calvo and J. M. Sanz-Serna. Heterogeneous multiscale methods for mechanical systems with vibrations. submitted.

[7] 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.

[8] G. Dahlquist. Convergence and stability in the numerical integration of ordinary differential equations. *Mathematica Scandinavica*, 4:33–53, 1956.

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

[10] W. E and B. Engquist. The heterogeneous multiscale methods. *Commun. Math. Sci.*, 1(1):87–132, 2003.

[11] W. E, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden. Heterogeneous multiscale methods: A review. *Comm. Comput. Phys.*, 2:367–450, 2007.

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

[13] 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.

[14] 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.

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

[16] C. W. Gear and I. G. Kevrekidis. Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum. *SIAM J. Sci. Comput.*, 24(4):1091–1106 (electronic), 2003.

[17] 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.

[18] 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.

[19] E. Hairer and G. Wanner. *Solving ordinary differential equations. II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, 1996.

[20] Sanz-Serna J.M., G. Ariel, and R. Tsai. Multiscale methods for stiff and constrained mechanical systems. *J Comp. Phys.* submitted.

[21] H.-O. Kreiss. Difference methods for stiff ordinary differential equations. *SIAM J. Numer. Anal.*, 15(1):21–58, 1978.

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

[23] V. I. Lebedev and S. A. Finogenov. The use of ordered Čebyšhev parameters in iteration methods. *Ž. Vyčisl. Mat. i Mat. Fiz.*, 16(4):895–907, 1084, 1976.

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

[25] M. Levi. Geometry and physics of averaging with applications. *Physica D*, 132:150–164, 1999.

[26] J.E. Marsden and M. West. Discrete mechanics and variational integrators. *Acta Numerica*, pages 357–514, 2001.

[27] R.L. Petzold, O.J. Laurent, and Y. Jeng. Numerical solution of highly oscillatory ordinary differential equations. *Acta Numerica*, 6:437–483, 1997.

[28] J. A. Sanders and F. Verhulst. *Averaging Methods in Nonlinear Dynamical Systems*, volume 59 of *Applied Mathematical Sciences*. Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1985.

[29] J. M. Sanz-Serna. Modulated fourier expansions and heterogeneous multiscale methods. *IMA J. Numer. Anal.* to appear.

[30] R.E. Scheid. The accurate numerical solution of highly oscillatory ordinary differential equations. *Mathematics of Computation*, 41(164):487–509, 1983.

[31] R. Sharp, Y.-H. Tsai, and B. Engquist. Multiple time scale numerical methods for the inverted pendulum problem. In *Multiscale methods in science and engineering*, volume 44 of *Lect. Notes Comput. Sci. Eng.*, pages 241–261. Springer, Berlin, 2005.

[32] E. Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Comm. Math. Sci.*, 1:385–391, 2003.