

A MULTISCALE METHOD FOR HIGHLY OSCILLATORY ORDINARY DIFFERENTIAL EQUATIONS WITH RESONANCE

IN MEMORY OF GERMUND DAHLQUIST

GIL ARIEL, BJORN ENGQUIST, AND RICHARD TSAI

ABSTRACT. A multiscale method for computing the effective behavior of a class of stiff and highly oscillatory ordinary differential equations (ODEs) is presented. The oscillations may be in resonance with one another and thereby generate hidden slow dynamics. The proposed method relies on correctly tracking a set of slow variables whose dynamics is closed up to ϵ perturbation, and is sufficient to approximate any variable and functional that are slow under the dynamics of the ODE. This set of variables is detected numerically as a preprocessing step in the numerical methods. Error and complexity estimates are obtained. The advantages of the method is demonstrated with a few examples, including a commonly studied problem of Fermi, Pasta, and Ulam.

1. INTRODUCTION

Solutions of ordinary differential equations (ODEs) often involve a wide range of time scales. In many problems, one is only interested in the slow dynamics, or on the long-time behavior of the solutions. However, it is often the case that fast oscillations, or an otherwise small perturbation build up to an observable effect that cannot be neglected. A typical example, suggested by Germund Dahlquist, is the drift path of a mechanical alarm clock when it set off on a hard surface¹. The rapid vibrations of the clock's arm are not precisely symmetric. Together with the heavy, inhomogeneous internal structure and the interaction with the surface these oscillations can cause the clock to drift slowly in a complicated trajectory that is difficult to calculate or predict. Moreover, this drift seems to be deterministic and does not resemble a random walk.

Within the large literature considering dynamical systems that evolve on two or more well separated time scales, analytic averaging techniques [3, 33] have been found to be one of the methods of choice for providing approximate solutions. In this paper such an averaging theorem is used to construct a numerical method for integrating stiff ODE systems that may include oscillatory and resonant components. Our method follows the framework of the heterogeneous multiscale method (HMM) [9, 10, 12].

¹A short clip showing the trajectory of a vibrating alarm clock is available at www.ma.utexas.edu/CNA/alarm_clock/alarm.htm

Consider an ODE system of the general form

$$(1.1) \quad \dot{\mathbf{x}} = \epsilon^{-1}f(\mathbf{x}) + g(\mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x}_0,$$

where $0 < \epsilon \leq \epsilon_0$ is a small parameter that characterizes the separation of time scales in the problem. In this paper we consider solutions in a bounded domain $\mathbf{x} \in \mathcal{D}_0 \subset \mathbb{R}^d$ and in a bounded time interval $I = [0, T]$. We assume that the solution of (1.1) exists in I . Although the solution of (1.1), $\mathbf{x}(t; \epsilon)$, depends on the parameter ϵ , this dependence is suppressed and for short hand we often write $\mathbf{x}(t) \equiv \mathbf{x}(t; \epsilon)$.

Typically, the fast dynamics in equations such as (1.1) is one of two types. The first are modes that are attracted to a low dimensional manifold in ϵ time scale. These modes are called transient or dissipative modes. The second type are modes that oscillate with a frequency that is inversely proportional to ϵ . One of the main difficulties in numerical integration of (1.1) using explicit methods is that stability requirements force a step size that is of order ϵ . This generally implies that the computational complexity of such explicit methods for a integrating (1.1) over a fixed time T is at least of the order of ϵ^{-1} .

Several different approaches have been suggested, each appropriate to some class of ODEs. Dahlquist laid down the fundamental work for designing linear multi-step methods [5, 6, 7, 8] and studied their stability properties. Problems with fast transients can be optimally solved by implicit schemes [5, 19, 24]. The Chebyshev methods [1, 28] as well as the projective integrator approach [16] provide stable and explicit computational strategies for this class of problems in general. For harmonic oscillatory problems, traditional approaches attempt to filter out or fit fast, $O(\epsilon^{-1})$ oscillations to some known functions in order to reduce the complexity, e.g. [15, 25, 34], or use some notion of Poincaré map to determine slow changes in the orbital structure [17, 32]. 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 [18] and [29] for more extensive list of literatures. 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 [14] as the cost may be even more than dealing with the fast oscillations that appear in the systems. Using matrix exponentials and the [21], we can effectively compute oscillatory solutions to a class of problems with leading harmonic oscillations.

For problems in meteorology [4, 26] or celestial mechanics [27], it is possible to carefully prepare suitable initial conditions near accurate observational data so that the fast modes in the system will not be exerted, and thus long time solutions can be computed accurately using large time steps.

The different methods described above vary not only in scope and underling assumptions, but also in their approach to broader questions. To name a few, how to characterize or capture the "slow constituents" of the dynamics which may involve non-trivial functionals of the state variable \mathbf{x} ? How can one obtain slowly varying observables out of trajectories that evolve on a faster time scale? Which

observables should the multiscale integrator approximate and in what sense? Finally, which variables or observables are essential to the algorithm and which are not? This question is related to the closure problem in the formulation of averaged equations.

In many examples, it is not clear how to characterize the slow parts of the dynamics in systems such as (1.1). To this end we define slow variables or observables as follows.

Definition 1.1. Let $\mathbf{x}(t) = \mathbf{x}(t; \epsilon) \in \mathcal{D}_0$ denote the solution of (1.1) for some initial conditions.

- A smooth function $a(t) = a(t; \epsilon)$ is said to be slow if $|da/dt| \leq C_0$, for some constant C_0 independent of ϵ in $t \in I$.
- A smooth function $\alpha(\mathbf{x}) : \mathcal{D}_0 \mapsto \mathbb{R}$ is said to be a slow variable with respect to $\mathbf{x}(t)$ if

$$(1.2) \quad \left| \frac{d}{dt} \alpha(\mathbf{x}(t)) \right| \leq C_0, \quad t \in I$$

for some constant C_0 independent of ϵ .

- A bounded functional $\beta : C^1(\mathcal{D}_0) \times [0, T] \mapsto \mathbb{R}$ is said to be a slow observable with respect to $\mathbf{x}(t)$ if

$$(1.3) \quad \bar{\beta}(t) = \int_0^t \beta(\mathbf{x}(\tau), \tau) d\tau,$$

or

$$(1.4) \quad \langle \beta(\mathbf{x}) \rangle_\eta(t) = \int_{t-\eta}^{t+\eta} K_\eta(t-\tau) \beta(\mathbf{x}(\tau), \tau) d\tau,$$

where $K_\eta(\cdot) = \eta^{-1} K(\cdot/\eta)$ for some kernel function $K \in C^1$ with support on $[-1, 1]$ and $\int_{-1}^1 K(t) dt = 1$. Functionals of the second kind will also be referred to as local time averages.

In section 4 we show that under appropriate scaling of η with ϵ , $(d/dt)\langle \beta \rangle_\eta(t)$ is indeed slow.

Next we define a concept which we call "effective closure." It is a necessary condition for the macroscopic slow variables in our schemes.

Definition 1.2. We say that the dynamics of ξ is effectively closed in $[0, T]$ with respect to ϵ if ξ satisfies an equation of the form

$$\frac{d\xi}{dt} = f_I(\xi, t) + \epsilon f_{II}(\xi, t, z(t)), \quad \frac{dz}{dt} = g(\xi, z, t)$$

in a time interval $[0, T]$, where $T < \infty$ is independent of ϵ , and f_I, f_{II} , and g are smooth and bounded in $\mathcal{D}_0 \times [0, T]$.

In the next section it is shown that if a sufficient number of independent slow variables are approximated accurately by the algorithm, then, these variables are effectively closed. Furthermore, the same variables can be used to consistently approximate any other slow variable and functional. The algorithm described in the next section indeed achieves this goal and the accuracy of the different approximations is calculated.

In this paper we propose an HMM algorithm that first construct, by a numerical procedure, a transformation $\mathbf{x} \rightarrow \xi(\mathbf{x})$ such that $\xi(\mathbf{x}(t))$ are slow for all solution of (1.1) $\mathbf{x}(t)$ and its dynamics are effectively closed. This means that, for a given $\mathbf{x}(0)$ and the corresponding evolution $\mathbf{x}(t)$, $\xi(t) := \xi(\mathbf{x}(t))$ is well described for small ϵ by an effective equation of the form

$$(1.5) \quad \dot{\xi} = F(\xi), \quad \xi(0) = \xi_0 = \xi(\mathbf{x}_0),$$

for $t \in I$. Note that we do not assume that the effective equation (1.5) is available as an explicit formula. Instead, the idea behind the HMM algorithm is to evaluate $F(\xi)$ by numerical solutions of (1.1) on significantly reduced time intervals. In this way, the HMM algorithm approximates an assumed effective equation whose form is typically unknown.

The layout of the paper is as follows. Section 2 presents a well-known averaging theorem. Based on this theorem we prove that by using an appropriate set of slow variables, all slow observables admitted by (1.1) are approximated by a class of HMM schemes. The outline of the algorithm is also described. Section 3 gives an example of a particular, limited class of ODEs, in which this set of slow variables are polynomial in \mathbf{x} . Several strategies for identifying these polynomials are suggested. Section 4 considers local time averages which form an essential part of the HMM algorithm. Section 5 describes high order explicit schemes. In Section 6 we estimate the global accuracy and complexity of the method. Section 7 presents several numerical examples. We conclude in Section 8 by suggesting different possible generalizations of our algorithm and compare to other techniques.

2. THE HMM SCHEME

We begin with a well-known averaging theorem. A proof can be constructed along the lines of Sanders and Verhulst [33]; e.g. Theorem 3.2.10 on page 39.

2.1. An averaging theorem. Let $(\xi(t), \phi(t), \gamma(t))$ denote the solution of

$$(2.1) \quad \begin{cases} \dot{\xi} = F(\xi, \phi, \gamma), & \xi(0) = \xi_0 \\ \dot{\phi} = \epsilon^{-1}\Omega(\xi) + G(\xi, \phi, \gamma), & \phi(0) = \phi_0 \\ \dot{\gamma} = -\epsilon^{-1}H(\gamma), & \gamma(0) = \gamma_0, \end{cases}$$

in a bounded domain $\mathcal{D} \times \mathcal{S}^1 \times \mathcal{D}_1$, where $\mathcal{D} \subset \mathbb{R}^r$ and $\mathcal{D}_1 \subset \mathbb{R}^{d-r-1}$. In this paper, we identify \mathcal{S}^1 as \mathbb{R}/\mathbb{Z} , i.e., the space \mathbb{R} modulus one, with the corresponding topology and Lebesgue measure. In this construction, the functions, F , G , H and

Ω are \mathcal{C}^1 and independent of ϵ . Additionally, we assume that $\Omega \geq m > 0$ and $\xi(t)$ remains bounded for $0 \leq t \leq T$.

The function $H(\gamma)$ is chosen such that, for all $\gamma \in \mathcal{D}_1$, $\gamma(t)$ is attracted on an ϵ time scale to an invariant manifold, \mathcal{M} , on which γ is relaxed to be of order ϵ . More specifically, it is assumed that

$$(2.2) \quad |\gamma(t)| \leq \frac{C}{D + (t/\epsilon)^l},$$

for some constants $C, D, l > 1$. Hence, γ are transient, or dissipative variables which are relaxed to zero after an initial time layer that vanishes in the limit $\epsilon \rightarrow 0$.

Let $\zeta(t)$ denote the solution of

$$(2.3) \quad \dot{\zeta} = \bar{F}(\zeta) = \int_{S^1} F(\zeta, \sigma, \gamma = 0) d\sigma, \quad \zeta(0) = \xi(0).$$

Then,

$$(2.4) \quad \sup_{0 \leq T_0(\epsilon) \leq t \leq T} |\xi(t) - \zeta(t)| \leq \epsilon C T e^{L_{\bar{F}} T},$$

where $L_{\bar{F}} > 0$ is a Lipschitz constant of \bar{F} , $C > 0$ is a constant that is independent of ϵ and $T_0(\epsilon) = o(1)$ is the time required for the relaxation of the dissipative modes, which depends on the rate of decay of γ , l . This theorem suggests that simulating the averaged equation for ζ introduces an error on the order of ϵ .

The above condition on the dissipative modes can be generalized to a case in which the equation for $\gamma \in \mathbb{R}^{d-r-1}$ has the form

$$\dot{\gamma} = \epsilon^{-1} g(\xi, \gamma),$$

where g is such that γ is attracted to an $r + 1$ dimensional invariant manifold, \mathcal{M}_1 , on which $g(\xi, \gamma)$ is of order ϵ . See [31] for sufficient conditions that guaranty stability and uniqueness of this manifold and a proof that γ decays to \mathcal{M}_1 exponentially fast on a time scale of the order of ϵ . For simplicity, we do not consider the interesting problem of turning points. The algorithm described in the following section automatically guaranties that all dissipative models are relaxed and that the dynamics stays close to \mathcal{M}_1 . This generalization is not discussed any further since the main interest of this paper are the oscillatory modes.

Assume that there exists a change of variables

$$\Phi : \mathcal{D}_0 \subset \mathbb{R}^d \mapsto \mathcal{D} \in \mathbb{R}^r \times \mathcal{S}^1 \times \mathbb{R}^{d-r-1}, \quad \Phi(\mathbf{x}) = (\xi(\mathbf{x}), \phi(\mathbf{x}), \gamma(\mathbf{x}))$$

such that for the solutions of (1.1), the functions $\xi(t) = \xi(\mathbf{x}(t))$, $\phi(t) = \phi(\mathbf{x}(t))$, and $\gamma(t) = \gamma(\mathbf{x}(t))$ satisfy some equations in the form (2.1). For $0 < \epsilon \ll 1$, the fast dynamics in the new coordinate system appears only in $\phi(t)$ and $\gamma(t)$. Hence, the effective behavior of the system can be described by $\zeta(t)$, which approximates $\xi(t)$.

The most important aspect of equation (2.3) is that it is closed, i.e., the time derivative of ζ is expressed in terms of ζ alone. As a result, it is possible to construct

an algorithm that only approximates ζ . Later, it is shown that if ζ is approximated accurately, then all slow variables and observables are also approximated consistently.

System (1.1) can be analyzed and computed efficiently if there exists a diffeomorphism between \mathbf{x} and (ξ, ϕ, γ) . Clearly, the choice of Φ is not unique. However, since we are only interested in the slow parts of the dynamics, it is enough to find a sufficient number of slow variables $\xi = (\xi_1, \dots, \xi_r)$ such that $\partial\xi/\partial\mathbf{x}$ has rank r . Once $\xi(\mathbf{x})$ is identified, $F(\xi) = F(\xi(\mathbf{x}))$ is easily calculated. The key step is then to relax all dissipative modes and average over ϕ to obtain the unknown function $\bar{F}(\xi)$. In section 4, we show that this can be approximated by local time averages

$$\bar{F}(\xi) \sim \langle F(\xi(\mathbf{x}(t))) \rangle_\eta.$$

$\dot{\xi}(t) = \bar{F}(\xi)$ is then used in an integrator to obtain ξ at some later time $t + H$. Efficiency is attained if $H > \eta$ since we do not need to evolve the stiff systems for $\mathbf{x}(t)$ for all time.

2.2. Algorithm. The first step in our algorithm is to numerically find the diffeomorphism Φ and identify the slow variables $\xi(\mathbf{x})$. Then, the ODE (1.1) is integrated using a two level algorithm; each level corresponds to the integration of (1.1) in a different time scale. The first is a Macro-solver, which integrates the averaged equation (2.3). 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 (1.1) using a suitable initial data. For example, suppose the Macro-solver applies a forward Euler scheme with step size H . Higher order methods are described in Section 5. Sample times of the Macro-solver are denoted t_0, \dots, t_N , where $N = T/H$. The output of the Macro-solver is denoted $\mathbf{x}_0, \dots, \mathbf{x}_N$. The output of the micro-solver with step size h , initiated at time t_n with initial conditions \mathbf{x}_n is denoted $\mathbf{x}_n^1, \dots, \mathbf{x}_n^M$, where $M = 2\eta/h$ is an even integer. The coordinates of \bar{F} are denoted $\bar{F}(\zeta) = (\bar{F}_1(\zeta), \dots, \bar{F}_r(\zeta))^T$. We also assume that the averaging kernel K is symmetric with respect to its mid-point. This implies that if $Mh \gg \epsilon$, then all transient modes practically vanish at the midpoint. The outline of the algorithm, depicted in Figure 1 for a one dimensional example and in 2 for a two dimensional case, is as follows.

(1) Construction of slow variables:

Find functions $\xi_1(\mathbf{x}), \dots, \xi_r(\mathbf{x})$ such that, for all $\mathbf{x} \in \mathcal{D}_0 \cap \{F \neq 0\}$, $|\nabla_{\mathbf{x}} \xi \dot{\mathbf{x}}| \leq C_0$ and $\text{rank}(\partial\xi/\partial\mathbf{x}) = r$. See Section 3 for details.

(2) Multiscale evolution:

(a) Initial conditions: $\mathbf{x}(0) = \mathbf{x}_0$ and $n = 0$.

(b) Force estimation:

(i) micro-simulation: Solve (1.1) in $t \in [t_n, t_n + 2\eta]$ with initial conditions \mathbf{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) Macro-step (forward Euler example):

$\mathbf{x}_{n+1} = \mathbf{x}_n^{M/2} + H\delta\mathbf{x}$, where $\delta\mathbf{x}$ is the least squares solution to the linear system

$$\delta\mathbf{x} \cdot \nabla \xi_i = \bar{F}_i(\xi) = \langle \dot{\xi}_i \rangle_\eta,$$

for all $i = 1 \dots r$.

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

Note that there is no need to actually change the original ODE (1.1) to the form used in the averaging theorem (2.1).

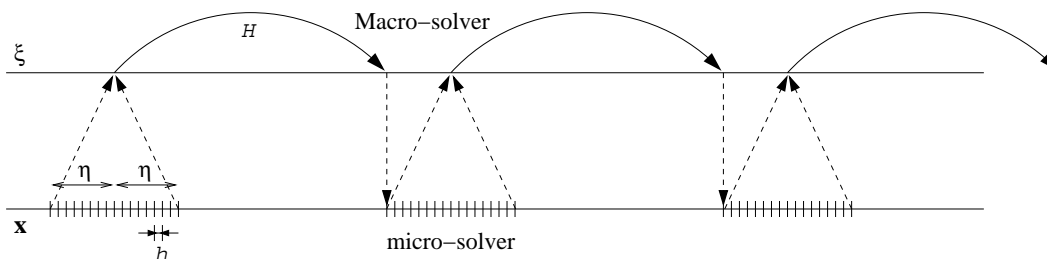


FIGURE 1. The cartoon depicts the time steps taken by the HMM scheme. At the n -th macro step, a micro-solver with step size h integrates (1.1) to approximate $\mathbf{x}(t)$ in a time segment $[t_n, t_n + 2\eta]$. This data is used to approximate $\bar{F}(\xi)$ by $\langle \xi(\mathbf{x}(t)) \rangle_\eta$. Then, the Macro-solver takes a big step of size $H\delta\mathbf{x}$, where $\delta\mathbf{x}$ is consistent with $\langle \dot{\xi} \rangle_\eta$, i.e., $\delta\mathbf{x} \cdot \nabla \xi_i = \langle \dot{\xi}_i \rangle_\eta$ for the identified slow variables ξ_i .

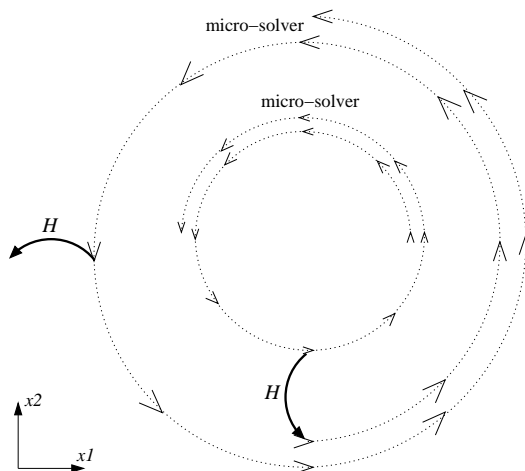


FIGURE 2. The cartoon depicts the outline of the HMM scheme for a simple two dimensional system that oscillates with a frequency of the order ϵ^{-1} and expands on an $O(1)$ time scale (c.f. Section 7.1). The dotted line depicts the trajectory obtained by the micro solver. The bold arcs depict the steps taken by the Macro-solver.

2.3. Approximation of slow quantities. With the proposed algorithm, ξ is implicitly and accurately evolved. Thus we are able to compute the slow variables in Definition 1.1.

Let $\alpha(\mathbf{x}) \in \mathbb{R}$ be a slow variable with respect to $\mathbf{x}(t)$, i.e., $|\frac{d}{dt}\alpha(\mathbf{x}(t))| \leq C_0$. Denote $\tilde{\alpha}(\xi, \phi, \gamma) = \alpha(\Phi^{-1}(\xi, \phi, \gamma))$, and $\bar{\alpha}(t) = \tilde{\alpha}(\xi(t), \phi(t), \gamma(t)) = \alpha(\mathbf{x}(t)) = \alpha(t)$. Differentiating with respect to time, we have

$$\frac{d}{dt}\alpha = \frac{d}{dt}\tilde{\alpha} = \tilde{\alpha}_\xi \cdot \dot{\xi} + \tilde{\alpha}_\phi \cdot \dot{\phi} + \tilde{\alpha}_\gamma \cdot \dot{\gamma},$$

where $\tilde{\alpha}_\xi$, $\tilde{\alpha}_\phi$ and $\tilde{\alpha}_\gamma$ denote partial derivatives with respect to ξ , ϕ and γ , respectively. This implies that if $\alpha(\mathbf{x})$ is slow, then

$$\tilde{\alpha}(\xi, \phi, \gamma) = \tilde{\alpha}(\xi, \phi_0, 0) + \epsilon C(\xi, \phi, \gamma),$$

where $C(\xi, \phi, \gamma)$ is bounded independent of ϵ . However, since α or equivalently, $\tilde{\alpha}$ is independent of ϵ , we have that $C \equiv 0$. In other words, any slow variable is a function of ξ alone. Consequently, if the algorithm approximates $\xi(t)$, then any smooth function $\tilde{\alpha}(\xi(t))$ is also approximated.

The second type of slow observables we consider are time averages of the form $\langle \beta(\mathbf{x}(\cdot)) \rangle_\eta$. The discussion above implies that

$$(2.5) \quad \langle \beta(\mathbf{x}(\cdot)) \rangle_\eta = \langle \beta(\mathbf{x}(\xi, \phi, \gamma)) \rangle_\eta \equiv \langle \tilde{\beta}(\xi, \phi, \gamma) \rangle_\eta,$$

for some function $\tilde{\beta}$. However, the variables γ decay to zero and, as shown in section 4, for sufficiently large η time averages can be approximated by averaging over the angle variable ϕ . Therefore, we have that

$$(2.6) \quad \langle \tilde{h}(\xi, \phi, \gamma) \rangle_\eta \sim \int_{S^1} \tilde{h}(\xi, \phi, \gamma = 0) d\phi \equiv B(\xi),$$

which is a function of ξ alone and hence a slow variable. The discussion above shows that the algorithm approximates the value of $\langle \beta(\mathbf{x}) \rangle_\eta$. In practice the time average $\langle \beta(\mathbf{x}) \rangle_\eta$ can be calculated from the output of the micro-simulation initiated at each Macro-step. The error introduced in approximating time averages by angular ones, and vice versa, is analyzed in Section 4.

Convergence of functionals follows from the observation that for any smooth functions $\alpha(\mathbf{x}, t)$,

$$\bar{\alpha}(t) = \int_0^t \alpha(\mathbf{x}(s), s) ds$$

is slow since $|(d/dt)\bar{\alpha}(t)| = |\alpha(\mathbf{x}(t), t)| \leq C_0$ for $\mathbf{x}(t) \in \mathcal{D}_0$.

In the simple case in which the integrand $\alpha(\mathbf{x}, t)$ is a slow variable itself, the functional $\int_0^t \alpha(\mathbf{x}(\tau), \tau) d\tau$ can be approximated by quadrature over the sample points of

the Macro-solver. In general, $\alpha(\mathbf{x}(t), t)$ can exhibit fast oscillations with frequency proportional to ϵ^{-1} . However, we observe that $\bar{\alpha}(t)$ solves the ODE

$$(2.7) \quad \begin{cases} \dot{\bar{\alpha}} &= \alpha(\Phi^{-1}(\xi, \phi, \gamma), \tau) \\ \dot{\tau} &= 1, \end{cases}$$

which complies to the form required by the averaging theorem. Therefore, $\bar{\alpha}(t)$ can be integrated by the proposed HMM scheme as a passive variable at the macroscopic level. In other words, it can be approximated by

$$\bar{\alpha}(t) \sim \int_0^t \langle \alpha(\mathbf{x}(\tau)) \rangle_\eta d\tau.$$

3. SLOW VARIABLES

The idea behind the proposed method is to find (by numerical construction) a set of slow variables so that the dynamics of the given system is decomposed into this set of slow variables, a *single* fast oscillator and dissipative modes. The slow variables share some resemblance with action-angle variables in the context of Hamiltonian dynamics.

In this section, we consider a class of equations, of a less general form than (1.1), for which it is possible to analytically classify the slow variables and describe the diffeomorphism Φ . We should mention that the HMM algorithm described in section 2 applies to the general form (1.1).

3.1. Polynomial slow variables. Consider a system of ODEs of the form

$$(3.1) \quad \begin{cases} \dot{x} &= \epsilon^{-1}Ax + f_I(x, y, z, t) \\ \dot{y} &= f_{II}(x, y, z, t) \\ \dot{z} &= \epsilon^{-1}p(x, y) + f_{III}(x, y, z, t), \end{cases}$$

where $\mathbf{x} = (x, y, z) \in \mathbb{R}^d$ with $x \in \mathbb{R}^{d_x}$, $y \in \mathbb{R}^{d_y}$ and $z \in \mathbb{R}^{d_z}$, $d_x + d_y + d_z = d$, A is a real, diagonalizable $d_x \times d_x$ matrix whose eigenvalues are non zero and either have a negative real part or are purely imaginary. Eigenvalues with negative real parts are denoted $-\lambda_1, \dots, -\lambda_l$. Purely imaginary eigenvalues appear in complex conjugate pairs and are denoted, with multiplicity, $\pm i\omega_1, \dots, \pm i\omega_r$, so $l + 2r = d_x$. Without loss of generality, we may take $\omega_k > 0$, $1 \leq k \leq r$. In addition, f_I , f_{II} and f_{III} are \mathcal{C}^1 in a bounded domain $\mathcal{D}_0 \subset \mathbb{R}^d$ to which the solution is restricted. $p = (p_1, \dots, p_{d_z})$ are polynomials such that the monomials in $p(x, y)$ are not slow. Initial conditions are $\mathbf{x}(0) = (x(0), y(0), z(0)) = (x_0, y_0, z_0)$. Note that the singular part of \dot{z} does not have to be linear, however, it may not depend on any of the coordinates of z itself.

Suppose Σ is a change of variables that diagonalizes A . We obtain the following decoupled scalar equations for $\Sigma^{-1}\mathbf{x}$

$$(3.2) \quad \begin{cases} \dot{\vartheta}_k^\pm(t) = \pm i\epsilon^{-1}\omega_k\vartheta_k^\pm(t) + f_1^\pm & k = 1 \dots r \\ \dot{\gamma}_k(t) = -\epsilon^{-1}\lambda_k\gamma_k(t) + f_2 & k = 1 \dots l. \end{cases}$$

Here, f_1^\pm and f_2^\pm are linear combinations of the components of f_I . We refer to the variable pairs $\{(\vartheta_i^+, \vartheta_i^-)\}_{k=1}^r$ as oscillators and to $\{\gamma_k\}_{k=1}^l$ as transient, or dissipative modes. Note that by hypotheses, f_1^\pm are bounded independent of ϵ , so for initial conditions away from the origin, we expect that $\vartheta_k^\pm(t) \neq 0$ for a time interval larger than the ϵ time scale.

The assumption that A is diagonalizable can be relaxed to include cases in which only the Jordan blocks corresponding to purely imaginary eigenvalues are diagonal while blocks that correspond to $-\lambda_k$ are not. This is the minimal condition to insure that the solution of (3.1) does not blow up on the ϵ time scale.

Lemma 3.1. *There exists a diffeomorphism*

$$\Phi : x \in \mathcal{U} \subset \mathbb{R}^{2r} \mapsto (\xi^{(E)}, \xi^{(\theta)}, \phi) \in \mathbb{R}^r \times \mathbb{R}^{r-1} \times S^1,$$

such that $\xi^{(E)} = (\xi_1^{(E)}, \dots, \xi_r^{(E)}) \in \mathbb{R}^r$ and $\xi^{(\theta)} = (\xi_1^{(\theta)}, \dots, \xi_{r-1}^{(\theta)}) \in \mathbb{R}^{r-1}$ are slow variables with respect to (3.1), where $\mathcal{U} = \mathbb{R}^{2r} - \{\xi_k^{(E)} \neq 0, k = 1 \dots r\}$.

The Lemma shows that one may consider $(\xi^{(E)}, \xi^{(\theta)}, \phi)$ as a new coordinates system for the subspace spanned by the oscillatory modes of A . The slow variables $\xi^{(E)}$ correspond to the total energy (kinetic + potential) of the oscillators $\{(\vartheta_i^+, \vartheta_i^-)\}_{k=1}^r$. The slow variables $\xi^{(\theta)}$ correspond to a chain of relative phases between the oscillators. They capture the relative progress of the oscillators along their period. Using $\xi^{(E)}$ and $\xi^{(\theta)}$, all the fast oscillations are driven by a *single* fast process $\phi \in S^1$.

Proof. Consider the following map $\tilde{\Phi} : \mathbb{C}^{2r} \rightarrow \mathbb{C}^{2r}$ given by $\tilde{\Phi} = (\tilde{\Phi}_1, \dots, \tilde{\Phi}_{2r}) = \tilde{\Phi}(\zeta)$ with $\zeta = (\zeta_1, \dots, \zeta_{2r})$ and

$$(3.3) \quad \begin{aligned} \tilde{\Phi}_k &= \zeta_{2k-1}\zeta_{2k}, \quad k = 1 \dots r \\ \tilde{\Phi}_{k+r} &= (\zeta_{2k})^{\omega_{k+1}}(\zeta_{2k+1})^{\omega_k} + (\zeta_{2k-1})^{\omega_{k+1}}(\zeta_{2k+2})^{\omega_k}, \quad k = 1 \dots r-1 \\ \tilde{\Phi}_{2r} &= \zeta_{2r}^{1/\omega_r}, \end{aligned}$$

where we define $\arg[z]$, the argument of z , to be in $(\pi, \pi]$. Its inverse in $\Xi_0 = \{\zeta_k \neq 0, k = 1 \dots 2r\}$ is given by

$$(3.4) \quad \begin{aligned} \zeta_{2r} &= \tilde{\Phi}_{2r}^{\omega_r} \\ \zeta_{2r-1} &= \tilde{\Phi}_r / \zeta_{2r}, \end{aligned}$$

and the rest of the variables $\{\zeta_k\}_{k=1}^{2r-2}$ are given recursively by

$$(3.5) \quad \begin{aligned} \zeta_{2k} &= \frac{1}{2} \zeta_{2k+1}^{-\omega_k} \left[\tilde{\Phi}_k^{\omega_{k+1}} + \sqrt{\tilde{\Phi}_k^{2\omega_{k+1}} - 4\tilde{\Phi}_{k+1}^{\omega_k}} \right] \\ \zeta_{2k-1} &= \tilde{\Phi}_k / \zeta_{2k}. \end{aligned}$$

Next, let

$$(3.6) \quad (\xi^{(E)}, \xi^{(\theta)}, \phi) = \Phi(x) = \left(\tilde{\Phi}_1, \dots, \tilde{\Phi}_{2r-1}, (2\pi)^{-1} \arg[\tilde{\Phi}_{2r}] \right) (\Sigma^{-1}x).$$

Recall that Σ is a complex matrix that diagonalizes the real matrix A , i.e, $x = \Sigma(\vartheta_1^+, \vartheta_1^-, \dots, \vartheta_r^+, \vartheta_r^-)$. We note that the columns of Σ^{-1} consists of complex conjugate pairs. Hence, $\zeta_{2k-1}(\Sigma^{-1}(x))$ is the complex conjugate of $\zeta_{2k}(\Sigma^{-1}(x))$ for all $k = 1 \dots r$. Substituting into (3.3) we see that $\Phi(x)$ is indeed real valued, differentiable and has a differentiable inverse. We conclude that Φ is a diffeomorphism on $\{\vartheta_k^\pm \neq 0, k = 1 \dots r\}$.

Finally, differentiating with respect to time shows that $\xi^{(E)}$ and $\xi^{(\theta)}$ are slow with respect to (3.1). This is easily verified by noting that $\tilde{\Phi}(\vartheta)$ is slow with respect to (3.2). We conclude that Φ is a diffeomorphism on \mathcal{U} . □

As an example, consider the simple case of two real valued oscillators with frequency ϵ^{-1} ,

$$(3.7) \quad \begin{cases} \dot{x}_1 &= \epsilon^{-1}x_2 + f_1(\mathbf{x}) \\ \dot{x}_2 &= -\epsilon^{-1}x_1 + f_2(\mathbf{x}) \\ \dot{x}_3 &= \epsilon^{-1}x_4 + f_3(\mathbf{x}) \\ \dot{x}_4 &= -\epsilon^{-1}x_3 + f_4(\mathbf{x}) \end{cases}$$

The slow variables given by (3.6) are

$$(3.8) \quad \begin{aligned} \xi_1^{(E)} &= x_1^2 + x_2^2 \\ \xi_2^{(E)} &= x_3^2 + x_4^2 \\ \xi_1^{(\theta)} &= x_1x_3 + x_2x_4, \end{aligned}$$

which correspond to the internal energy of the two oscillators and the cosine of their relative phase. The fast variable can be taken to be $\phi = \arctan(x_2/x_1)$. However, our algorithm does not require identifying it. If the frequencies of the two oscillators (x_1, x_2) and (x_3, x_4) is different, then $\xi_1^{(\theta)}$ will have a more complicated form. For example, if the matrix A in (3.1) is real, and the ratio ω_i/ω_{i+1} is rational, then $\xi^{(\theta)}$ can be taken to be polynomial (see 7.2 for an example in which $\xi^{(\theta)}$ is cubic). This is made clear by the following lemma.

Lemma 3.2. *Suppose $m\omega_k = n\omega_{k+1}$ with integer $m, n > 0$. Then, there exists a polynomial, p , such that $p(\vartheta_k^+(\mathbf{x}), \vartheta_{k+1}^-(\mathbf{x}))$ is slow with respect to $\mathbf{x}(t)$, the solution of (3.1), for all initial conditions.*

Proof. It is easily verified that

$$(3.9) \quad \tilde{\xi}^{(\theta)} = (\vartheta_k^+)^m (\vartheta_{k+1}^-)^n + (\vartheta_k^-)^m (\vartheta_{k+1}^+)^n,$$

is indeed a real valued slow variable. Since the eigenfunctions $\{\vartheta_k^\pm\}_{k=1}^r$ are obtained by diagonalization, $\xi^{(\theta)}(x) = \tilde{\xi}^{(\theta)}(\Sigma^{-1}x)$ is polynomial in x . \square

The next lemma states that the z variables will result in some new slow variables of the polynomial form. For convenience, we introduce multi-index notation of vectors in \mathbb{R}^d . Let $\mathbf{n} = (n_1, n_2, \dots, n_d)$ denote a d -tuple of non-negative integers; $|\mathbf{n}| = n_1 + n_2 + \dots + n_d$. For $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$ we denote $\mathbf{x}^{\mathbf{n}} = x_1^{n_1} x_2^{n_2} \dots x_d^{n_d}$. Using this notation, an m 'th degree polynomial of \mathbf{x} can be written as

$$p(\mathbf{x}) = \sum_{|\mathbf{n}| \leq m} c_{\mathbf{n}} \mathbf{x}^{\mathbf{n}}$$

where $\{c_{\mathbf{n}}\}_{|\mathbf{n}| \leq m}$ is a family of real coefficients that determines $p(\mathbf{x})$. Note that vectors in \mathbb{R}^d or \mathbb{Z}^d are denoted with bold letters while lower dimensional vectors, for example x and y , are not.

Lemma 3.3. *Given a system of (3.1) with the standing hypothesis. There exists a polynomial $Q(x, y)$ of degree m such that $\xi^{(z)} = z - Q(x, y)$ is slow with respect to $(x(t), y(t), z(t))$.*

Proof. Without loss of generality, assume that $\xi^{(z)}$ is a scalar and that A is diagonal with eigenvalues $\mu_1 \dots \mu_{d_x}$. Differentiating with respect to time yields

$$(3.10) \quad \dot{\xi}^{(z)} = \dot{z} + \nabla_{(x,y)} Q(x, y) \cdot (\dot{x}, \dot{y}) = \epsilon^{-1} [p(x, y) - \nabla_x Q(x, y) \cdot Ax] + f_3(x, y, z),$$

where $\nabla_{(x,y)}$ denotes the gradient with respect to both x and y and ∇_x with respect to only x . Thus $\xi^{(z)}$ is slow if $|p(x, y) - \nabla_x Q(x, y) \cdot Ax| \leq C_1 \epsilon$. For a polynomial p as described above, one can write $p(x, y) = \sum_{|k|+|l| \leq m, |k| \geq 1} c_{k,l} x^k y^l$. Then,

$$Q(x, y) = \sum_{|k|+|l| \leq m, |k| \geq 1, \mu \cdot k \neq 0} \frac{c_{k,l}}{\mu \cdot k} x^k y^l,$$

satisfies $\nabla_x Q(x, y) \cdot Ax = p(x, y)$ for all x and y , where, for a multi-index $k = (k_1, \dots, k_{d_x})$, we defined $\mu \cdot k = \sum_{j=1}^{d_x} \mu_j k_j$. Note that $\mu \cdot k = 0$ if and only if the monomial $x^k y^l$ is a slow variable. In this case, there are two possible options. If $x^k y^l$ vanishes, or is of order ϵ , then $\epsilon^{-1} x^k y^l$ is bounded independent of ϵ and $z - Q(x, y)$ is still slow. On the other hand, if $x^k y^l$ is of order one, then the variable z leaves \mathcal{D}_0 on a time scale of order ϵ , which is a case not considered in this paper. The proof above generalizes directly to $A = \Sigma \Lambda \Sigma^{-1}$. \square

Lemmas 3.1-3.3 yield the following theorem:

Theorem 3.4. Let $\xi^{(y)} = y$ and $\xi = (\xi^{(E)}, \xi^{(\theta)}, \xi^{(y)}, \xi^{(z)})$. Then,

$$(3.11) \quad \begin{aligned} \Phi(\mathbf{x}) &= \Phi(x, y, z) = (\tilde{\Phi}^{-1}(x), y, z - Q(x, y)) = (\gamma, \phi, \xi) \\ &= (\gamma, \phi, \xi^{(E)}, \xi^{(\theta)}, \xi^{(y)}, \xi^{(z)}) \end{aligned}$$

is a diffeomorphism on $\{\xi_i^{(E)} \neq 0\}$.

Proof. The Jacobian $\partial\Phi/\partial\mathbf{x}$ is given by

$$\frac{\partial\Phi}{\partial\mathbf{x}} = \begin{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & (\frac{\partial\Phi}{\partial x})^{-1} \end{pmatrix} & 0 & 0 \\ -\frac{\partial Q}{\partial x} & -\frac{\partial Q}{\partial y} & 1 \end{pmatrix}$$

Using Lemma 3.1, $\det(\partial\Phi/\partial\mathbf{x}) \neq 0$. □

In section 7 we consider several example systems, including the Fermi-Pasta-Ulam problem, that satisfy the assumptions of Lemmas 3.1-3.3, and hence theorem 3.4 holds.

3.2. Identifying slow variables. Theorem 3.4 constructs a diffeomorphism $\Phi : \mathbb{R}^d \rightarrow \mathbb{R}^d$ that decomposes \mathbf{x} into slow and fast variables, such that the slow variables lie in a \mathbb{R}^{2r-1+d_z} dimensional space. Previously, we concluded that the effective behavior of the system can be described by functions or functionals of these slow variables. Under our definition, the choice of slow variables is not unique and we do not necessarily have to use the ones that are defined in the previous subsection. The essential criterion is to have a sufficient number of independent slow variables $\xi = \{\xi_j\}$ such that $\text{rank}\partial\xi/\partial\mathbf{x} = 2r - 1 + d_z$ in some open set.

The general idea is to decompose the vector field locally by orthogonal projections of the right hand side f_ϵ to the subspace that is orthogonal to the gradient of the slow variables. One should then search for sufficient number of slow variables ξ such that the projection

$$P_{\nabla\xi} f_\epsilon(\mathbf{x}) := \left(I - \frac{\nabla\xi \otimes \nabla\xi}{|\nabla\xi|^2} \right) f_\epsilon(\mathbf{x})$$

cannot contain any possible slow variables with respect to $\mathbf{x}(t)$. This way, by applying the averaging theorem, the dynamics of ξ is effectively closed and can be evolved to track the effective behavior of the system. Below, we describe our rationale and strategy of using a numerical procedure to determine a sufficient set of slow variables.

Let $\xi(\mathbf{x})$ be a potential slow variable with respect to $\mathbf{x}(t)$. Following our definition for slow variables, this implies that $d\xi(\mathbf{x}(t))/dt = \nabla_{\mathbf{x}}\xi(\mathbf{x}) \cdot d\mathbf{x}/dt$, is expected to be bounded independent of ϵ . More importantly, this is expected to hold for the

dynamics that originate from initial values in an non-empty open set. Therefore, we may attempt to solve the minimization problem

$$(3.12) \quad \min_{\xi \in X} \int_{\mathcal{A} \subset \mathbb{R}^d} |\nabla_{\mathbf{x}} \xi(\mathbf{x}) \cdot \phi(\mathbf{x})|^2 d\mu_{\mathbf{x}},$$

here $\phi(\mathbf{x})$ denotes the function $d\mathbf{x}/dt$, i.e., the right hand side of (3.1), or more generally (1.1). \mathcal{A} is some open set in \mathbb{R}^d endowed with a measure $d\mu_{\mathbf{x}}$ and X is an a priori chosen function space.

Based on Lemmas 3.1-3.3, we propose to search for slow variables in the space of polynomials of the unknown variables of the original equation, \mathbf{x} . Polynomials are also used to locally approximate the potential functions in certain problems. We proceed by looking for a polynomial with no constant term

$$(3.13) \quad p(\mathbf{x}) = \sum_{1 \leq |\mathbf{n}| \leq m} c_{\mathbf{n}} \mathbf{x}^{\mathbf{n}}$$

that minimizes (3.12).

In order to choose the set \mathcal{A} and measure $d\mu_{\mathbf{x}}$ consider the following Lemma.

Lemma 3.5. *For any nonzero $a \in \mathbb{R}$ and $\mathbf{x}_0 \in \mathbb{R}^d$, consider the grid $\{\mathbf{x}_0 + a\mathbf{n} | 1 \leq |\mathbf{n}| \leq m\}$. For any point on this grid we assign a value $b_{\mathbf{n}}$. Then, there exists a unique polynomial $p(\mathbf{x}) = \sum_{1 \leq |\mathbf{n}| \leq m} c_{\mathbf{n}} \mathbf{x}^{\mathbf{n}}$ in \mathbb{R}^d of order m such that $p(\mathbf{x}_0 + a\mathbf{n}) = b_{\mathbf{n}}$ for all $1 \leq |\mathbf{n}| \leq m$.*

Proof. We wish to show that the Vandermonde system $\sum_{1 \leq |\mathbf{n}| \leq m} c_{\mathbf{n}} (\mathbf{x}_0 + a\mathbf{n})^{\mathbf{n}}$, $1 \leq |\mathbf{n}| \leq m$ has a unique solution for all $\{b_{\mathbf{n}}\}_{1 \leq |\mathbf{n}| \leq m}$. Since the number of equations is equal to the number of unknown, it is sufficient to show that the homogeneous system, $b_{\mathbf{n}} = 0$, $\forall \mathbf{n}$ has only the trivial solution.

The cases $d = 1$ or $m = 0$ are trivial. The rest is proved by induction: Without loss of generality, let $\mathbf{x}_0 = 0$ and denote $\mathbf{x} = (x_1, \dots, x_d)$ and similarly for \mathbf{n} . Consider $q(\mathbf{x}) = p(x_1 = 0) = \sum_{1 \leq |\mathbf{n}| \leq m, n_1 = 0} c_{\mathbf{n}} \mathbf{x}^{\mathbf{n}}$. It is a polynomial in \mathbb{R}^{d-1} of degree m . By induction, $q(\mathbf{x})$ is the zero polynomial, i.e., $c_{\mathbf{n}} = 0$ for all $\{1 \leq |\mathbf{n}| \leq m, n_1 = 0\}$. Hence, $p(\mathbf{x})$ can be divided by x_1 . Let $w(\mathbf{x}) = p(\mathbf{x})/x_1$, a polynomial in \mathbb{R}^d of degree $m - 1$ that vanishes on the grid $a\mathbf{n}$, $|\mathbf{n}| \leq m$, $n_1 = 0$. By the induction hypothesis, $w(\mathbf{x})$ is the zero polynomial. Hence, $c_{\mathbf{n}} = 0$ of all \mathbf{n} . \square

In light of this lemma, the set \mathcal{A} can be taken to be the discrete set of points $\mathcal{A} = \{\mathbf{x}_0 + a\mathbf{n} : a \neq 0, 1 \leq |\mathbf{n}| \leq m\}$ and $d\mu_{\mathbf{x}}$ the counting measure. In other word, we minimize

$$(3.14) \quad \min_{p \in \mathbb{P}_m(\mathbb{R}^d)} \sum_{1 \leq |\mathbf{n}| \leq m} |\nabla_{\mathbf{x}} p(\mathbf{x}_0 + a\mathbf{n}) \cdot \phi(\mathbf{x}_0 + a\mathbf{n})|^2,$$

where $\mathbb{P}_m(\mathbb{R}^d)$ is the set of all polynomials in \mathbb{R}^d , and any $\mathbf{x}_0 \in \mathbb{R}^d$, $a \neq 0$. As before, $\phi(\mathbf{x})$ denotes the function $d\mathbf{x}/dt$. a and \mathbf{x}_0 should be chosen such that \mathcal{A} is a constant distance away from the zero of f_{ϵ} .

This is a least square problem whose unknowns are the coefficients of the polynomial p . Theorem 3.4 and Lemma 3.5 suggest that this minimum should be of order one. Using, for instance, singular value decomposition, one can find a basis to the space of all polynomials. The eigenvectors, i.e., the coefficients, that correspond to eigenvalues of the order one define our slow variables.

For systems of the form (3.1), we know that we should search for $r + d_z$ polynomials that can serve as a change of coordinates. In general, we begin by searching for low degree polynomials that involve only a small number of coordinates and gradually look for more complicated ones. Once a slow variable is detected, the procedure is repeated with an additional constraint that the coefficients of the new polynomial are perpendicular (in the space of coefficients) to the ones already detected. The constraint can be implemented as a penalty to the minimization. Once a new slow variable ξ_i is identified, we check the rank of $\partial\xi/\partial\mathbf{x}$ at an appropriate set of points (also on a grid $x_0 + a\mathbf{n}$). If $\partial\xi/\partial\mathbf{x}$ does not have full rank, then the new slow variable is discarded. For high dimensional problems, it is sometimes possible to reduce the number of free parameters using symmetries of the particular equations of interest.

In addition, it is often useful to try identify linear combinations of the slow variables that involve the least possible number of non-zero entries. We employed a ‘‘clean up’’ algorithm in order to obtain a simple set of slow variables that can help in our understanding of the slow dynamics. This process is not essential for the HMM algorithm.

4. TIME AVERAGING

In this section we estimate the difference between angle and local time averages for integrable functions $\alpha(\mathbf{x})$. Our estimate applies to any system of ODEs that, following a change of variables $\Phi : \mathbf{x} \rightarrow (\xi, \phi, \gamma)$, takes the form (2.1) along with dissipative variables, i.e.,

$$(4.1) \quad \begin{cases} \dot{\xi} &= F(\xi, \phi, \gamma) \\ \dot{\phi} &= \epsilon^{-1}\Omega(\xi) + G(\xi, \phi, \gamma), \end{cases}$$

where $|\gamma| \leq C/(D + (t/\epsilon)^l)$, for some $l > 0$ and $f, g, \Omega \in \mathcal{C}^1$. For the rest of the paper we use C and D as generic positive constants whose values may change between expressions.

Let $\alpha(\mathbf{x}) = \alpha(\Phi^{-1}(\xi, \phi, \gamma)) \equiv \tilde{\alpha}(\xi, \phi, \gamma)$. For simplicity, we drop the tilde notation. We recall that

$$(4.2) \quad \langle \alpha \rangle(t) \equiv \langle \alpha(x(\cdot)) \rangle_\eta(t) = \int_{t-\eta}^{t+\eta} K_\eta(t - \tau) \alpha(x(\tau)) d\tau.$$

and

$$(4.3) \quad \bar{\alpha}(t) \equiv \bar{\alpha}(\xi, \phi)(t) = \int_{S^1} \alpha(\xi(t), \phi, \gamma = 0) d\phi.$$

Let $\alpha^{(q)}(t)$ denote the q -th time derivative and $\alpha^{[q]}(t)$ a q 'th anti-derivative of a \mathcal{C}^q function α . In particular, $\alpha^{(0)} = \alpha^{[0]} = \alpha$. The integration constant will be specified in each case.

Definition 4.1. Let $\mathbb{K}^{p,q}$ denote the space of \mathcal{C}^q functions which are even, supported on $[-1, 1]$, normalized and have p vanishing moments, i.e.

$$(4.4) \quad \int_{-1}^1 K(\tau)\tau^\nu d\tau = \begin{cases} 1 & \nu = 0 \\ 0 & \nu = 1 \dots p. \end{cases}$$

The following well-known Lemma will be useful.

Lemma 4.2. Let $\beta(t)$ denote an integrable function with period T_0 . For $K \in \mathbb{K}^{p,q}$, for some positive constant C ,

$$(4.5) \quad |K_\eta(\cdot) * \beta(\epsilon^{-1}\cdot) - \bar{\beta}| \leq C \|\beta\|_\infty \|K\|_{W^{1,q}} \left(\frac{\epsilon}{\eta}\right)^q,$$

where $\|\cdot\|_\infty$ denotes the sup norm in \mathcal{D}_0 ,

$$(4.6) \quad \|\beta\|_\infty = \sup_{\mathbf{x} \in \mathcal{D}_0} |\beta(\mathbf{x})|,$$

and

$$(4.7) \quad \|K\|_{W^{1,q}} = \int_{-1}^1 K^{(q)}(t) dt.$$

A proof can be found in [12].

The following theorem estimates the difference $|\langle \alpha \rangle_\eta - \bar{\alpha}|$ for any \mathcal{C}^1 function $\alpha(\mathbf{x}) = \alpha(\xi, \phi, \gamma)$. It is a generalization of a similar theorem found in [12]. Without loss of generality, we take $t = 0$.

Theorem 4.3. Let $\mathbf{x}(t)$ solve (1.1) in $[-\eta, \eta]$, $0 < \eta < 1$, with initial conditions $x(-\eta) = x_{-\eta} = (\xi_{-\eta}, \phi_{-\eta}, \gamma_{-\eta}) \in \mathcal{D}$. Also, suppose that $K \in \mathbb{K}^{p,q}$ and $\xi(t) \in \mathcal{C}^1$. Then

$$(4.8) \quad |\langle \alpha(\mathbf{x}) \rangle_\eta(0) - \bar{\alpha}(\xi_0)| \leq C \max \left\{ \eta, \eta^l, \left(\frac{\epsilon}{\eta}\right)^q \right\},$$

where $\xi_0 = \xi(0)$.

Proof. Denote $\xi(t) := \xi(x(t))$, $\gamma(t) := \gamma(x(t))$, and

$$\begin{aligned} \dot{\phi} &= \Omega(\xi(t)) + \epsilon g(\xi(t), \phi(t), \gamma(t)), & \theta(-\eta) &= \phi_{-\eta}; \\ \dot{\psi} &= \Omega(\xi_0), & \psi(-\eta) &= \phi_{-\eta}. \end{aligned}$$

For $|t| \leq \eta$, expanding $\alpha(\xi(t), \psi(t), \gamma(t))$ around $\alpha(\xi_0, \phi(t), 0)$ we obtain

$$\alpha(\xi(t), \phi(t), \gamma(t)) = \alpha(\xi_0, \psi(t), 0) + R_I(\xi, \psi, \gamma, t) + R_{II}(\xi, \psi, \gamma, t) + R_{III}(\xi, \psi, \gamma, t)$$

where $|R_I| \leq C|\xi(t) - \xi_0|$, $|R_{II}| \leq C|\psi(t) - \phi(t)|$, and $|R_{III}| \leq |\gamma(t)|$ for some positive constant C . For smooth $\xi(t)$, R_I is thus bounded by the Lipschitz constant

of $\xi(t)$; i.e. $|R_I| \leq C_1\eta$. By the hypothesis on $|\gamma|$, $|R_{III}| \leq \tilde{C}\eta^l$. Let $w(t) = \phi(t) - \psi(t)$, then we have the inequalities $\dot{w} \leq C_1|\xi(t) - \xi_0| + \epsilon C_3 \leq C_4(\eta + \epsilon)$. Using the initial condition $w(-\eta) = 0$, we have that for $|t| \leq \eta$,

$$|\phi(t) - \psi(t)| = |w(t)| \leq C_4(\eta^2 + \eta\epsilon).$$

Now average (4) by the kernel K_η :

$$\begin{aligned} & K_\eta * \alpha(\xi(\cdot), \phi(\cdot), \gamma(\cdot)) = \\ & K_\eta * \alpha(\xi_0, \psi(\cdot), 0) + K_\eta * (R_I(\xi, \psi, \gamma, \cdot) + R_{II}(\xi, \psi, \gamma, \cdot) + R_{III}(\xi, \psi, \gamma, \cdot)). \end{aligned}$$

Using Lemma 4.2, and the estimates obtained above, $|\bar{\alpha}(\mathbf{x}) - \langle \alpha(\mathbf{x}) \rangle| =$
 $|\bar{\alpha}(\mathbf{x}) - K_\eta * \alpha(\xi_0, \psi(\cdot), 0) + K_\eta * (R_I(\xi, \psi, \gamma, \cdot) + R_{II}(\xi, \psi, \gamma, \cdot) + R_{III}(\xi, \psi, \gamma, \cdot))|.$

□

The above theorem shows that time averages approximate averages using angular coordinates for any systems of ODEs that takes the form of (4.1). However, the estimate (4.8) is not tight. Improved bounds can be obtained for particular cases. For example, we consider a simple case with no dissipative modes and constant ϕ , i.e., $\Omega \equiv \omega$.

Theorem 4.4. *Let $K \in \mathbb{K}^{p,q}$ and $\alpha \in \mathcal{C}^{p+1}$. Suppose also that $\xi(t) = \xi(\mathbf{x}(t)) \in \mathcal{C}^{p+1}$. Then,*

$$(4.9) \quad |\bar{\alpha}(t) - \langle \alpha \rangle(t)| \leq C \left(\sum_{k=0}^p \|\alpha^{(k)}\|_\infty \right) \|K\|_{W^{1,q}} \max \left\{ \eta^{p+1}, \left(\frac{\epsilon}{\eta} \right)^q \right\}.$$

Proof. The proof is essentially the same as the proof of Theorem 4.3. The only difference is that $\xi(t)$ and $\alpha(\xi(t), \phi)$ are expanded to order $p+1$:

$$(4.10) \quad \begin{aligned} \xi(\tau) &= \xi_0 + \sum_{k=1}^p \tau^k \frac{\xi^{(k)}(0)}{k!} + \tau^{p+1} \frac{\xi^{(p+1)}(\tau^*)}{(p+1)!} \\ \alpha(\xi(\tau), \phi) &= \alpha(\xi_0, \phi) + \sum_{k=1}^p \tau^k \frac{\alpha_k(\tau=0, \phi)}{k!} + \tau^{p+1} \frac{\alpha_{p+1}(\tau=\tau^{**}, \phi)}{(p+1)!}, \end{aligned}$$

where $\xi_0 = \xi(0)$, $\tau^*, \tau^{**} \in [-\tau, \tau]$ and α_k are some functions of ϕ . A calculation similar to Lemma 4.2 shown that

$$(4.11) \quad |K_\eta * (\tau^k \beta_k)| \leq C \|\alpha^{(k)}\|_\infty \|K\|_{W^{1,q}} \left(\frac{\epsilon}{\eta} \right)^q.$$

In addition, the last term is trivially bounded by $C\eta^{p+1}$ for $|\tau| \leq \eta$. □

Suppose we identify $\Xi(\mathbf{x}) = \xi(\mathbf{x}) + \delta(\mathbf{x})$, as slow variables instead of the correct ones $\xi(\mathbf{x})$. For short hand we denote $\xi = \xi(\mathbf{x}(\cdot))$ and similarly for ϕ and δ . As a consequence, local time averages of a function $g(\xi, \phi)$ are replaced by

$$\langle g(\Xi, \phi) \rangle_\eta = K_\eta * [g(\xi + \delta, \phi)] = K_\eta * g(\xi, \phi) + K_\eta * [\delta g'(\xi, \phi)] + O(\delta^2).$$

The last term, which is the leading order term in the error introduced using the wrong slow variable, is of order δ/η .

5. HIGHER ORDER SCHEMES

In this Section we describe how the algorithm outlined in Section 2.2 can be generalized to other explicit Macro-solvers of order $s \geq 2$. For simplicity, we concentrate on the explicit mid-point method. It is possible to construct other methods in a similar fashion.

Let $z(t)$ denote the solution of

$$(5.1) \quad \dot{z} = f(z), \quad z(0) = z_0.$$

Recall the form of a single step of size H using the usual second order mid-point rule for integrating the equation for $z(t)$:

$$(5.2) \quad \begin{aligned} k_1 &= f(\mathbf{x}_n), \quad \mathbf{x}_{n+1/2} = \mathbf{x}_n + \frac{1}{2}Hk_1 \\ k_2 &= f(\mathbf{x}_{n+1/2}), \quad \mathbf{x}_{n+1} = \mathbf{x}_n + Hk_2. \end{aligned}$$

For the problem at hand, consider first a case in which (1.1) has no dissipative modes. The benefit of this situation is that it is possible to integrate the ODE (1.1) both forward and backward in time. Suppose that at time t_n , we have $\mathbf{x}(t_n) = \mathbf{x}_n$, which corresponds to (ξ_n, ϕ_n) . We would like to approximate the averaged force $\bar{F}(\xi_n)$ using local time averages. We therefore have

$$(5.3) \quad \bar{f}(\xi_n) \sim \langle \dot{\xi} \rangle_\eta = (K'_\eta * \xi(\cdot)),$$

where $\xi(t)$ denotes $\xi(\mathbf{x}(t))$ as obtained by the micro-solver for $t \in [t_n - \eta, t_n + \eta]$, integrated from t_n backwards to $t_n - \eta$ and from t_n forward to $t_n + \eta$ with initial condition $\mathbf{x}(t_n) = \mathbf{x}_n$. Note that the difference between (5.3) and the expression appearing in the algorithm described in Section 2.2 is that in (5.3) the average is evaluated at $t = t_n$ rather than at $t = t_n + \eta$. In order to integrate ξ using the mid-point rule we take

$$(5.4) \quad \begin{aligned} k_1^\eta &= (K'_\eta * \xi(\cdot)), \quad k_1^{\mathbf{x}} = \delta\mathbf{x} \\ \mathbf{x}_{n+1/2} &= \mathbf{x}_n + \frac{1}{2}Hk_1^{\mathbf{x}}, \end{aligned}$$

where $\delta\mathbf{x}$ is consistent with $\langle \dot{\xi} \rangle_\eta(t_n)$, i.e., $\delta\mathbf{x} \cdot \nabla \xi_i(\mathbf{x}_n) = \langle \dot{\xi}_i \rangle_\eta(t_n)$ for all $i = 1 \dots r$. The second half of the step at $t_{n+1/2} = t_n + H/2$ is,

$$(5.5) \quad \begin{aligned} k_2^\eta &= (K'_\eta * \tilde{\xi}(\cdot)), \quad k_2^{\mathbf{x}} = \delta\tilde{\mathbf{x}} \\ \mathbf{x}_{n+1} &= \mathbf{x}_0 + Hk_2^{\mathbf{x}}, \end{aligned}$$

where $\tilde{\xi}(t)$ denotes $\xi(\mathbf{x}(t))$ as obtained by the micro-solver for $t \in [t_{n+1/2} - \eta, t_{n+1/2} + \eta]$, integrated from $t_{n+1/2}$ backwards to $t_{n+1/2} - \eta$ and from $t_{n+1/2}$ forward to $t_{n+1/2} + \eta$ with initial condition $\mathbf{x}(t_{n+1/2}) = \mathbf{x}_{n+1/2}$. As before, $\delta\tilde{\mathbf{x}}$ is consistent with $\langle \dot{\xi} \rangle_\eta(t_{n+1/2})$. Using the method described above, the error in each

step of the Macro-solver is $O(H^3)$. Additional integration schemes, for instance, fourth order Runge-Kutta, are implemented in a similar way.

Generally, if (1.1) has one or more dissipative modes then it is not possible to integrate the ODE backwards and the method described above needs to be modified. Our implementation for the second order mid-point scheme is depicted in Figure 3. The first step is the same as before:

$$(5.6) \quad \begin{aligned} k_1^\eta &= (K'_\eta * \xi(\cdot)), \quad k_1^x = \delta \mathbf{x} \\ \mathbf{x}_{n+1/2} &= \mathbf{x}(t_n + \eta) + \frac{1}{2} H k_1^x, \end{aligned}$$

where $\xi(t)$ denotes $\xi(\mathbf{x}(t))$ as obtained by the micro-solver for $t \in [t_n, t_n + 2\eta]$, integrated from t_n forward to $t_n + 2\eta$ with initial conditions $\mathbf{x}(t_n) = \mathbf{x}_n$ and $\mathbf{x}(t_n + \eta)$ denotes the value of \mathbf{x} at the middle of the micro-simulation. For the second step we would like to evaluate $\langle \dot{\xi} \rangle_\eta(t_{n+1/2})$ at $\mathbf{x} = \mathbf{x}_{n+1/2}$. However, since the kernel K is symmetric, we need to start the micro-solver a time η earlier, at a point that is not known. This initial point can be approximated by taking a step $\delta \mathbf{x}_n$, starting at \mathbf{x}_n (rather than $\mathbf{x}_{n+1/2}$) and consistent with $\langle \dot{\xi} \rangle_\eta(t_{n+1/2} + \eta)$. The second step in the mid-point rule therefore has the form

$$(5.7) \quad \begin{aligned} k_2^\eta &= (K'_\eta * \tilde{\xi}(\cdot)), \quad k_2^x = \delta \tilde{\mathbf{x}} \\ \mathbf{x}_{n+1} &= \mathbf{x}_0 + H k_2^x, \end{aligned}$$

where $\tilde{\xi}(t)$ denotes the solution of the micro-solver for $t \in [t_{n+1/2}, t_{n+1/2} + 2\eta]$ with initial conditions $\mathbf{x}(t_{n+1/2}) = \mathbf{x}_n + H \delta \mathbf{x}_n / 2$. This method does not cancel the second order term (in H) exactly and leaves a residual error of the order of $H\eta$.

6. ACCURACY AND EFFICIENCY

In this Section we analyze the accuracy of the suggested method outlined in Section 2.2. Each step of the approximations performed in our algorithm introduces a numerical error. In order to optimize performance, the different sources of errors are balanced to a fixed prescribed accuracy Δ . We show how the different parameters: ϵ , η , h and H scale with Δ in order to have a global accuracy of order Δ . Note that the maximal possible accuracy is $\Delta = \epsilon$, since this is the error introduced by simulating the averaged equation rather than the original one. We also study the Δ dependence of the complexity of the algorithm.

We begin with estimating the error in our evaluation of the averaged force \bar{F} . There are several sources of errors:

- Global error in each micro-simulation. Using an m 'th order method with step size h the global error is $\eta h^m / \epsilon^{m+1}$.
- Quadrature error in $K'_\eta * \xi$: Using a quadrature formula of degree r the error is $\eta h^m / \epsilon^{(m+1)}$. However, due to the regularity of the kernel used $K \in \mathcal{C}^q$, the integrand is smooth and periodic. Hence, the coefficients of its Fourier

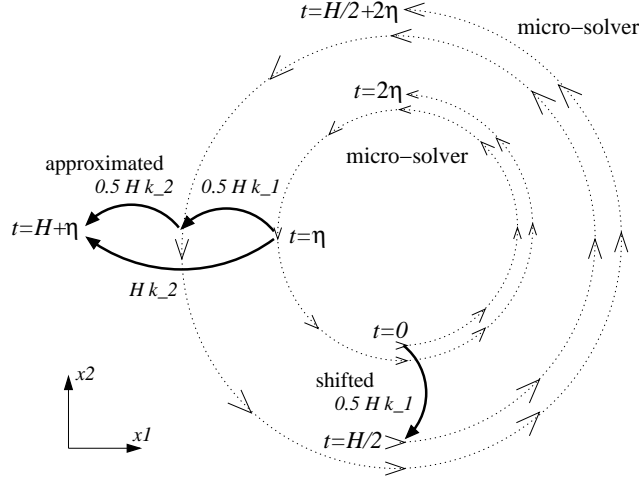


FIGURE 3. The cartoon depicts the implementation of the midpoint rule using (5.6) and (5.7) for a simple two dimensional system that oscillates with a frequency of the order ϵ^{-1} and expands on an $O(1)$ time scale. The dotted line depicts the trajectories obtained by the micro solver. The bold arcs depict the steps taken by the Macro-solver.

decomposition decay very fast. As a result, it is advantageous to use the trapezoidal rule, which is exact for $e^{2\pi i k x}$, $k \in \mathbb{N}$. This implies that the quadrature error is typically very small and can be neglected.

- Approximating \bar{F} by $\langle F \rangle_\eta$: Using a kernel $K \in \mathbb{K}^{p,q}$ the error is the larger between η^p and $(\epsilon/\eta)^q/\eta$. Note that we are losing one order of η compared to lemma 4.2 since \bar{F} is found through integration by parts (cf. Section 2.2). The above two bounds to the averaging error are equal if $\eta^{p+q+1} = \epsilon^q$, where, for large η , the term η^p dominates, while for small η the other. Since we would like to optimize our complexity, it is always preferable to work in the latter regime. Hence, we can take the averaging error to be $(\epsilon/\eta)^q/\eta$.
- Error due to inaccurate slow variables: Denoting the accuracy of the slow variables by $\delta < \eta$, the error in $\langle F \rangle_\eta$ is δ/η . For the rest of this discussion this error is also neglected.

For simplicity, we describe the error analysis for systems without dissipative modes. System with dissipative modes may involve additional errors, both from local time averages, which are evaluated in theorem 4.3, and from the use of high-order methods, as discussed in section 5. Balancing all terms yields the optimal scaling of the simulation parameters with Δ .

The global accuracy of integrating the original ODE (1.1) to time $T = O(1)$ using a Macro-solver of order s with step size H is, assuming errors are accumulative,

$$(6.1) \quad E \leq D \max \left\{ H^s, \frac{\eta h^m}{\epsilon^{m+1}}, \frac{\epsilon^q}{\eta^{q+1}} \right\},$$

For some $D > 0$. For short hand we drop the constant in all following expressions. Balancing the different sources of errors to a prescribed accuracy Δ yields

$$(6.2) \quad \begin{aligned} \eta &= \epsilon^{\frac{q}{q+1}} \Delta^{-\frac{1}{q+1}} \\ H &= \Delta^{\frac{1}{s}} \\ h &= \epsilon^{1+\frac{1}{m(q+1)}} \Delta^{\frac{s+1}{sm} + \frac{1}{m(q+1)}}. \end{aligned}$$

The complexity is With a smooth kernel we can consider the $q \rightarrow \infty$ limit.

$$(6.3) \quad C = \frac{\eta T}{h H} = \epsilon^{-\frac{m+1}{m(q+1)}} \Delta^{-\frac{1}{s} - \frac{s+1}{sm} - \frac{m+1}{m(q+1)}}.$$

In this case the complexity is reduced to

$$(6.4) \quad C(q \rightarrow \infty) = \Delta^{-\frac{1}{s} - \frac{s+1}{sm}}.$$

Figure 4 depicts the relative error of the HMM approximation compared to the analytical solution of the linear system discussed in Section 7.1 (with dissipative modes).. The kernel was constructed from polynomials to have exactly two continuous derivatives and a single vanishing moments, i.e., $q = 2$ and $p = 1$. Fourth order Runge-Kutta schemes were used for both the micro and the Macro solvers. The simulation parameters are chosen to balance all errors as discussed above.

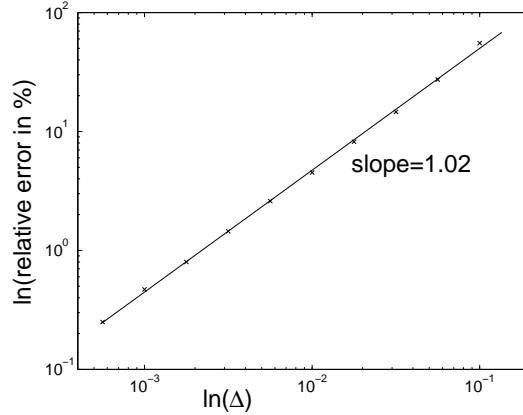


FIGURE 4. A log-log plot of the relative error of the HMM approximation to a linear ODE compared to the exact solution: $E = \max_{t_n \in [0, T]} 100 \times |\xi_{\text{HMM}}(t_n) - \xi_{\text{exact}}(t_n)| / |\xi_{\text{exact}}(t_n)|$, as a function of Δ .

From the parameter scaling (6.2) it is clear that the step size of the Macro-solver, H , does not depend on the stiffness ϵ , but only on the prescribed accuracy Δ . Our algorithm is therefore multiscale in the sense that it converges uniformly for all $\epsilon < \epsilon_0$ [10]. More precisely, denote the sample times of the Macro-solver by $t_0 = 0, \dots, t_N = T$ and the corresponding numerical approximations for \mathbf{x} by

$\mathbf{x}_0, \dots, \mathbf{x}_N$. The exact solution is denoted $\mathbf{x}(t)$. We have that, for any variable $\alpha(\mathbf{x})$ that is slow with respect to $\mathbf{x}(t)$

$$(6.5) \quad \lim_{H \rightarrow 0} \sup_{k=0 \dots N} \sup_{\epsilon < \epsilon_0} |\alpha(\mathbf{x}(t_k)) - \alpha(\mathbf{x}_k)| \rightarrow 0.$$

Note that the order of the limits is important.

7. EXAMPLES

In this Section we present a few example ODE systems that fall under the category of Equation (3.1) and compare the numerical solution of the HMM to other methods.

7.1. A simple linear example. We begin with a simple example of a linear system that contains both oscillatory and dissipative modes:

$$(7.1) \quad \begin{cases} \dot{x}_1 &= \epsilon^{-1}x_2 + x_1 + 2x_3 \\ \dot{x}_2 &= -\epsilon^{-1}x_1 + x_2 \\ \dot{x}_3 &= -\epsilon^{-1}x_3, \end{cases}$$

with initial conditions $(x_1, x_2, x_3)(0) = (1, 0, 1)$. The solution is

$$(7.2) \quad \begin{aligned} x_1(t) &= (1 + \epsilon)e^t \cos \epsilon^{-1}t - \epsilon e^{-\epsilon^{-1}t} \\ x_2(t) &= (1 + \epsilon)e^t \sin \epsilon^{-1}t \\ x_3(t) &= e^{-\epsilon^{-1}t}. \end{aligned}$$

The system admits a single slow variable $\xi = x_1^2 + x_2^2$. Outside a neighborhood of $\xi = 0$ the diffeomorphism Φ can be taken as

$$(7.3) \quad \Phi(x) = (x_1^2 + x_2^2, \arctan(x_2/x_1), x_3) \equiv (\xi, \phi, \gamma).$$

The Jacobian is $\det \nabla \Phi = 2$ in $\xi > 0$. Figure 5a depicts the HMM approximation of x_1 and x_2 compared to the analytical one (7.2). Our algorithm correctly approximates the slow variable $\xi = x_1^2 + x_2^2$, while the phase, ϕ , and the dissipative variable, x_3 , are lost. Simulation parameters are $\epsilon = 10^{-5}$, $\eta = 5.4\epsilon$, $h = \epsilon/15$, $T = 10$ and $H = 0.5$. The kernel used in averaging is:

$$(7.4) \quad K(t) = Z^{-1} \exp \left(-\frac{5}{4} \frac{1}{(t+1)(t-1)} \right),$$

where Z is a normalization constant. Hence, $K \in \mathbb{K}^{\infty,1}$, i.e., \mathcal{C}^∞ with a single vanishing moment. Both micro and Macro solvers implement a fourth order Runge-Kutta scheme. Figure 5b depicts the HMM approximation for several local averages and functionals. Note that the algorithm correctly approximates the value of the oscillating observables even though the Macro step size may be much larger than their period.

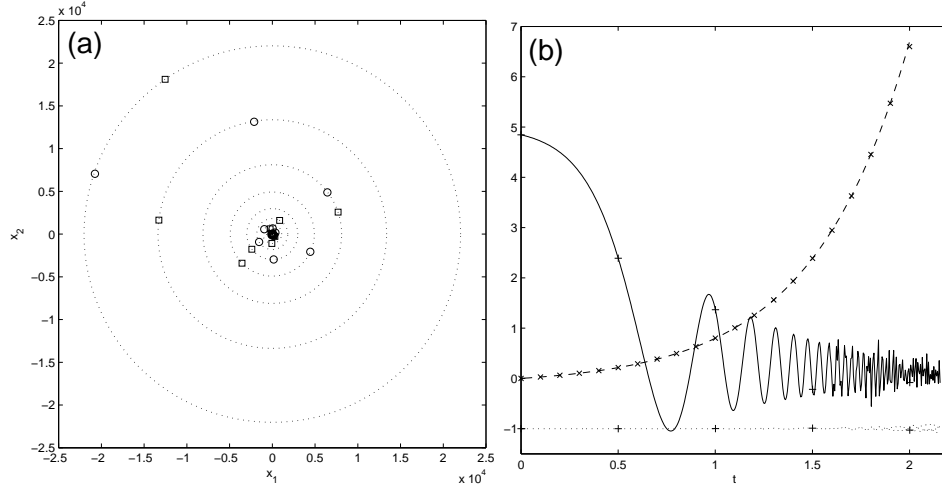


FIGURE 5. The HMM approximation to the solution of the linear example (7.1). (a) Trajectory of the x_1 and x_2 coordinates: The HMM time steps are denoted by circles while squares denote the exact solution at the same times. The dotted circles are guides for the eye. (b) Example local time averages and functionals. The dotted curve denotes the exact value for $\langle x_1^2 x_2 - 1 \rangle_\eta$, the solid curve of $\langle 5 \cos(x_1^2 x_2) \rangle_\eta$ and the dashed curve of $\int_0^t 0.5 x_1^2(\tau) d\tau$. The HMM values for the same observables are denoted by plus signs.

7.2. Stellar orbits in a galaxy. The following is a well studied system taken from the theory of stellar orbits in a galaxy [22, 23]

$$(7.5) \quad \begin{cases} r_1'' + a^2 r_1 &= \epsilon r_2^2 \\ r_2'' + b^2 r_2 &= 2\epsilon r_1 r_2, \end{cases}$$

where $r_1(s)$ stands for the radial displacement of the orbit of a star from a reference circular orbit and $r_2(s)$ stands for the deviation of the orbit from the galactic plane. The time-like variable $s \in [0, \epsilon^{-1}S]$ denotes the angle of the planets in a reference plane. Initial conditions are $r_1(0) = r_2(0) = 1$ and $r_1'(0) = r_2'(0) = 0$. Changing variable to $\mathbf{x} = (x_1, v_1, x_2, v_2)^T = (r_1, r_1'/a, r_2, r_2'/b)^T$ and $t = \epsilon s$, equation (7.5) becomes

$$(7.6) \quad \dot{\mathbf{x}} = \epsilon^{-1} A \mathbf{x} + f(\mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x}_0,$$

where

$$(7.7) \quad A = \begin{pmatrix} 0 & a & 0 & 0 \\ -a & 0 & 0 & 0 \\ 0 & 0 & 0 & b \\ 0 & 0 & -b & 0 \end{pmatrix}, \quad f(\mathbf{x}) = \begin{pmatrix} 0 \\ x_2^2/a \\ 0 \\ 2x_1 x_2/b \end{pmatrix}, \quad \mathbf{x}_0 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}.$$

To see how resonances occur, consider the following change of variables

$$(7.8) \quad \begin{aligned} \xi_1 &= x_1^2 + v_1^2, \quad \tan \phi_1 = v_1/x_1 \\ \xi_2 &= x_2^2 + v_2^2, \quad \tan \phi_2 = v_2/x_2. \end{aligned}$$

The coordinates ξ_1 and ξ_2 correspond to the amplitudes of the two oscillators, (x_1, y_1) and (x_2, y_2) , respectively. ϕ_1 and ϕ_2 correspond to the phase of each one of the oscillators. Under (7.8), the ODE (7.6) takes the form

$$(7.9) \quad \begin{cases} \dot{\xi}_1 &= \frac{1}{2}\sqrt{\xi_1}\xi_2 [\sin(\phi_1 + 2\phi_2) - \sin(\phi_1 - 2\phi_2)] \\ \dot{\xi}_2 &= \frac{1}{2}\sqrt{\xi_1}\xi_2 [2\cos\phi_1 - \cos(\phi_1 + 2\phi_2) - \cos(\phi_1 - 2\phi_2)] \\ \dot{\phi}_1 &= -\epsilon^{-1} + \sqrt{\xi_1}\sin\phi_1 \sin^2\phi_2/(2\xi_2) \\ \dot{\phi}_2 &= -\epsilon^{-1} + 2\sqrt{\xi_1}\sin\phi_1 \sin^2\phi_2. \end{cases}$$

It is clear that ξ_1 and ξ_2 are slow. Averaging $\dot{\xi}_1$ and $\dot{\xi}_2$ independently over ϕ_1 and ϕ_2 , we obtain the limiting effective equations for ζ_1 and ζ_2 :

$$(7.10) \quad \dot{\zeta}_1 = 0; \quad \dot{\zeta}_2 = 0.$$

However, for the special cases in which either the $\phi_1 + 2\phi_2$ or $\phi_1 - 2\phi_2$ are slow, the effective equations become more complicated. When $a = \pm 2b$, the leading term in the fast evolution of $\theta = \phi_1 \mp 2\phi_2$ is cancelled exactly and θ is a slow variable. The system is then said to be in resonance. In our algorithm, the requirement that θ is slow can be taken into account by adding a third slow variable. The algorithm described in Section 3.2, identifies the cubic polynomial

$$(7.11) \quad \theta = x_1x_2^2 + 2v_1x_2v_2 - x_1v_2^2.$$

The fast variable can be taken to be $\phi = \phi_1$ although our algorithm does not require identifying it. The choice of cubic polynomial (7.11) is not unique. However, any other slow variable can be expressed as a function of ξ_1, ξ_2 and θ . Figure 6 compares the HMM solution to a numerical integration of (7.5) using the fourth order Runge-Kutta method with a step size of $\epsilon/50$. HMM parameters are $\epsilon = 10^{-5}$, $h = \epsilon/50$, $H = 0.3$ and $\eta = 10.28\epsilon$. In graph (b) $\eta = 30.28\epsilon$ was used. Both micro and Macro solvers are fourth order Runge-Kutta. It is important to note that although θ is constant throughout the simulation, it is not possible to approximate ξ_1 and ξ_2 correctly without taking account of θ as well.

7.3. Kapitza's inverted pendulum. The following example, suggested by P.L. Kapitza [36] considers a pendulum with a rigid arm that is attached at one of its ends to a mechanical motor. The set up of the system is depicted in Figure 7a. 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

$$(7.12) \quad l\ddot{\theta} = [g + \epsilon^{-1}\sin(2\pi\epsilon^{-1}t)] \sin\theta,$$

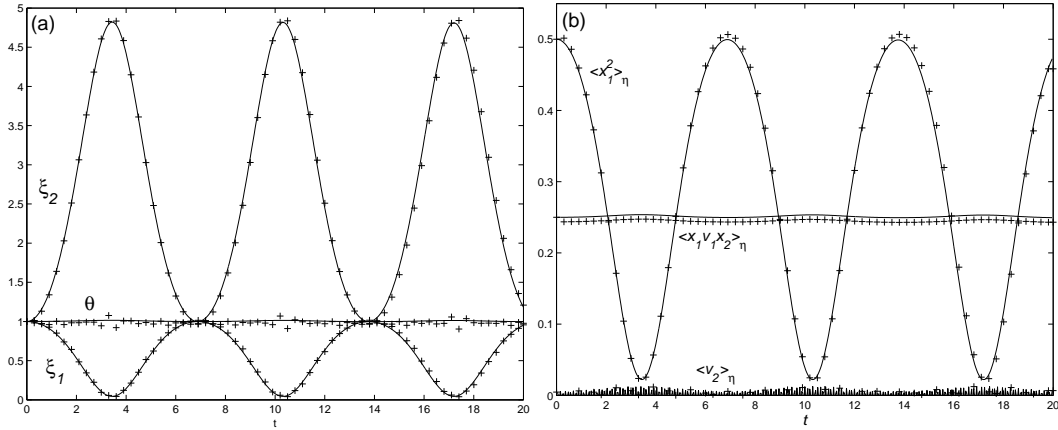


FIGURE 6. Numerical solution of the stellar equations (7.6). The solid line is the Runge-Kutta solution with step size of order ϵ while plus signs are the HMM approximation. Figure (a) shows the slow variables and figure (b) examples of local time averages.

where l is the length of the pendulum's arm and g is the gravitational constant. The averaged dynamics of the pendulum was studied analytically in [30]. Sharp et. al. [35] used the HMM framework to numerically integrate (7.12). Their approach, however, is different from the one described in this paper.

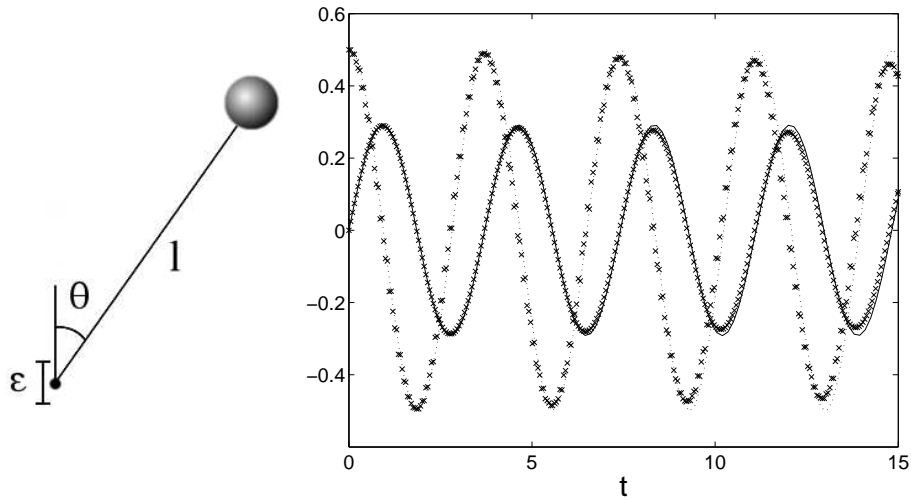


FIGURE 7. (a) Kapitza's pendulum has a rigid arm which is attached to a motor that is vibrating fast. The centrifugal force pulls the arm upwards. (b) Comparison of the HMM approximation for the solution of the equations describing the dynamics of Kapitza's pendulum to the Verlet method with step size of order ϵ . The solid curve depicts θ while the dotted one depicts $\psi = \dot{\theta} + \cos \theta \cos(2\pi\epsilon^{-1}t)/(2\pi l)$.

In order to put (7.12) in a form for which our method for finding polynomial slow variables can be applied, let $x_1 = \cos(2\pi\epsilon^{-1}t)$, $x_2 = \sin(2\pi\epsilon^{-1}t)$, $y_1 = \sin\theta$, $y_2 = \cos\theta$ and $z = \theta$. Equation (7.12) becomes

$$(7.13) \quad \begin{aligned} \dot{x}_1 &= 2\pi\epsilon^{-1}x_2, \quad \dot{x}_2 = -2\pi\epsilon^{-1}x_1 \\ \dot{y}_1 &= y_2z, \quad \dot{y}_2 = -y_1z \\ \dot{z} &= \epsilon^{-1}l^{-1}x_2y_1 + gl^{-1}y_1 \end{aligned}$$

which has the form (3.1). The slow variables admitted by (7.13) are

$$(7.14) \quad \begin{aligned} \xi_1 &= y_1, \quad \xi_2 = y_2 \\ \xi_3 &= x_1^2 + x_2^2, \quad \xi_4 = z - x_1y_1/(2\pi l) \end{aligned}$$

Going back to the original coordinates system, the slow variables are θ and $\psi = \dot{\theta} + \cos\theta \cos(2\pi\epsilon^{-1}t)/(2\pi l)$. Figure 7b depicts the HMM approximation for θ and ψ compared to numerical integration of (7.12) using the Verlet method with a step size of order ϵ . Simulation parameters are $\epsilon = 10^{-5}$, $h = \epsilon/40$, and $H = 0.25$ and $\eta = 25.4\epsilon$.

7.4. The Fermi-Pasta-Ulam model. The Fermi-Pasta-Ulam model [13] 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. [18] 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 following Hamiltonian

$$(7.15) \quad H(p, q) = \frac{1}{2} \sum_{i=1}^{2k} p_i^2 + \frac{1}{4}\epsilon^{-2} \sum_{i=1}^k (q_{2i} - q_{2i-1})^2 + \sum_{i=0}^k (q_{2i+1} - q_{2i})^4.$$

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

$$(7.16) \quad x_i = \epsilon^{-1}(q_{2i-1} - q_{2i})/\sqrt{2}, \quad v_i = (p_{2i-1} - p_{2i})/\sqrt{2},$$

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

$$(7.17) \quad y_i = (q_{2i-1} + q_{2i})/\sqrt{2}, \quad u_i = (p_{2i-1} + p_{2i})/\sqrt{2},$$

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

$$(7.18) \quad \begin{cases} \dot{y}_i &= u_i \\ \dot{x}_i &= \epsilon^{-1}v_i \\ \dot{u}_i &= -(y_i - \epsilon x_i - y_{i-1} - \epsilon x_{i-1})^3 + (y_{i+1} - \epsilon x_{i+1} - y_i - \epsilon x_i)^3 \\ \dot{v}_i &= -\epsilon^{-1}x_i + (y_i - \epsilon x_i - y_{i-1} - \epsilon x_{i-1})^3 + (y_{i+1} - \epsilon x_{i+1} - y_i - \epsilon x_i)^3. \end{cases}$$

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, $\phi_k = x_1 x_i + v_1 v_i$, $i = 1 \dots k - 1$. Any other function $\alpha(x, y, v, u)$ which is slow under the dynamics of (7.18) can be written as a function of the $4k - 1$ variables described above.

On the $O(1)$ time scale the system can be evolved using the algorithm described in section 2. We find that 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 complicated, non-harmonic way. Figure 8a depicts our results for systems with three stiff springs, $k = 3$, and with ten springs, $k = 10$, in Figure 9a. 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 [13, 18]. Unfortunately, the averaging theorem can not be generally extended to the ϵ^{-1} time scale due to the exponential dependence on time that appears in (2.4). However, in this case, due to the oscillatory nature of the soft degrees of freedom, the dynamics undergoes additional averaging. To this end we construct the Macro-solver to be almost time-reversible, i.e., the integrator is reversible for $\epsilon = 0$. A single Macro step is implemented the following way. First, the soft variables y_i and u_i are advanced by half a time step, $H/2$. Then, the stiff variables x_i and v_i are advanced by a full time step H while keeping y_i and u_i fixed. Finally, y_i and u_i are advance again by half a step. Although we did not prove convergence of the scheme in this set up, the numerical results depicted in Figure 8b for $k = 3$ and in Figure 9b, agree with integration of the model using the Verlet method with a step size of order ϵ . Note that both methods do not approximate the soft degrees of freedom correctly on the longer, ϵ^{-1} , time scale. Simulation parameters for $k = 3$ are $\epsilon = 10^{-4}$, $h = \epsilon/15$, and $H = 0.02$ and $\eta = 20.4\epsilon$. For $k = 10$ we used $\epsilon = 10^{-4}$, $h = \epsilon/35$, and $H = 0.02$ and $\eta = 35.4\epsilon$.

8. CONCLUSION

We have presented a numerical class of algorithms that compute the effective slow behavior of highly oscillatory solutions to ordinary differential equations. A key step is to first numerically detect a set of slow variables, ξ , that are effectively closed, i.e., their dynamics is closed in the limit of $\epsilon \rightarrow 0$. The main idea is then to integrate an averaged equation for these slow variables. The time stepping operates on two scales. First, a micro-solver evaluates the time derivatives of the identified slow variables by solving the original system in a short time segment. The micro-solver is an explicit integrator with step size of the order of ϵ . Then, a Macro-solver evolves the original variables, \mathbf{x} , by taking a large step that is consistent with the time derivative of the slow variables, obtained by the micro-solver. The Macro-solver, which is effectively integrating the averaged equation, can take steps of size that is almost independent of ϵ . Hence, in order to achieve an a priori fixed accuracy

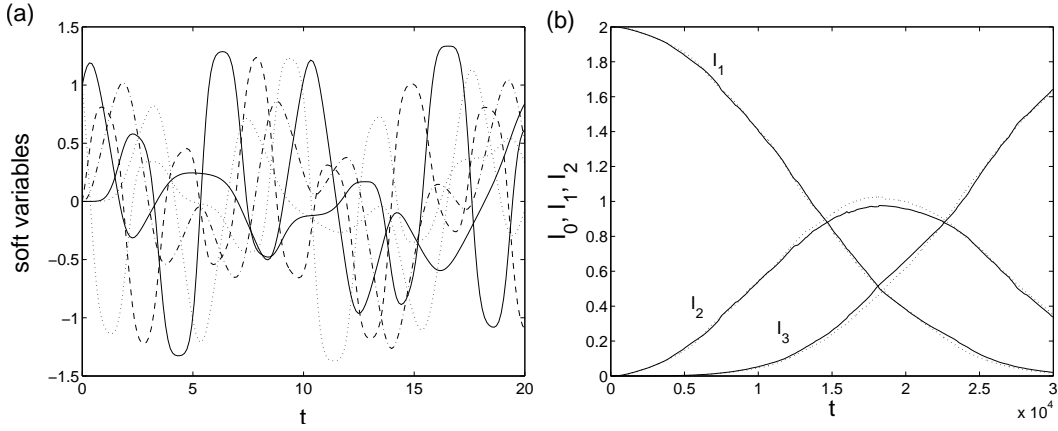


FIGURE 8. Comparison of the HMM approximation for the solution of the Fermi-Pasta-Ulam equations of motion (7.18) with 3 stiff springs, $k = 3$, to the one obtained using the Verlet method with step size of the order of ϵ . (a) soft variables on a $O(1)$ time scale and (b) I_1, I_2 and I_3 on a $O(\epsilon^{-1})$ 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 .

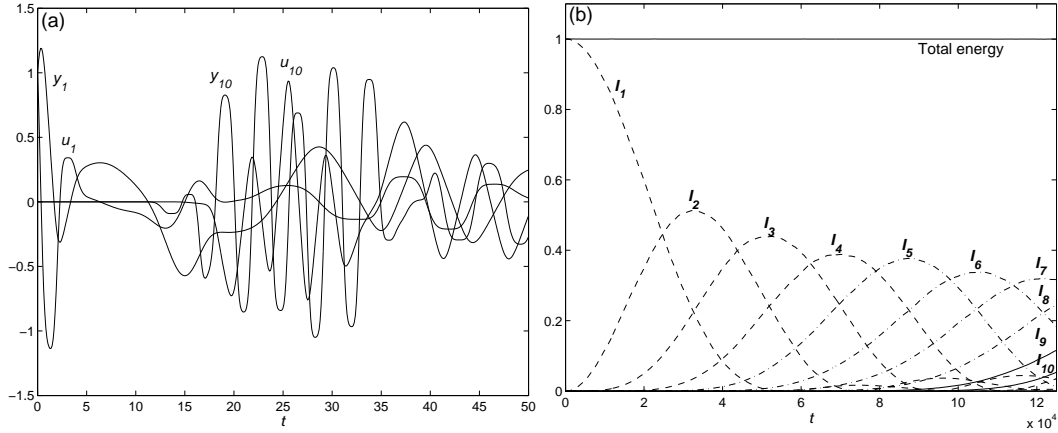


FIGURE 9. The HMM approximation for the solution of the Fermi-Pasta-Ulam equations of motion (7.18) with 10 stiff springs, $k = 10$. (a) y_1, u_1, y_{10} and u_{10} on a $O(1)$ time scale and (b) $I_1 \dots I_{10}$ on a $O(\epsilon^{-1})$ scale. The Verlet method takes too long to integrate.

Δ , the overall efficiency of the algorithm is independent of ϵ asymptotically. This is achieved by simulating a stiffer version of the original ODE.

An important observation is that the slow behavior of a system can be a result of internal mutual cancellation of the oscillations. Such cancellations are called resonance. The slow variables serve as a set of constraints to the fast dynamics of the system. Keeping track of the evolution of these constraints maintains the correct

phase difference between different stiff oscillators. As a result, the resonances are resolved and fully accounted for.

We applied this approach successfully to several systems, including the Fermi-Past-Ulam problem. This paper considers predominantly problems with fast dynamics in the form of harmonic oscillators. Some of the simpler examples considered in Section 7 can be integrated by other numerical methods. For example, trigonometric or exponential integrators [15, 18, 20], or envelope tracking methods [32] may also be appropriate. In fact, we suspect that most of these schemes will outperform HMM when applied to reversible linear systems that are not in resonance. However, the advantage of HMM in general, and the algorithm proposed here in particular, is its applicability to a wider class of ODE systems, including the difficult case of resonance. Several directions await for further study. An extension of our approach to the cases of variable coefficients and to fast anharmonic oscillators will be presented in a future publication [2]. In addition, already in Fermi-Past-Ulam problem, we see a need to design a three-scale method so that the ϵ^{-1} time scale can be computed consistently and efficiently.

Finally, as discussed in the introduction, for suitable systems our method correctly approximates all variables and functionals that are slow with respect to the system dynamics. It may be argued that this criterion is too strict. It is often the case that we are only interested in a smaller set of observables. For instance, temperature, heat capacity, or other statistical averages of a large system. In this case, it is useful to understand which slow variables are essential to a consistent approximation of particular observables. Another generalization of the methods proposed in this paper is to stochastic ordinary differential equations, compare [11, 37].

ACKNOWLEDGMENTS

Support from NSF through Grant DMS-0714612 is gratefully acknowledged. The authors thank Heinz-Otto Kreiss for his stimulating comments and insightful suggestions. The authors also thank Richard Sharp and Eric Vanden-Eijnden. Tsai's research is supported by an Alfred P. Sloan Fellowship. Tsai and Engquist thank the Isaac Newton Institute for mathematical sciences for hosting parts of this research. Tsai also thanks the Tröskö-Storö Institute of Mathematics.

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 weakly coupled oscillators. Submitted to *Multi. Model. Simul.*
- [3] N. N. Bogoliubov and Yu. A. Mitropolski. *Asymptotic Methods in the Theory of Nonlinear Oscillations*. Gordon and Breach, New York, 1961.
- [4] G. L. Browning and H.-O. Kreiss. Multiscale bounded derivative initialization for an arbitrary domain. *J. Atmospheric Sci.*, 59(10):1680–1696, 2002.

- [5] G. Dahlquist. Convergence and stability in the numerical integration of ordinary differential equations. *Mathematica Scandinavica*, 4:33–53, 1956.
- [6] G. Dahlquist. Stability and error bounds in the numerical integration of ordinary differential equations. *Kungl. Tekn. Högsk. Handl. Stockholm. No.*, 130:87, 1959.
- [7] G. Dahlquist. A special stability problem for linear multistep methods. *Nordisk Tidskr. Informations-Behandling*, 3:27–43, 1963.
- [8] G. Dahlquist. Error analysis for a class of methods for stiff non-linear initial value problems. In *Numerical analysis (Proc. 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975)*, pages 60–72. Lecture Notes in Math., Vol. 506. Springer, Berlin, 1976.
- [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, D. Liu, and E. Vanden-Eijnden. Analysis of multiscale methods for stochastic differential equations. *Commun. on Pure and Applied Math.*, 58:1544–1585, 2005.
- [12] B. Engquist and Y.-H. Tsai. Heterogeneous multiscale methods for stiff ordinary differential equations. *Math. Comp.*, 74(252):1707–1742, 2005.
- [13] 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.
- [14] B. García-Archilla, J. M. Sanz-Serna, and R. D. Skeel. Long-time-step methods for oscillatory differential equations. *SIAM J. Sci. Comput.*, 20(3):930–963, 1999.
- [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] M. Hochbruck, C. Lubich, and H. Selhofer. Exponential integrators for large systems of differential equations. *SIAM J. Sci. Comp.*, 19:1552–1574, 1998.
- [21] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. In *Acta numerica, 2000*, volume 9, pages 215–365. Cambridge Univ. Press, 2000.
- [22] J. Kevorkian and J. D. Cole. *Perturbation Methods in Applied Mathematics*, volume 34 of *Applied Mathematical Sciences*. Springer-Verlag, New York, Heidelberg, Berlin, 1980.
- [23] J. Kevorkian and J. D. Cole. *Multiple Scale and Singular Perturbation Methods*, volume 114 of *Applied Mathematical Sciences*. Springer-Verlag, New York, Berlin, Heidelberg, 1996.
- [24] H.-O. Kreiss. Difference methods for stiff ordinary differential equations. *SIAM J. Numer. Anal.*, 15(1):21–58, 1978.
- [25] H.-O. Kreiss. Problems with different time scales. In *Acta numerica, 1992*, pages 101–139. Cambridge Univ. Press, 1992.
- [26] H.-O. Kreiss and J. Lorenz. Manifolds of slow solutions for highly oscillatory problems. *Indiana Univ. Math. J.*, 42(4):1169–1191, 1993.
- [27] J Laskar. Large scale chaos in the solar system. *Astron. Astrophys*, 287, 1994.

- [28] V. I. Lebedev and S. A. Finogenov. The use of ordered Čebyšev parameters in iteration methods. *Ž. Vyčisl. Mat. i Mat. Fiz.*, 16(4):895–907, 1084, 1976.
- [29] B. Leimkuhler and S. Reich. *Simulating Hamiltonian dynamics*, volume 14 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, 2004.
- [30] M. Levi. Geometry and physics of averaging with applications. *Physica D*, 132:150–164, 1999.
- [31] R.E. O’Malley. On nonlinear singularly perturbed initial value problems. *SIAM Review*, 30(2):193–212, 1988.
- [32] R.L. Petzold, O.J. Laurent, and Y. Jeng. Numerical solution of highly oscillatory ordinary differential equations. *Acta Numerica*, 6:437–483, 1997.
- [33] 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.
- [34] R. E. Scheid. The accurate numerical solution of highly oscillatory ordinary differential equations. *Mathematics of Computation*, 41(164):487–509, 1983.
- [35] 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.
- [36] D. Ter Haar, editor. *Collected Papers of P.L. Kapitza*, volume II. Pergamon Press, 1965.
- [37] E Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Comm. Math. Sci.*, 1:385–391, 2003.

DEPARTMENT OF MATHEMATICS, THE UNIVERSITY OF TEXAS AT AUSTIN, AUSTIN, TX, 78712, USA

E-mail address: ariel@math.utexas.edu

E-mail address: engquist@math.utexas.edu

E-mail address: ytsai@math.utexas.edu