MULTISCALE MODEL. SIMUL.
Vol. 10, No. 4, pp. 1180–1203

A MULTISCALE TECHNIQUE FOR FINDING SLOW MANIFOLDS OF STIFF MECHANICAL SYSTEMS*

G. ARIEL†, J. M. SANZ-SERNA‡, AND R. TSAI§

Abstract. In the limit of infinite stiffness, the differential equations of motion of stiff mechanical systems become differential algebraic equations whose solutions stay in a constraint submanifold $\widehat{\mathcal{P}}$ of the phase space. Even though solutions of the stiff differential equations are typically oscillatory with large frequency, there exists a slow manifold $\widetilde{\mathcal{P}}$ consisting of nonoscillatory solutions; $\widetilde{\mathcal{P}}$ has the same dimension as $\widehat{\mathcal{P}}$ and converges to it as the stiffness approaches infinity. We introduce an iterative projection algorithm, IPA, that projects points in the phase space of a stiff mechanical system onto the associated slow manifold $\widetilde{\mathcal{P}}$. The algorithm is based on ideas such as micro-integration and filtering coming from the field of multiscale simulation and is applicable to initializing integration algorithms for both stiff ODEs and DAEs, including the initialization of Lagrange multipliers. We also illustrate in a model situation how the algorithm may be combined with numerical integrators for both the stiff system and the limit constrained system. These combinations may speed up the solution of stiff problems and also be used to integrate DAEs with explicit algorithms.

Key words. slow manifolds, filtering, holonomic constraints

AMS subject classifications. 65L, 65P, 37M, 65Z

DOI. 10.1137/120861461

1. Introduction. In this paper we propose an iterative projection algorithm, IPA, for computing slow points of stiff mechanical systems, i.e., points that, when taken as initial data, do not give rise to rapidly oscillatory solutions. We provide examples of several possible uses of that algorithm to initialize simulations of both stiff ordinary differential equations (ODEs) and differential algebraic equations (DAEs) (including the initialization of Lagrange multipliers). We illustrate in a model situation how the algorithm may be combined with numerical integrators for both the stiff system and the limit constrained system. These combinations may speed up the solution of stiff problems and also be applied to integrate DAEs without solving algebraic equations.

We are concerned with the stiff ODEs

$$\ddot{q} = F(q;t) - \omega^2 g'(q;t)^T g(q;t), \qquad \omega \gg 1, \quad q \in \mathbb{R}^d, \quad g \in \mathbb{R}^k, \quad k \le d, \tag{1}$$

and with the associated (index 3) DAEs [6], [15]

$$\ddot{q} = F(q;t) - g'(q;t)^T \lambda, \qquad g(q;t) = 0. \tag{2}$$

A simple model example is given by the DAEs that govern the motion of a planar pendulum and the ODEs that describe the mechanical system obtained by replacing

*Received by the editors January 5, 2012; accepted for publication (in revised form) July 6, 2012; published electronically October 11, 2012.
http://www.siam.org/journals/mms/10-4/86146.html
†Department of Mathematics, Bar-Ilan University, Ramat Gan, 52900, Israel (arielg@math.biu.ac.il).
‡Departamento de Matemática Aplicada, Facultad de Ciencias, Universidad de Valladolid, Valladolid, Spain (sanzsern@mac.uva.es). The work of this author was supported by grant MTM2010-18246-C03-01 (Ministerio de Ciencia e Innovación).
§Department of Mathematics and Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78712 (ytsai@math.utexas.edu). The work of this author was partially supported by NSF grants DMS-0714612 and DMS-0914465.

the pendulum rod by a stiff spring (see section 2.3). Molecular dynamics simulations may be performed including very stiff forces associated with chemical bonds between atoms or alternatively with constrained models where interatomic distances are assumed to remain constant [22, section 13.5]. Similar considerations apply to other cases, including circuit and multibody simulations, where small, fast oscillations may be removed from the model by the introduction of constraints.

In (1) and (2), F and g are smooth functions and, furthermore, it is assumed that for each q and t the $k\times d$ Jacobian matrix $g'(q;t)$ of g with respect to q has independent rows. In this way $g(q;t)=0$ defines a $d-k$ time-dependent submanifold $\widehat{\mathcal{C}}=\widehat{\mathcal{C}}(t)$ of $\mathbb{R}^d$ and (2) describes a mechanical system with forces F whose configuration is constrained to be in $\widehat{\mathcal{C}}(t)$ at each time t. The equations in (2) include redundant, nonindependent configuration coordinates q and are sometimes called the Lagrangian equations of motion of the first kind of the mechanical system. The term $-g'(q;t)^T\lambda$ represents the forces exerted by the constraints, and $\lambda\in\mathbb{R}^k$ is the time-dependent vector of Lagrange multipliers. Throughout this paper, we denote the velocities by $p=\dot{q}$. If $(\widehat{q}(t),\widehat{p}(t))$ is a solution of (2), from $g(\widehat{q}(t);t)=0$, we may write

$$\dot{g}(\widehat{q}(t),\widehat{p}(t);t)=g'(\widehat{q}(t);t)\,\widehat{p}(t)+\frac{\partial g}{\partial t}(\widehat{q}(t);t)=0,$$

so that, at each time t, the point $(\widehat{q}(t),\widehat{p}(t))$ is on the time-dependent $2(d-k)$-dimensional submanifold $\widehat{\mathcal{P}}=\widehat{\mathcal{P}}(t)$ of $\mathbb{R}^d\times\mathbb{R}^d$ defined by the equations

$$g(q;t)=0,\qquad \dot{g}(q,p;t)=0.$$

We shall say that $\widehat{\mathcal{P}}$ is the *constraint* submanifold of the DAEs (2).[1] In the important case where g does not depend on t, the equations of $\widehat{\mathcal{P}}$ become $g(q)=0$, $g'(q)p=0$ and $\widehat{\mathcal{P}}$ is the tangent bundle of the (time-independent) submanifold $\widehat{\mathcal{C}}$ of $\mathbb{R}^d$. Of course the initial value given by (2) in tandem with $q(t_0)=q_0$, $p(t_0)=p_0$ only has a solution if the given initial point (q_0,p_0) is on $\widehat{\mathcal{P}}(t_0)$.

The motions of (1) are not constrained; the corresponding initial value problem is solvable for arbitrary (q_0,p_0) in the *phase space* $\mathbb{R}^d\times\mathbb{R}^d$. In (1), $-\omega^2 g'(q;t)^T g(q;t)$ represents large $\mathcal{O}(\omega^2)$ forces that are derived from the potential

$$V(q;t)=\frac{\omega^2}{2}g(q;t)^T g(q;t). \tag{3}$$

Those forces lie in the column space of $g'(q;t)^T$; i.e., they are orthogonal, in the *configuration space* $\mathbb{R}^d$, to $\widehat{\mathcal{C}}$. Their presence implies that, *typically*, solutions $q(t)$ of (1) oscillate with large $\mathcal{O}(\omega)$ frequencies in directions orthogonal to $\widehat{\mathcal{C}}$. However, if $g(q;t)$ is small, i.e., if q is not far away from $\widehat{\mathcal{C}}$, the force $-\omega^2 g'(q;t)^T g(q;t)$ is of moderate size. In fact it turns out (see section 2) that there is a time-dependent $2(d-k)$-dimensional submanifold $\widetilde{\mathcal{P}}=\widetilde{\mathcal{P}}(\omega;t)$ of $\mathbb{R}^d\times\mathbb{R}^d$, $\mathcal{O}(\omega^{-2})$ close to the constraint manifold $\widehat{\mathcal{P}}$ such that for (q_0,p_0) in $\widetilde{\mathcal{P}}(\omega;t_0)$ the solution of the initial value problem given by (1) and $q(t_0)=q_0$, $p(t_0)=p_0$ has a *slow* solution, i.e., a solution that does not include rapidly oscillatory components. We say that $\widetilde{\mathcal{P}}$ is the *slow manifold* of (1).[2] This manifold is left invariant by (1): for solutions of (1), $(q(t_0),p(t_0))\in\widetilde{\mathcal{P}}(\omega,t_0)$ at a given t_0 implies $(q(t),p(t))\in\widetilde{\mathcal{P}}(\omega,t)$ for each t.

[1] Throughout this paper the notation $\widehat{\ }$ will refer to entities related to the constraint manifold.
[2] Entities associated with the slow manifold will be identified by the notation $\widetilde{\ }$.

It is well known [19], [14] that as ω increases, the slow solutions of the stiff system (1) converge to solutions of the DAEs (2). Moreover if, as $\omega \to \infty$, we consider fixed initial data (q_0, p_0) bounded away from the slow manifold, then the corresponding solutions $(q(t), p(t))$ exhibit oscillations of increasing energy and therefore become unphysical. These are the arguments that allow the introduction of the constrained Lagrangian model (2) as the limit of unconstrained systems (2) of very large stiffness; see, e.g., [4, section 17]. Slow manifolds are important in the analysis and numerical treatment of dynamical systems; a list of nearly one hundred references is given in the survey [20].

The present article is devoted to introducing and analyzing an algorithm that, for any given time t_0 and point (q_0, p_0) in the phase space $\mathbb{R}^d \times \mathbb{R}^d$, finds the projection $(\widetilde{q}_0, \widetilde{p}_0)$ of (q_0, p_0) onto the slow manifold $\widetilde{\mathcal{P}}(\omega; t_0)$. The algorithm is based on the fact that the solution of (1) with initial condition (q_0, p_0) oscillates around the slow manifold. We integrate with an $\mathcal{O}(\omega^{-1})$ step size h the corresponding stiff initial value problem in a small time window of $\mathcal{O}(\omega^{-1})$ length to obtain a so-called microsolution; the oscillations in the microsolution are then filtered away by convolution with an averaging kernel, and this results in a point (q_1, p_1) closer to the slow manifold than the initial (q_0, p_0). The procedure is then *iterated* a few times to obtain a point sufficiently close to the slow manifold. An important feature of the algorithm is that, since the step size and and the length of the micro-integration window are both $\mathcal{O}(\omega^{-1})$, the required work does not grow with the stiffness parameter ω.

The ideas of micro-integration and filtering used here are borrowed from the field of multiscale methods for the integration of differential equations; see [12] and also [9], [10], [11], [13], [21], [1], [2], [3], [24], [7], [8]. In fact the iterative projection algorithm may be used to build *explicit* integrators to compute both slow solutions of (1) and solutions of (2). This is discussed in section 6.

The paper is divided into seven sections. Section 2 presents background material on slow solutions and slow manifolds, together with some examples that illustrate the relevant theory. The iterative algorithm IPA is introduced in section 3. That section also reviews the process of filtering, with particular attention to the role of two parameters of the kernel that are crucial in the analysis: the number of vanishing moments and the number of continuous derivatives. The algorithm is analyzed in section 4. Sections 5 and 6 are devoted to numerical experiments, and section 7 contains our conclusions.

All of our results are trivially extended to cases where the left-hand sides of (1)–(2) are changed into $M\ddot{q}$, where M is a symmetric, positive definite mass matrix. The format $-\omega^2 g'(q;t)^T g(q;t)$ may be replaced, as in [19] or [14], by a more general $-\nabla V_\omega(q;t)$ with V_ω a potential of size $\mathcal{O}(\omega^2)$, but the required treatment is somewhat more complicated than that presented here.

2. Slow solutions. This section contains some background material that is needed to present the iterative projection algorithm.

2.1. The slow manifold. We begin by defining the concept of *slow function* (see [16, Definition 2.1], [17, Definition 1.1]). Let $\omega > 0$ parameterize a family of vector-valued functions v of the variable t, $0 \le t \le T$. Here the dependence of v on ω is not incorporated into the notation for simplicity. We say that v is slow of order $\ell = 0, 1, \ldots$ if v has ℓ continuous derivatives and there exist $\omega_0 > 0$ such that

$$\sup_{\omega \ge \omega_0} \sup_{0 \le t \le T} \left| \frac{\partial^j v}{\partial t^j} \right| < \infty, \qquad j = 0, \ldots, \ell.$$

We say that v is *slow* if it is slow to any order ℓ.

Let us review some relevant results from the literature. If $(\widehat{q}_0, \widehat{p}_0)$ is a point on the constraint manifold $\widehat{\mathcal{P}}(t_0)$, there exists (see [19, Theorem 2.2]) a point $(\widetilde{q}_0^\omega, \widetilde{p}_0^\omega)$, $\mathcal{O}(\omega^{-2})$ away from $(\widehat{q}_0, \widehat{p}_0)$, such that the solution of the initial value problem given by (1) and $\widetilde{q}(t_0) = \widetilde{q}_0^\omega$, $\widetilde{p}(t_0) = \widetilde{p}_0^\omega$ exists in an ω-independent time interval $0 \le t \le T$ and is slow. The collection of all pairs $(\widetilde{q}_0^\omega, \widetilde{p}_0^\omega)$ forms, for each t_0 and ω, a $2(d-k)$-submanifold $\widetilde{\mathcal{P}}(\omega; t_0)$ of $\mathbb{R}^d \times \mathbb{R}^d$. The time-dependent manifold $\widetilde{\mathcal{P}}(\omega; t)$ is invariant with respect to (1).

The dynamics of the solutions of (1) when the initial data (q_0, p_0) is in the neighborhood of $\widetilde{\mathcal{P}}(\omega; t_0)$ (but not on it) is studied carefully in [14, section XIV.3] (see also section 4 below). For the purposes of this paper it is crucial to note that those solutions oscillate around the slow manifold with frequencies proportional to ω. Furthermore, these fast oscillations create in general a drift in the tangential direction; if κ measures the distance between the initial condition and the slow manifold, then the magnitude of the drift is $\mathcal{O}(\kappa\omega^{-1} + \kappa^2)$; see [16, Theorem 2.4].

Also of importance are the following remarks.

Remark 1. With the notation above, the solution $(\widetilde{q}(t), \widetilde{p}(t))$ of the stiff ODEs (1) with initial condition $\widetilde{q}(t_0) = \widetilde{q}_0^\omega$, $\widetilde{p}(t_0) = \widetilde{p}_0^\omega$ and the solution $(\widehat{q}(t), \widehat{p}(t))$ of the DAEs (2) with initial condition $\widehat{q}(t_0) = \widehat{q}_0$, $\widehat{p}(t_0) = \widehat{p}_0$ differ by $\mathcal{O}(\omega^{-2})$. However, if (1) is solved with initial condition $q(t_0) = \widehat{q}_0$, $p(t_0) = \widehat{p}_0$, the solution $(q(t), p(t))$ does *not* differ from $(\widetilde{q}(t), \widetilde{p}(t))$ by $\mathcal{O}(\omega^{-2})$. In fact while it is true that $q(t) - \widetilde{q}(t) = \mathcal{O}(\omega^{-2})$, the estimation for the velocity is worse: $p(t) - \widetilde{p}(t) = \mathcal{O}(\omega^{-1})$. This happens because an $\mathcal{O}(\omega^{-2})$ change in the variable q implies an $\mathcal{O}(\omega^{-2})$ change in the oscillatory potential (3), and therefore $\mathcal{O}(\omega^{-2})$ changes in kinetic energy or $\mathcal{O}(\omega^{-1})$ changes in velocity.

Remark 2. The convergence of the slow solutions of (1) to the solution of the DAEs (2) as $\omega \to \infty$ clearly imply [5]

$$\lim_{\omega\to\infty} \omega^2 g(\widetilde{q}(t); t) = \lambda(t). \tag{4}$$

The following simple examples illustrate these points.

2.2. A linear example. The scalar equation

$$\ddot{q} = -\omega^2(q - \cos t) \tag{5}$$

has the unique slow solution

$$\widetilde{q} = A\cos t, \qquad \widetilde{p} = -A\sin t, \qquad A = A(\omega) = \frac{\omega^2}{\omega^2 - 1}. \tag{6}$$

Since here the constraints[3] are

$$g = q - \cos t = 0, \qquad \dot{g} = p + \sin t = 0,$$

the corresponding constrained problem (2) has the unique solution $\widehat{q}(t) = \cos t$, $\widehat{p}(t) = -\sin t$. The fact that $A = 1 + \mathcal{O}(\omega^{-2})$ implies that both $\widetilde{q}(t)$ and $\widetilde{p}(t)$ are $\mathcal{O}(\omega^{-2})$ away from their constrained counterparts $\widehat{q}(t)$, $\widehat{p}(t)$.

Consider now the initial value problem given by (5) together with the initial conditions $(q(t_0), p(t_0)) = (q_0, p_0)$. We write $q_0 = A\cos t_0 + \rho_0$, $p_0 = -A\sin t_0 + \sigma_0$,

[3]Here the constraint manifold $\widehat{\mathcal{P}}$ is reduced to a single (time-dependent) point.

so that ρ_0 and σ_0 measure the deviation from the "slow" initial values $\widetilde{q}(t_0) = A\cos t_0$, $\widetilde{p}(t_0) = -A\sin t_0$. The solution of the problem is given by

$$q(t) = A\cos t + \rho_0 \cos\omega(t-t_0) + \frac{\sigma_0}{\omega}\sin\omega(t-t_0), \tag{7}$$

$$p(t) = -A\sin t - \omega\rho_0 \sin\omega(t-t_0) + \sigma_0\cos\omega(t-t_0); \tag{8}$$

it oscillates with the fast frequency ω *around the slow solution*. It is important to point out the presence of the large factor ω in (8) which implies that if ρ_0 and σ_0 are $\mathcal{O}(\omega^{-2})$, the difference $p(t) - \widetilde{p}(t)$ will be $\mathcal{O}(\omega^{-1})$ in agreement with Remark 1 above. In particular if (q_0, p_0) is taken on the constraint manifold, the velocity $p(t)$ in (8) will be $\mathcal{O}(\omega^{-1})$ away from $\widetilde{p}(t)$.

Since we have $\ddot{\widetilde{q}} = -\cos t$ and $g' = 1$, we conclude from (2) that the Lagrange multiplier is $\lambda = \cos t$. On the other hand, on the slow solution $g = \widetilde{q}(t) - \cos t = (A-1)\cos t$, and therefore (4) holds.

2.3. A nonlinear example. As a second example we study a strong spring in the plane. Consider a unit point mass with coordinates (x, y) moving in a plane. It is attached to one end of a stiff harmonic spring with unit length at rest and elastic constant ω^2; the other end of the spring is linked to a pivot fixed at the origin. If $r = (x^2 + y^2)^{1/2}$, the equations of motion

$$\ddot{x} = -\omega^2(r-1)\frac{x}{r},$$
$$\ddot{y} = -\omega^2(r-1)\frac{y}{r}$$

provide a simple instance of (1) with $d = 2$, $q = (x, y)$, $k = 1$, $g = r - 1$, $F \equiv 0$. The corresponding DAEs (2) read

$$\ddot{x} = -\lambda\frac{x}{r},$$
$$\ddot{y} = -\lambda\frac{y}{r},$$
$$r = 1.$$

Clearly these equations govern the motion of a pendulum in the absence of gravity.

The problem is best analyzed after changing to polar coordinates, when the stiff ODEs become

$$\ddot{r} = -\omega^2(r-1) + r\dot{\phi}^2, \qquad \frac{d}{dt}(r^2\dot{\phi}) = 0,$$

while the DAEs become

$$\ddot{r} = -\lambda + r\dot{\phi}^2, \qquad \frac{d}{dt}(r^2\dot{\phi}) = 0, \qquad r = 1.$$

The solution of the DAEs is trivial: $r \equiv 1$ and ϕ increases linearly with t. The Lagrange multiplier $\lambda = \dot{\phi}^2$ does not change with t and represents the centripetal force exerted by the rod.

For the stiff ODEs, the areal velocity $M = (1/2)r^2\dot{\phi}$ is a constant of motion, and we may write

$$\ddot{r} = -\omega^2(r-1) + \frac{4M^2}{r^3}, \tag{9}$$

so that r varies with t as the abscissa of a unit point mass under the effective potential

$$\frac{\omega^2}{2}(r-1)^2 + \frac{2M^2}{r^2},$$

which possesses a minimum at the positive root $r^* = r^*(M,\omega)$ of the equation

$$\omega^2 r^3(r-1) = 4M^2.$$

This root obviously satisfies

$$r^* = 1 + \frac{4M^2}{\omega^2} + \mathcal{O}(\omega^{-4}).$$

If $r(0) = r^*$ and $\dot{r}(0) = 0$, then $r(t)$ and $\dot{\phi}(t)$ remain constant (the tension in the spring exactly balances the centrifugal force so as to have a uniform circular motion). Thus, in this example, the slow two-dimensional manifold $\widetilde{P}$ in the phase space $\mathbb{R}^4$ has the equations $r = r^*$ and $\dot{r} = 0$, and the constraint manifold $\widehat{P}$ is given by $r = 1$, $\dot{r} = 0$ (the tangent bundle to the unit circle); these manifolds are $\mathcal{O}(\omega^{-2})$ away from each other because $r^* = 1 + \mathcal{O}(\omega^{-2})$. Note that since r^* depends on $M = (1/2)r^2\dot{\phi}$, the relation $r = r^*$ mixes coordinates and velocities and on its own does *not* define a one-dimensional manifold in the configuration plane $\mathbb{R}^2$ of the variables (x, y); i.e., the slow manifold is not the tangent bundle of a one-dimensional configuration manifold.

If the initial values $r(0)$, $\dot{r}(0)$ do not lie on $\widetilde{P}$, the solution $r(t)$ of (9) is not constant and undergoes fast oscillations *around the equilibrium* r^* in the effective potential well. Those fast oscillations are also present in the (tangential) variable ϕ because $M = (1/2)r^2\dot{\phi}$ remains constant as t varies.

The validity of Remarks 1 and 2 in this example is easily checked.

2.4. The bounded derivative principle. Kreiss's bounded derivative principle [16], [17] provides a means to identify the slow manifold. We illustrate this for the equation (5). From the differential equation we see that for $\ddot{q}$ to be bounded independently of ω (i.e., to have a solution slow to order 2), we must impose $q = \cos t + \mathcal{O}(\omega^{-2})$. Differentiating twice in (5) leads to

$$\frac{d^4}{dt^4}q = -\omega^2(\ddot{q} + \cos t) = \omega^4 q - \omega^4 \cos t - \omega^2 \cos t,$$

and, accordingly, for solutions slow to order 4

$$q = \cos t + \frac{1}{\omega^2}\cos t + \mathcal{O}(\omega^{-4}).$$

The iteration of this procedure leads to (6). The slow manifold $r = r^*$, $\dot{r} = 0$ of the pendulum may be found in exactly the same way.

3. Iterated projection algorithm IPA. In this section we describe the iterated projection algorithm IPA. This is based on the application of filtering kernels that we discuss immediately below.

3.1. Kernels. For our purposes a (filtering) kernel is an *even* real-valued function $K(t)$ of the real variable t, $-\infty < t < \infty$, supported in the interval $[-1, 1]$, and having unit mass

$$\int_{-1}^{1} K(t)\,dt = 1.$$

In order to filter at a time t_0 a given vector-valued function v of the real variable t we compute the integral

$$v^*_{t_0} = \int_{-\infty}^{\infty} K_\delta(t) v(t+t_0)\, dt = \int_{-\delta}^{\delta} K_\delta(t) v(t+t_0)\, dt, \tag{10}$$

where $K_\delta(t)$ is the scaling of K to the segment $[-\delta, \delta]$, $\delta > 0$,

$$K_\delta(t) = \delta^{-1} K(\delta^{-1} t).$$

The performance of a kernel depends on the values of two parameters, μ, ν. The number μ of *vanishing moments* of K is the largest integer such that

$$\int_{-1}^{1} K(t) t^j\, dt = 0, \qquad j = 1, \ldots, \mu;$$

since K is an even function, $K(t) \equiv K(-t)$, μ is well defined and odd. If v is slowly varying and $\delta \ll 1$, then $v^*_{t_0}$ is close to $v(t_0)$ and the degree of closeness is governed by μ.

LEMMA 3.1. *If v is has a bounded $(\mu+1)$st derivative $v^{(\mu+1)}$, then*

$$\left| \int_{-\delta}^{\delta} K_\delta(t) v(t+t_0) dt - v(t_0) \right| \leq C ||v^{(\mu+1)}||_\infty \delta^{\mu+1} \tag{11}$$

for some constant $C > 0$ that depends only on K.

On the other hand, if v is oscillates rapidly around zero, then $v^*_{t_0} \approx 0$ and the size of $v^*_{t_0}$ is governed by the number ν of continuous derivatives of K.

LEMMA 3.2. *Assume that v is τ-periodic for some $\tau > 0$, continuous, and with zero average (i.e., $\int_0^\tau v(t)\, dt = 0$).*

1. *If K has $\nu \geq 1$ continuous derivatives, then*

$$\left| \int_{-\delta}^{\delta} K_\delta(t) v(\omega(t+t_0))\, dt \right| \leq C ||v||_\infty \frac{1}{(\omega\delta)^\nu}, \tag{12}$$

   where the constant C depends only on τ, K, and ν.
2. *The same result holds true if K, K', ..., $K^{(\nu-2)}$ are continuous and $K^{(\nu-1)}$, $K^{(\nu)}$ are continuous except perhaps at a number of points $-1 = t_1 < t_2 < \cdots < t_k = 1$, where they have jump discontinuities.*

The proofs of these lemmas are based on Taylor expansion of v and on integration by parts, respectively (see [12, Lemma 2.2][4]). Note that we use the symbol C for a generic constant that may have different values at different occurrences and that the choice

$$\delta = L/\omega, \tag{13}$$

with L a large constant, ensures that the right-hand side of (12) is small uniformly in ω.

Later the filtering process will be applied to functions v that may be well approximated by a sum of a slowly varying function u and several periodic functions w_i with frequencies $\lambda_i \omega$. Since filtering is a linear operation, $v^*_{t_0}$ will coincide with the sum of $u^*_{t_0}$ and $w_i{}^*_{t_0}$ and, according to the lemmas, will be close to $u^*_{t_0}$.

[4]Only item 1 of Lemma 3.2 is considered in [12], but the proof there works under the weaker hypothesis in item 2.

3.2. The algorithm IPA. Suppose that, at a fixed time t_0, we are given a point (q_0, p_0) which is close to the slow manifold $\widetilde{\mathcal{P}}$, and we wish to obtain a point $(\widetilde{q}, \widetilde{p}) = \mathbb{P}(q_0, p_0)$ on $\widetilde{\mathcal{P}}$ and close to (q_0, p_0). This algorithm is based on the observation that if we take (q_0, p_0) as an initial condition for (1) (i.e., $(q(t_0), p(t_0)) = (q_0, p_0)$), the corresponding solution $(q(t), p(t))$ (the so-called microsolution) will consist of the superposition of a slow component and an oscillatory part. According to Lemmas 3.1 and 3.2, filtering the microsolution will lead to a point (q_1, p_1) close to the value at t_0 of the slow component, i.e., to a point close to the slow manifold. If (q_1, p_1) is not sufficiently close for the application in mind, the procedure may be iterated. In this way the algorithm IPA is as follows:

1. Initial data: Given ϵ, L, t_0, (q_0, p_0), set $m = 0$.
2. Micro-integration: Solve the stiff problem (1) in the narrow time window $t_0 - \delta \le t \le t_0 + \delta$, with δ chosen as in (13) and with initial condition $(q(t_0), p(t_0)) = (q_m, p_m)$. Denote by $(q_m(t), p_m(t))$ the corresponding microsolution.
3. Filtering: Filter $q_m(t)$ and $p_m(t)$ with a kernel as in (10) to obtain q^*_{m,t_0}, p^*_{m,t_0}. Set $q_{m+1} = q^*_{m,t_0}$, $p_{m+1} = p^*_{m,t_0}$.
4. Stopping test: If

$$\max\left(|g(q_{m+1}, t_0) - g(q_m, t_0)|, |\dot{g}(q_{m+1}, p_{m+1}, t_0) - \dot{g}(q_m, p_m, t_0)|\right) \ge \epsilon, \tag{14}$$

set $m = m + 1$ and repeat steps 2 and 3. Otherwise set $(\widetilde{q}, \widetilde{p}) = \mathbb{P}(q_0, p_0) = (q_{m+1}, p_{m+1})$ and stop.

The stopping criterion (14) will be discussed later. In practice the micro-integrations have to be carried out numerically with a step size $h = \mathcal{O}(\omega^{-1})$ (see section 5.2 below); it is important to note that, in view of (13), the computational cost is then *independent of* ω.

4. Analysis. In this section we first analyze carefully the iterated projection algorithm as applied to two model problems that exhibit many—but not all—important features of the general situation. This analysis is essential to understanding the role of the different parameters. We then provide a less detailed study of the general case.

4.1. First model problem: Damping of oscillations. We apply the iterated projection algorithm to the linear equation (5) at time t_0 with $q_0 = A\cos t_0 + \rho_0$, $p_0 = -A\sin t_0 + \sigma_0$ (ρ_0, σ_0 quantify the deviation from the slow solution $(A\cos t_0, -A\sin t_0)$). As we know, the microsolution is given by (7)–(8).

Let us denote by $\cos t_0 + c$, $\sin t_0 + s$, and R the results of applying at time t_0 the filtering procedure (10) to the functions $\cos t$, $\sin t$, and $\cos\omega(t - t_0)$, respectively. Then filtering the microsolution (7)–(8) leads to the new values

$$q_1 = A\cos t_0 + \rho_1, \qquad p_1 = -A\sin t_0 + \sigma_1,$$

with

$$\rho_1 = Ac + R\rho_0, \qquad \sigma_1 = -As + R\sigma_0.$$

The old deviations ρ_0, σ_0 are multiplied by the factor R. We have taken into account that since we are assuming the kernel to be even, the output of filtering at t_0 the function $\sin(\omega(t - t_0))$ is 0. This fact is of particular significance because, as pointed out before, $\sin\omega(t - t_0)$ appears in (8) multiplied by the large factor ω.

After iterating the projection m times, we will have the relations

$$\begin{aligned}\rho_m &= \Big(1 + R + \cdots + R^{m-1}\Big)Ac + R^m\rho_0,\\ \sigma_m &= -\Big(1 + R + \cdots + R^{m-1}\Big)As + R^m\sigma_0\end{aligned}$$

that provide the basis for the analysis that follows.

1. According to (12)–(13), $|R| \le C/L^\nu$. It is then possible to choose L large enough to ensure that $|R| < 1$, which we assume hereafter. Then, as $m \uparrow \infty$, the errors ρ_m and σ_m will converge to

$$\rho_\infty = \frac{Ac}{1-R}, \qquad \sigma_\infty = -\frac{As}{1-R}$$

and accordingly (q_m, p_m) will converge to a limit (q_∞, p_∞) close to the slow point

$$q_\infty = \widetilde{q}(t_0) + \rho_\infty, \qquad p_\infty = \widetilde{p}(t_0) + \sigma_\infty.$$

2. The convergence factor $|R|$ of the iteration is *independent* of ω.
3. Due to the factor L^ν in the bound $|R| \le C/L^\nu$, kernels with a high number ν of continuous derivatives allow for a faster decrease of the convergence factor $|R|$ as L is increased.
4. From (11) and (13),

$$|c| \le C\frac{L^{\mu+1}}{\omega^{\mu+1}}, \qquad |s| \le C\frac{L^{\mu+1}}{\omega^{\mu+1}} \tag{15}$$

and, correspondingly, item 1 above leads to the following main estimate:

$$|q_\infty - \widetilde{q}(t_0)| = \mathcal{O}(\omega^{-(\mu+1)}), \quad |p_\infty - \widetilde{p}(t_0)| = \mathcal{O}(\omega^{-(\mu+1)}), \qquad \omega \to \infty. \tag{16}$$

5. It should be observed that the error in q_∞, p_∞ is determined by the error in *filtering the slow solution*.
6. The condition $\mu > 1$ must hold if (q_∞, p_∞) is to provide a better approximation to $(\widetilde{q}(t_0), \widetilde{p}(t_0))$ than the values $(\widehat{q}(t_0), \widehat{p}(t_0))$ of the constrained solution that give an $\mathcal{O}(\omega^{-2})$ error.
7. Since L has to be chosen large enough to ensure $|R| < 1$, the factor $L^{\mu+1}$ in (15) will result in a large size of the error constants implied in (16). The algorithm will only be accurate if ω is large with respect to L.
8. According to (14), the iteration is stopped when an integer ℓ is found such that

$$|q_{\ell+1} - q_\ell| \le \epsilon, \qquad |p_{\ell+1} - p_\ell| \le \epsilon. \tag{17}$$

It is easy to show that then

$$\ell \approx \frac{\log(1/\epsilon) + \log\big(|Ac| + |As| + |\rho_0| + |\sigma_0|\big)}{\log(1/|R|)} \tag{18}$$

and

$$|q_{\ell+1} - q_\infty| \le \frac{R}{1-R}\epsilon, \qquad |p_{\ell+1} - p_\infty| \le \frac{R}{1-R}\epsilon. \tag{19}$$

Note that since R is independent of ω, these estimates are uniform in ω.

4.2. Second model problem: Tangential errors. Next we consider the uncoupled problem for $q = (x, y) \in \mathbb{R}^2$ given by

$$\ddot{x} = F(x; t), \qquad \ddot{y} = -\omega^2(y - \cos t).$$

Here ($k = 1$)

$$g = y - \cos t = 0, \qquad \dot{g} = \dot{y} + \sin t = 0;$$

in configuration space $\mathbb{R}^2$ the constraint manifold $\widehat{\mathcal{C}}$ is a time-dependent straight line that moves parallel to the x axis. If $(x_0, y_0, \dot{x}_0, \dot{y}_0)$ is the starting point for the algorithm at $t = t_0$, we would like the iterates $(x_m, y_m, \dot{x}_m, \dot{y}_m)$ to move towards

$$(x_0, A\cos t_0, \dot{x}_0, -A\sin t_0)$$

as this is the point on $\widetilde{\mathcal{P}}$ closest to $(x_0, y_0, \dot{x}_0, \dot{y}_0)$. We know from the preceding subsection that y_m, $\dot{y}_m$ converge to the desired limits. The variable x is slow and filtering will change the initial x_0 into x_1 with $|x_1 - x_0| \le CL^{\mu+1}/\omega^{\mu+1}$ (see (11)), thus causing a small glide in configuration parallel to $\widehat{\mathcal{C}}$. At the next iteration $x_2 - x_1 \approx x_1 - x_0$ because x_1 is close to x_0, and therefore the glide from x_0 will have approximately doubled. As more projections are carried out, x_m will move away linearly from x_0 and, analogously, $\dot{x}_m$ from $\dot{x}_0$. This is markedly different from the situation for the y component, where, as we saw, errors introduced at an iteration are subsequently dampened. In spite of these tangential displacements the algorithm will succeed because the iteration is stopped according to the criterion (14), i.e., when an integer ℓ is found for which (see (17))

$$|y_{\ell+1} - y_\ell| \le \epsilon, \qquad |\dot{y}_{\ell+1} - \dot{y}_\ell| \le \epsilon.$$

In fact we know that $y_{\ell+1}$, $\dot{y}_{\ell+1}$ will be close to their target as in (19). Moreover, since the number of required iterations (18) is logarithmic in ϵ,

$$|x_{\ell+1} - x_0| = \mathcal{O}\big(\log(1/\epsilon)\omega^{-(\mu+1)}\big), \quad |\dot{x}_{\ell+1} - \dot{x}_0| = \mathcal{O}\big(\log(1/\epsilon)\omega^{-(\mu+1)}\big).$$

Note that, except for a logarithmic factor, the overall error in the algorithm is governed by the errors in filtering the slow variables x and $\dot{x}$ together with the errors in filtering the slow components of the y, $\dot{y}$ variables. In other words the error is essentially independent of the distance of the initial (q_0, p_0) to the slow manifold $\widetilde{\mathcal{P}}$.

4.3. The general case. In the preceding model problem, the variables y and $\dot{y}$ transversal to the constraints are not coupled to the tangential variables x, $\dot{x}$. In general, all variables will be coupled (as illustrated by the pendulum example of section 2.3), and, in particular, there will be an influence of the fast vibrations onto the tangential dynamics. In order to investigate these issues, we now turn to the system (1) where, to simplify the exposition, we consider that F and g are independent of t. In this autonomous case we may also take $t_0 = 0$.

4.3.1. Setup. If the algorithm starting point (q_0, p_0) is close to the constraint manifold $\widehat{\mathcal{P}}$, then its orthogonal projection $(\widehat{q}_0, \widehat{p}_0)$ onto $\widehat{\mathcal{P}}$ is well defined (more precisely, $\widehat{q}_0$ is the orthogonal projection in $\mathbb{R}^d$ of q_0 onto $\widehat{\mathcal{C}}$ and $\widehat{p}_0$ is the orthogonal projection of p_0 onto the vector space $\mathcal{T}_0$ tangent to $\widehat{\mathcal{C}}$ at $\widehat{q}_0$); see Figure 1. Similarly,

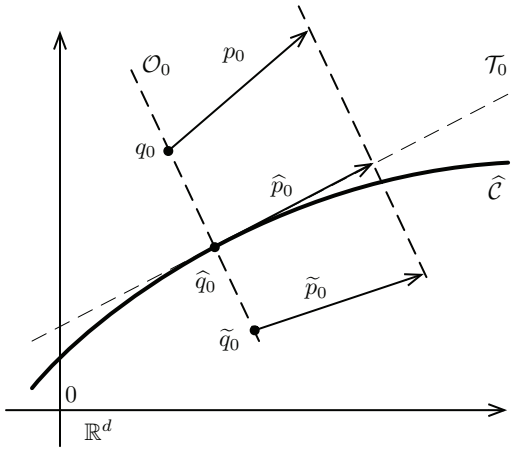

FIG. 1. *The diagram shows the space $\mathbb{R}^d$ of the configuration variable q. In the DAE q is constrained to be on $\widehat{\mathcal{C}}$ and p is constrained to be in the tangent space $\mathcal{T}_0$ at q. The point $q_0 \in \mathbb{R}^d$ and the velocity vector p_0 at q_0 are projected onto the constraint manifold $\widehat{\mathcal{P}}$ to yield $\widehat{q}_0$ and $\widehat{p}_0$ and onto the slow manifold $\widetilde{\mathcal{P}}$ to yield $\widetilde{q}_0$ and $\widetilde{p}_0$.*

let $(\widetilde{q}_0, \widetilde{p}_0)$ be the (ω-dependent) point on the slow manifold $\widetilde{\mathcal{P}}$ whose orthogonal projection onto $\widehat{\mathcal{P}}$ coincides with $(\widehat{q}_0, \widehat{p}_0)$; for ω large, $(\widetilde{q}_0, \widetilde{p}_0)$ is well defined and

$$|\widetilde{q}_0 - \widehat{q}_0| + |\widetilde{p}_0 - \widehat{p}_0| = \mathcal{O}(\omega^{-2}). \tag{20}$$

The next iterate (q_1, p_1) is obtained by filtering $(q(t), p(t))$, $|t| \le L/\omega$, the microsolution starting from (q_0, p_0), and our aim is to show that (q_1, p_1) is closer to $(\widetilde{q}_0, \widetilde{p}_0)$ than (q_0, p_0).

In a neighborhood of $\widehat{q}_0$ in configuration space, we will decompose the dynamics of (1) into components orthogonal and tangent to $\widehat{\mathcal{C}}$. Let $\mathcal{Q}_\perp$ and $\mathcal{Q}_\parallel$, respectively, denote full-rank $k \times d$ and $(d-k) \times d$ matrices such that the kernel of $\mathcal{Q}_\perp$ is the vector space $\mathcal{T}_0$ tangent to $\widehat{\mathcal{C}}$ at $\widehat{q}_0$, the kernel of $\mathcal{Q}_\parallel$ is the vector space $\mathcal{O}_0$ orthogonal to $\widehat{\mathcal{C}}$ at $\widehat{q}_0$, and, for each $v \in \mathbb{R}^d$,

$$|v|^2 = |\mathcal{Q}_\perp v|^2_{\mathbb{R}^k} + |\mathcal{Q}_\parallel v|^2_{\mathbb{R}^{d-k}}.$$

In particular $\mathcal{Q}_\perp$ restricted to the k-dimensional subspace $\mathcal{O}_0 \subset \mathbb{R}^d$ is an isometry onto $\mathbb{R}^k$, and $\mathcal{Q}_\parallel$ restricted to the $(d-k)$-dimensional subspace $\mathcal{T}_0 \subset \mathbb{R}^d$ is an isometry onto $\mathbb{R}^{d-k}$. Furthermore the $d \times d$ matrices $\mathcal{Q}_\perp^T \mathcal{Q}_\perp$ and $\mathcal{Q}_\parallel^T \mathcal{Q}_\parallel$ are the orthogonal projectors of $\mathbb{R}^d$ onto $\mathcal{O}_0$ and $\mathcal{T}_0$, respectively.

The definitions of $\widehat{q}_0$ and $\widehat{p}_0$ imply

$$\mathcal{Q}_\parallel(q_0 - \widehat{q}_0) = 0, \qquad \mathcal{Q}_\parallel(p_0 - \widehat{p}_0) = 0, \qquad \mathcal{Q}_\perp \widehat{p}_0 = 0. \tag{21}$$

Let us denote by κ an upper bound for the orthogonal components of $q_0 - \widehat{q}_0$ and $p_0 - \widehat{p}_0$:

$$|\mathcal{Q}_\perp(q_0 - \widehat{q}_0)| \le \kappa, \qquad |\mathcal{Q}_\perp(p_0 - \widehat{p}_0)| = |\mathcal{Q}_\perp p_0| \le \kappa. \tag{22}$$

We have to assume that κ is small: in a nonlinear situation it is not expected that

the algorithm will work if it starts from (q_0, p_0) far from the constraint manifold.[5] However, the applications presented later in the paper demand that no relation is assumed between κ and $1/\omega$; these must be seen as independent small parameters. The behavior of the microsolution $(q(t), p(t))$ varies substantially with the size of κ relative to $1/\omega$. For instance, if $\kappa \approx 1/\omega^2$, then the energy in the transversal oscillations will be small and the dynamics of the microsolution will be close to that of the slow solution. The case $\kappa \approx 1/\omega$ treated in [14] corresponds to oscillatory energy that remains bounded as $\omega \to \infty$. On the other hand, larger values of κ will allow fast oscillations with larger amplitudes, which, in principle, may have a significant influence on the slow components of q, p and therefore impair the performance of the algorithm. In what follows we restrict our attention to the *worst* case where κ is large relatively to $1/\omega$ (large oscillatory energy). In that scenario the distance from (q_0, p_0) to $(\widehat{q}_0, \widehat{p}_0)$ (of order κ) is much larger than the distance (see (20)) between $(\widehat{q}_0, \widehat{p}_0)$ and $(\widetilde{q}_0, \widetilde{p}_0)$, so that we may consider that our aim is to show that (p_1, q_1) is closer to $(\widehat{q}_0, \widehat{p}_0)$ than to (q_0, p_0).

After decomposing $q(t) = \widehat{q}_0 + (q(t) - \widehat{q}_0)$ and observing that filtering is a linear process, it is clear that we have to show that the outputs of filtering the functions

$$y(t) = \mathcal{Q}_\perp(q(t) - \widehat{q}_0), \qquad x(t) = \mathcal{Q}_\parallel(q(t) - \widehat{q}_0) \tag{23}$$

are sufficiently small, while the outputs of filtering the functions

$$\dot{y}(t) = \mathcal{Q}_\perp p(t), \qquad \dot{x}(t) = \mathcal{Q}_\parallel p(t) \tag{24}$$

are close to $\mathcal{Q}_\perp \widehat{p}_0 = 0$ (see (21)) and to $\mathcal{Q}_\parallel \widehat{p}_0$, respectively.

4.3.2. Transversal oscillations. From (23) and (22), it follows that $y(0) = \mathcal{O}(\kappa)$. Similarly (24) and (22) lead to $\dot{y}(0) = \mathcal{O}(\kappa)$. From (1) the differential equation for $y \in \mathbb{R}^k$ is

$$\ddot{y} = \mathcal{Q}_\perp F(q) - \omega^2 \mathcal{Q}_\perp g'(q)^T g(q). \tag{25}$$

In the short micro-integration interval $|t| \le L/\omega$, the value of $q(t)$ will not deviate much[6] from its initial value $q(0)$, and therefore we may replace $g'(q)$ by the constant $g'_0 = g'(\widehat{q}_0)$. We also note that

$$g(q(t)) = g(q(t)) - g(\widehat{q}_0) \approx g'(\widehat{q}_0)(q(t) - \widehat{q}_0) = g'_0(q(t) - \widehat{q}_0)$$

and, because g'_0 is in $\mathcal{O}_0$,

$$g'_0(q(t) - \widehat{q}_0) = g'_0 \mathcal{Q}_\perp^T \mathcal{Q}_\perp (q(t) - \widehat{q}_0) = g'_0 \mathcal{Q}_\perp^T y(t).$$

Finally, $\mathcal{Q}_\perp F(q) = \mathcal{O}(1)$, while the second term in the right-hand side of (25) is $\mathcal{O}(\kappa\omega^2)$ and is therefore large relatively to ω (recall that we are studying the regime where $\kappa\omega$ is large). In this way, going back to (25), we may write the approximation

$$\ddot{y} = -\omega^2 M_0 y, \qquad M_0 = \mathcal{Q}_\perp {g'_0}^T g'_0 \mathcal{Q}_\perp^T,$$

[5] Note that if (q_0, p_0) is far from the constraint manifold, the definition of $(\widehat{q}_0, \widehat{p}_0)$ may not be possible.

[6] Roughly speaking, $x(t) - x(0) = \mathcal{O}(\omega^{-1})$ and $y(t) - y(0) = \mathcal{O}(\kappa)$. This may easily be rendered rigorous by a Gronwall argument.

with a relative error $\mathcal{O}(\kappa)$ in the $\mathcal{O}(\kappa\omega^2)$ right-hand side. The matrix M_0 is symmetric and, since g_0' has full rank, it is also positive definite; therefore

$$y(t) = \cos(\omega\sqrt{M_0}t)y(0) + (\omega\sqrt{M_0})^{-1}\sin(\omega\sqrt{M_0}t)\dot{y}(0), \tag{26}$$

$$\dot{y}(t) = -\omega\sqrt{M_0}\sin(\omega\sqrt{M_0}t)y(0) + \cos(\omega\sqrt{M_0}t)\dot{y}(0). \tag{27}$$

These equations are very similar to (7)–(8), which were analyzed in detail. The variable $y(t)$ undergoes linear oscillations with amplitude $\mathcal{O}(\kappa)$ and frequencies $\omega\lambda_i > 0$, where λ_i^2, $i = 1, \ldots, k$, are the eigenvalues of M_0. Thus the filtering procedure, after choosing the filtering interval length appropriately, will dampen y and produce a point (q_1, p_1) closer to $\widehat{\mathcal{P}}$ than the original (q_0, p_0). Again the choice of an even filter is essential for dealing with the large first term in the right-hand side of (27).

Of course the analysis above may then be taken up from (q_1, p_1) in lieu of (q_0, p_0) with a correspondingly smaller value of the parameter κ.

4.3.3. Tangential dynamics. The initial conditions for the tangential part of the microsolution are $x(0) = 0 \in \mathbb{R}^{d-k}$ (see (21) and (23)) and $\dot{x}(0) = \mathcal{Q}_\parallel p_0 = \mathcal{Q}_\parallel \widehat{p}_0$ (see (21) and (24)). The projection of (1) onto $\mathcal{T}_0$ leads to

$$\ddot{x} = \mathcal{Q}_\parallel F(q) - \omega^2 \mathcal{Q}_\parallel g'(q)^T g(q) \approx \mathcal{Q}_\parallel F(q) - \omega^2 \mathcal{Q}_\parallel g'(q)^T g_0' \mathcal{Q}_\perp y(t).$$

The product $\mathcal{Q}_\parallel g'(q)$ vanishes at $q = \widehat{q}_0$ and, thus, the acceleration $\ddot{x}$ will not attain values of magnitude $\mathcal{O}(\kappa\omega^2)$ as $\ddot{y}$ does. Freezing $g'(q)^T$ at $q = \widehat{q}_0$ as we did in (25) will not do now, and we have to consider first-order terms:

$$g'(q(t))^T \approx {g_0'}^T + B_0(q(t) - \widehat{q}_0)$$

(B_0 is the corresponding Jacobian matrix). Moreover, since $\mathcal{Q}_\parallel(q(t) - \widehat{q}_0)$ is negligible with respect to $\mathcal{Q}_\perp(q(t) - \widehat{q}_0)$, we may replace $(q(t) - \widehat{q}_0)$ by its projection onto $\mathcal{O}_0$, and that would result in an equation

$$\ddot{x} = \mathcal{Q}_\parallel F(q) - \omega^2 C_0[y(t), y(t)]$$

for a suitable bilinear operator C_0, whose actual form is of no consequence. Finally, since $\omega^2 C_0[y(t), y(t)] = \mathcal{O}(\kappa^2\omega^2)$ is large, we may disregard $\mathcal{Q}_\parallel F(q)$ to obtain

$$\ddot{x} = \omega^2 C_0[y(t), y(t)], \tag{28}$$

an equation which shows that, in the regime under consideration, the tangential acceleration is mainly induced by the transversal fast oscillations. (This coupling is of course not present in simple cases where g' is constant and $B_0 = 0$.)

From (26), we know that $y(t)$ is a combination of harmonic contributions with $\mathcal{O}(\kappa)$ amplitudes and frequencies $\omega\lambda_i$; then the right-hand side of (28) will be a combination of sine and cosine functions with amplitudes $\mathcal{O}(\kappa^2)$ and frequencies $\omega|\lambda_i \pm \lambda_j|$. In particular the resonant combinations $\omega|\lambda_i - \lambda_i| = 0$ will be present. In other words

$$\ddot{x} = \omega^2(\ell + D(t)),$$

where ℓ is an $\mathcal{O}(\kappa^2)$ constant vector and $D(t)$ is oscillatory with Fourier modes with $\mathcal{O}(\kappa^2)$ amplitudes and large frequencies proportional to ω. Additionally, since the first term in the right-hand side of (26) is dominant, the function $D(t)$ is even, except for an $\mathcal{O}(\kappa^2/\omega)$ perturbation.

Integrating once, we may write

$$\dot{x}(t) = \mathcal{Q}_{\|}\widehat{p}_0 + \omega^2 t\ell + \int_0^t \omega^2 D(s)\, ds.$$

Here the large $\omega^2 t\ell$ term is annihilated by the (even) filter, and the same is true for the dominant odd part of the integral. The even part of the integral is highly oscillatory with small $\mathcal{O}(\kappa^2)$ amplitude (integration brings in a $1/\omega$ factor); its contribution to p_1 will also be $\mathcal{O}(\kappa^2)$, but with a smaller error constant by virtue of the dampening effected by the filter. Thus $\mathcal{Q}_{\|}p_1$ will be $\mathcal{O}(\kappa^2)$ away from $\mathcal{Q}_{\|}\widehat{p}_0$.

A new integration yields

$$x(t) = t\mathcal{Q}_{\|}\widehat{p}_0 + \omega^2 \frac{t^2}{2}\ell + \int_0^t ds \int_0^s \omega^2 D(u)\, du;$$

the iterated integral is oscillatory with small $\mathcal{O}(\kappa^2)$ amplitude, and $t\mathcal{Q}_{\|}\widehat{p}_0$ will be destroyed by the filter. The quadratic term $\omega^2(t^2/2)\ell$, $|t| \le L/\omega$, will at most produce a small $\mathcal{O}(\kappa^2)$ contribution to q_1; in fact for a kernel with $\mu \ge 3$ vanishing moments, that contribution will actually vanish. We conclude that $\mathcal{Q}_{\|}(q_1 - \widehat{q}_0) = \mathcal{O}(\kappa^2)$; the tangential gliding of q_1, p_1 will be negligible relative to the distance between the starting point and the slow manifold.

5. Numerical experiments: Computing slow points. We now provide examples on the application of IPA. This section illustrates the direct application of the technique to find slow points and compute Lagrange multipliers. The next section explores possible combinations of the projection algorithm with ODE solvers.

5.1. A test problem. We use the test problem with $d = 4$, $q = (x_1, y_1, x_2, y_2)$ given by[7]

$$\begin{aligned}
\ddot{x}_1 &= -\omega_1^2(r_1 - 1)\frac{x_1}{r_1} - \omega_2^2(r_{1,2} - 1)\frac{x_1 - x_2}{r_{1,2}},\\
\ddot{y}_1 &= -\omega_1^2(r_1 - 1)\frac{y_1}{r_1} - \omega_2^2(r_{1,2} - 1)\frac{y_1 - y_2}{r_{1,2}},\\
\ddot{x}_2 &= \qquad\qquad + \omega_2^2(r_{1,2} - 1)\frac{x_1 - x_2}{r_{1,2}},\\
\ddot{y}_2 &= \qquad\qquad + \omega_2^2(r_{1,2} - 1)\frac{y_1 - y_2}{r_{1,2}},
\end{aligned}$$

where

$$r_1 = (x_1^2 + y_1^2)^{1/2}, \qquad r_{1,2} = \big((x_1 - x_2)^2 + (y_1 - y_2)^2\big)^{1/2}.$$

For this system, $k = 2$ and

$$g_1(q) = r_1 - 1, \quad g_2(q) = r_{1,2} - 1.$$

These differential equations govern the planar motion of two unit point masses: the first is joined to the origin through a stiff spring of elastic constant $\omega_1^2 \gg 1$ and the second is joined to the first through a second stiff spring of elastic constant $\omega_2^2 \gg 1$.

[7]Note that now x and y represent Cartesian coordinates in the plane, and not tangential and transversal components as they did in the preceding section.

In the limit where $\omega_1 \to \infty$, $\omega_2 \to \infty$, the springs become rigid rods of unit length and the system becomes a double pendulum with no gravity. The dynamics of this simple test problem includes all the features we are interested in (fast transversal oscillations, coupling from the fast oscillations to the slow tangential dynamics, etc.). More realistic problems with large values of d will be considered in future work.

The experiments reported below have $\omega_1 = \omega_2 = \omega$; simulations with $\omega_1 = \omega$, $\omega_2 = \alpha\omega$, with α a constant of moderate size, were also carried out, but do not provide additional insights. We also performed experiments where an external force of moderate size, such as gravity, was added to the model; again the corresponding results are very similar to those presented here.

The algorithm (and the underlying theory) require that $\omega \gg 1$. The experiments reported have $\omega = 100, 1000, 10000$ (spring constants 10^4, 10^6, 10^8); Figures 2 to 6 indicate that simulations with ω significantly smaller than 100 would yield unacceptably large errors.[8]

5.2. Algorithmic details. All experiments in this paper use the piecewise cubic filter from [23]:

$$K(t) = \begin{cases} 2 - 2|t| - 8|t|^2 + 8|t|^3, & 0 \le |t| \le 1/2, \\ 2 - \frac{22}{3}|t| + 8|t|^2 - \frac{8}{3}|t|^3, & 1/2 < |t| \le 1. \end{cases}$$

This has $\mu = 3$ vanishing moments and satisfies the hypothesis in item 2 of Lemma 3.2 with $\nu = 2$. The parameter L that determines the filtering window (see (13)) is taken $L = 6\pi$ in all experiments. Micro-integrations are performed by the standard Verlet algorithm with step size $h = (2\pi/\omega)/6$; since the micro-integration window has length proportional to $1/\omega$ the number of steps in each micro-integration is independent of ω.

The tolerance ϵ in the stopping criterion (14) is 10^{-9} in the experiments with $\omega = 1000$ or $\omega = 10000$ and 10^{-7} when $\omega = 100$.

5.3. Projecting onto the slow and constraint manifolds. We used the algorithm to project onto $\widetilde{\mathcal{P}}$ the point

$$x_1 = 1, \qquad y_1 = 0.25, \qquad x_2 = 2, \qquad y_2 = 0, \tag{29}$$

$$\dot{x}_1 = 0, \qquad \dot{y}_1 = -0.5, \qquad \dot{x}_2 = 0, \qquad \dot{y}_2 = 0.5, \tag{30}$$

when $\omega = 1000$ and 10000; the choice of t_0 is inconsequential as the problem is autonomous. For both values of ω the algorithm takes five iterations to meet the required tolerance of 10^{-9}. Table 1 displays the evolution, as the iteration proceeds, of the values of the holonomic constraint functions g_1, g_2 and the implied velocity constraints $\dot{g}_1$, $\dot{g}_2$. At the starting configuration in (29) the violation of the holonomic constraint is of the order of 10^{-2}; for $\omega = 10000$ this implies an enormous $\approx 10^5$ initial potential energy in the springs. We observe that in the first iteration the behavior of the algorithm is the same for both values of ω. Away from the slow/constraint manifolds, the main effect of each iteration is the dampening of the fast oscillations in the microsolution, and it was shown in the preceding section that such dampening is uniform in ω. For $\omega = 1000$ the size of the components of g, $\dot{g}$ converge to values $\approx 10^{-6}$. For $\omega = 10000$ the limit values are 100 times smaller, in agreement with the

[8] The minimal value of ω for the algorithm to operate successfully is of course problem-dependent. It also depends crucially on the quality of the filter: if the constant C in (12) is large, then δ has to be large, and this decreases the accuracy in recovering the slow components; see (11).

TABLE 1

Reduction of the constraint functions as the iterated projection proceeds from the starting point (29)–(30).

	$\omega = 1000$			
	g_1	g_2	$\dot{g}_1$	$\dot{g}_2$
0	3.08(−2)	3.08(−2)	−1.21(−1)	−2.42(−1)
1	−3.40(−4)	−2.41(−4)	2.36(−3)	5.25(−3)
2	9.71(−7)	8.11(−7)	2.01(−5)	1.18(−5)
3	1.01(−6)	8.95(−7)	2.43(−6)	1.61(−6)
4	1.01(−6)	8.95(−7)	2.43(−6)	1.61(−6)
5	1.01(−6)	8.95(−7)	2.43(−6)	1.61(−6)

	$\omega = 10000$			
	g_1	g_2	$\dot{g}_1$	$\dot{g}_2$
0	3.08(−2)	3.08(−2)	−1.21(−1)	−2.42(−1)
1	−3.40(−4)	−2.41(−4)	2.34(−3)	5.28(−3)
2	−3.00(−8)	7.49(−8)	1.75(−5)	1.02(−5)
3	1.01(−8)	8.91(−9)	3.00(−8)	1.45(−8)
4	1.01(−8)	8.95(−9)	2.43(−8)	1.61(−8)
5	1.01(−8)	8.95(−9)	2.43(−8)	1.62(−8)

fact that the slow manifold is $\mathcal{O}(\omega^{-2})$ away from the constraint manifold $g = \dot{g} = 0$. As a side remark, let us comment that, in view of (4), the table shows that the Lagrange multipliers associated with g_1 and g_2 have values 1.01 and 0.895, respectively.

Note that the closeness between the slow and constraint manifold for ω large opens the possibility for an alternative use of the algorithm: in order to find initial data $(\widehat{q}_0, \widehat{p}_0))$ for the DAEs (2) one may choose a very large value of ω and project onto the slow manifold an initial guess (q_0, p_0). Of importance here is the fact that, as pointed out above, the computational cost of the projection is independent of ω.

As a second example we started from the point

$$\widehat{x}_1 = 1, \qquad \widehat{y}_1 = 0, \qquad \widehat{x}_2 = 2, \qquad \widehat{y}_2 = 0, \tag{31}$$

$$\dot{\widehat{x}}_1 = 0, \qquad \dot{\widehat{y}}_1 = -0.5, \qquad \dot{\widehat{x}}_2 = 0, \qquad \dot{\widehat{y}}_2 = 0.5, \tag{32}$$

which clearly satisfies the constraints $g(q) = 0$, $g'(q)p = 0$ and therefore is not far away from the slow manifold. With $\omega = 1000$ after two iterations the algorithm yields a point

$$\widetilde{x}_1 \approx 1.00000150, \quad \widetilde{y}_1 = 0, \quad \widetilde{x}_2 \approx 2.00000275, \quad \widetilde{y}_2 = 0, \tag{33}$$

$$\dot{\widetilde{x}}_1 = 0, \quad \dot{\widetilde{y}}_1 \approx -0.4999951, \quad \dot{\widetilde{x}}_2 = 0, \quad \dot{\widetilde{y}}_2 \approx 0.4999973. \tag{34}$$

Both springs have been stretched slightly as a consequence of the centrifugal effect of the nonzero tangential velocities. For $\omega = 10000$, two iterations are also required and the slow point given by the algorithm is

$$\widetilde{x}_1 \approx 1.0000000150, \qquad \widetilde{y}_1 = 0,$$
$$\widetilde{x}_2 \approx 2.0000000275, \qquad \widetilde{y}_2 = 0, \tag{35}$$

$$\dot{\widetilde{x}}_1 = 0, \qquad \dot{\widetilde{y}}_1 \approx -0.499999952,$$
$$\dot{\widetilde{x}}_2 = 0, \qquad \dot{\widetilde{y}}_2 \approx 0.499999973. \tag{36}$$

These results show once more the $\mathcal{O}(\omega^{-2})$ distance between $\widehat{\mathcal{P}}$ and $\widetilde{\mathcal{P}}$.

5.4. Computing Lagrange multipliers. Assume that it is required to integrate the DAEs (2) with compatible initial data $\widehat{q}(t_0)$, $\widehat{p}(t_0)$. At each step, the DAE solver will use an iteration to find the next values of q, p, λ. While at the very first step the initial data $\widehat{q}(t_0)$, $\widehat{p}(t_0)$ provide good starting guesses to initialize q and p in the iteration, it may be difficult to initialize the Lagrange multipliers for which no initial condition is available. After recalling (see (4)) that $\lambda \approx \omega^2 g(\widetilde{q}(t_0))$, we may apply the iterative projection algorithm to approximate λ. Here is an example. For the configuration in (33), $g_1 = 1.49 \times 10^{-6}$, $g_2 = 1.23 \times 10^{-6}$, and for the configuration in (35), $g_1 = 1.50 \times 10^{-8}$, $g_2 = 1.25 \times 10^{-8}$. We conclude that the value of the Lagrange multipliers λ_1, λ_2 at the point (31)–(32) have values ≈ 1.50 and ≈ 1.25, respectively. If the multipliers have to be determined with great accuracy, it is of course possible to resort to a Richardson extrapolation with respect to ω.

6. Numerical experiments: Computing slow solutions. In this section we discuss several possibilities for numerically computing slow solutions $(\widetilde{q}(t), \widetilde{p}(t))$. As an example we use again the test problem of the preceding section. We shall compute the slow solution for $0 \le t \le 10$ which coincides at $t = 0$ with the slow point $(\widetilde{q}_0, \widetilde{p}_0)$ obtained by projecting the initial point (31)–(32) onto $\widetilde{\mathcal{P}}$. We shall report errors in the variables q and p at the final time $t = 10$ measured in the Euclidean norm in $\mathbb{R}^4$; of course here "error" refers to the difference between the result provided by the particular algorithm being tested and the true $\widetilde{q}(10)$ or $\widetilde{p}(10)$ (these were found by the algorithm in section 6.3 with `RelTol` $= 10^{-12}$, `AbsTol` $= 10^{-14}$).

We emphasize that *the aim of this experiments is to test the iterative projection algorithm* and to illustrate the theory that underlies it; a full comparison between the efficiency of the different approaches is out of the scope of this contribution. For this reason we limit the methods tested to simple integrators like RATTLE, the classical fourth-order Runge–Kutta formula, or the codes in the MATLAB suite. When comparing the results of the different approaches, one should keep in mind the small dimensionality of the test problem and recall that while the work in explicit ODE solvers grows linearly with the dimension d of the problem, DAE algorithms are necessarily implicit and may require an amount of work that grows like a higher power of d (the details vary with the implicit solver and with the sparsity of the problem).

6.1. Solving the DAE. It is clearly possible to obtain an approximation to $(\widetilde{q}(t), \widetilde{p}(t))$ by solving numerically the associated DAE with initial condition $(\widehat{q}(0), \widehat{p}(0))$. We have implemented the well-known method RATTLE (see, e.g., [18]), with the implicit equations solved by Newton iteration and the required Jacobian matrices computed analytically. The errors in approximating the slow solution are given in Figure 2, where the step sizes are $H = 1/8, 1/16, \ldots$ and $\omega = 100, 1000, 10000$ (in order to render comparisons easier, the axes in this figure and those in the figures that follow share the same scaling). As we know, the difference between the exact DAE solution and the target $(\widetilde{q}(t), \widetilde{p}(t))$ is $\mathcal{O}(\omega^{-2})$ in both q and p, and this explains the saturation of errors as $H \to 0$. As a consequence, for $\omega = 100$ this method may only provide a very rough approximation to the solution sought. We note that this limitation will not disappear if RATTLE were replaced by a more accurate DAE solver.

6.2. Starting from the constraint manifold. Next we try to compute numerically $(\widetilde{q}(t), \widetilde{p}(t))$ by solving the stiff ODE with the point (31)–(32) as an initial condition. This point is compatible with the constraints and accordingly $\mathcal{O}(\omega^{-2})$ away from the correct initial data $(\widetilde{q}(0), \widetilde{p}(0))$. We performed this integration with *each* of

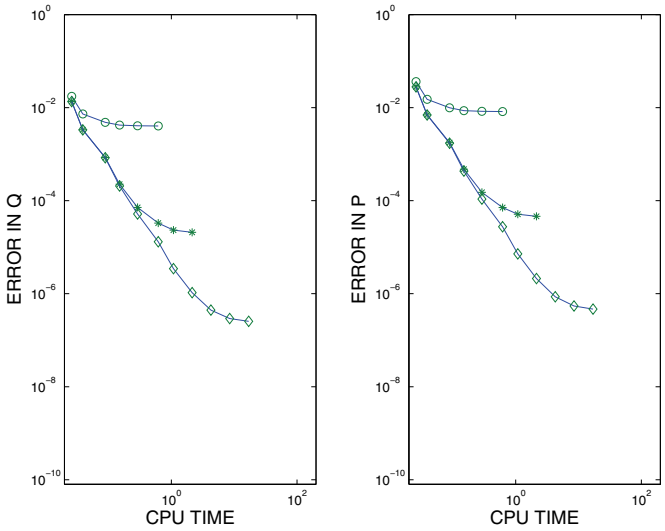

FIG. 2. *Errors for $\omega = 100$ (circles), $\omega = 1000$ (stars), $\omega = 10000$ (diamonds) when the slow solution is approximated by numerical integration of the DAE.*

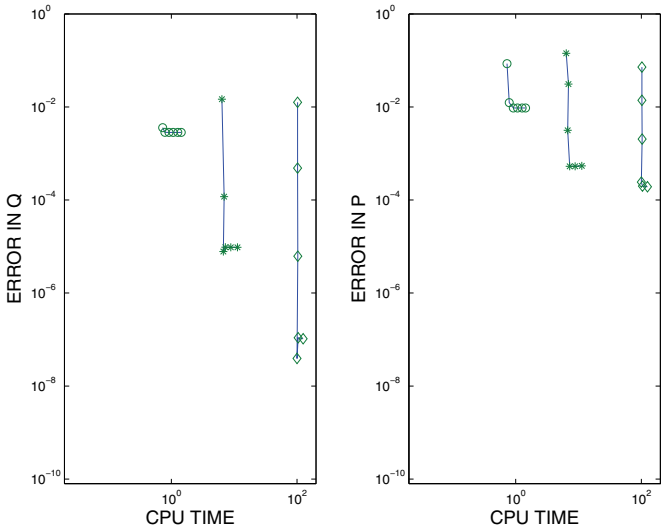

FIG. 3. *Errors for $\omega = 100$ (circles), $\omega = 1000$ (stars), $\omega = 10000$ (diamonds) when the slow solution is approximated by numerical integration with* `ode113` *starting from the constraint manifold.*

the codes from the MATLAB suite, including the *implicit* solvers `ode15s`, `ode23s`, `ode23t`, `ode23tb`; the results reported correspond to the explicit high-order Adams code `ode113` as this turned out to be the most efficient. Six pairs of relative and absolute tolerances were tested: (`RelTol, AbsTol`) $= (10^{-3}, 10^{-6})$ (the default option); $(10^{-4}, 10^{-7})$; ...; $(10^{-8}, 10^{-11})$. The results are given in Figure 3. The code chooses very small step sizes. Moving from $\omega = 100$ to $\omega = 1000$ multiplies the computational effort by 10; and the increase is even larger when going from $\omega = 1000$ to $\omega = 10000$. With larger tolerances, the computational effort is virtually independent of (`RelTol`,

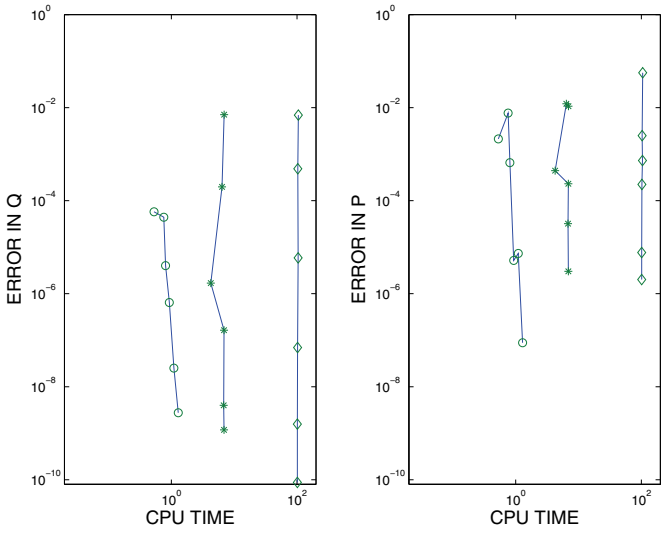

FIG. 4. *Errors for $\omega = 100$ (circles), $\omega = 1000$ (stars), $\omega = 10000$ (diamonds) when the slow solution is approximated by numerical integration with* `ode113` *starting from the slow manifold.*

`AbsTol`); for those tolerances the choice of step size is dictated by stability considerations, and demanding additional accuracy does not require a substantial decrease in step size with a high-order integrator. However, for stringent tolerances, the code tries to follow accurately the small ripples that result from the starting point being close to the slow manifold $\widetilde{\mathcal{P}}$ but not on it. As we know from Remark 1 in section 2.1, those ripples are of amplitude $\mathcal{O}(\omega^{-2})$ in q and $\mathcal{O}(\omega^{-1})$ in p. In that regime of stringent tolerances, a decrease in tolerance does imply an increase in computational effort. At the same time, in that regime, the reported errors saturate, because the `ode113` integration errors become negligible with respect to the difference between $(\widetilde{q}(t), \widetilde{p}(t))$ and $(\widehat{q}(t), \widehat{p}(t))$. For $\omega = 1000$ the saturation takes place at around 10^{-5} in q and 10^{-3} in p, for $\omega = 10000$ at around 10^{-7} and 10^{-4}, respectively. Clearly this technique is both more expensive and more inaccurate than that based on solving the DAE.

6.3. Projecting the initial condition onto the slow manifold. Next we integrated with each of the codes in the MATLAB suite the stiff ODE but using the iterative projection algorithm to identify the correct starting point $(\widetilde{q}(0), \widetilde{p}(0))$. Again the best performance corresponded to `ode113`, whose results are presented in Figure 4. Comparison with Figure 3 shows that the iterative projection destroys the saturation of errors, and it is now possible to obtain high accuracy. However, the computational costs here are high and approximately coincide with those in Figure 3: we are still solving the stiff equation and, even though the numerical solution starts on the slow manifold, the integration errors will force the solution to leave that manifold. We also note that the errors in p are considerably larger than the errors in q.

We remark that also for the implicit codes `ode15s`, `ode23s`, `ode23tb` the computational load for a given tolerance is roughly the same whether the starting point is $(\widehat{q}(0), \widehat{p}(0))$ or $(\widetilde{q}(0), \widetilde{p}(0))$. Only for `ode23t`, based on the trapezoidal rule, integrations from $(\widehat{q}(0), \widehat{p}(0))$ are more expensive than integrations from $(\widetilde{q}(0), \widetilde{p}(0))$ (the trapezoidal rule differs from the other implicit methods tested in that it is not strictly stable at infinity).

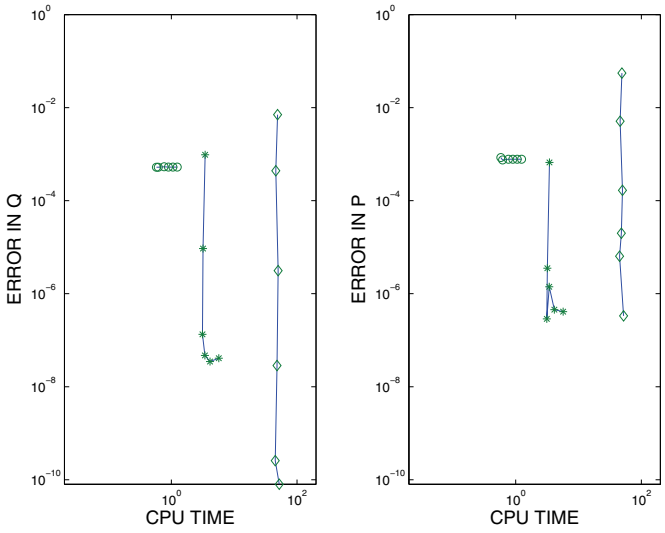

FIG. 5. *Errors for $\omega = 100$ (circles), $\omega = 1000$ (stars), $\omega = 10000$ (diamonds) when the slow solution is approximated by numerical integration with* `ode113` *starting from the slow manifold and projected back at $t = 1, 2, \ldots, 9$.*

6.4. Projecting periodically onto the slow manifold. As in the preceding subsection, the integration starts from $(\widetilde{q}(0), \widetilde{p}(0))$ and the results reported correspond to `ode113`. At times $t = 1, 2, \ldots, 9$, the integration is halted and the numerical solution projected onto the slow manifold $\widetilde{\mathcal{P}}$; the integration is then taken up from the projected point. (Note that in order not to give this technique an unfair advantage, we do not project at the final time where errors are measured.) The results may be seen in Figure 5. For $\omega = 1000$ and $\omega = 10000$ this technique is a clear improvement over that of the preceding subsection: the errors are smaller and the computational cost lower because the solution is not allowed to move too far away from the slow manifold. For $\omega = 100$ the errors are independent of the tolerance as the error of integrating with `ode113` is dominated by the $\mathcal{O}(\omega^{-4})$ errors of the projections at $t = 2, 3, \ldots$. In this connection we also observe that for $\omega = 1000$ the errors saturate at $\approx 5 \times 10^{-8}$ in q and 5×10^{-7} in p, again due to the presence of the projection errors.

It is clear that the projection may be applied more or less frequently than at $t = 1, 2, \ldots$, perhaps monitoring the growth in the constraint functions $q(q)$, $g'(q)p$ as the integration proceeds. Rather than attempting to fine-tune the frequency of the projections, we explore in the next subsection the possibility of projecting at each function evaluation of the integrator so as to pick information only from the slow manifold.

6.5. Projection onto the slow manifold at each function evaluation. To be definite we shall illustrate the technique by means of the classical fourth-order Runge–Kutta (RK4) scheme, but it may be combined with any other integrator.

The system being solved is written in first-order format $\dot{z} = \phi(z)$, $z = (q, p)$; given the approximation z_n at the beginning of a step, RK4 evaluates ϕ at four stage

vectors,

$$\begin{aligned} Z^1 &= z_n, \\ Z^2 &= z_n + \frac{H}{2}\phi(Z^1), \\ Z^3 &= z_n + \frac{H}{2}\phi(Z^2), \\ Z^4 &= z_n + H\phi(Z^3), \end{aligned}$$

and then sets

$$z_{n+1} = z_n + \frac{H}{6}\big(\phi(Z^1) + 2\phi(Z^2) + 2\phi(Z^3) + \phi(Z^4)\big). \tag{37}$$

We change the computation of the evaluation points into (as before, $\mathbb{P}$ stands for iterative projection)

$$\begin{aligned} Z^1 &= \mathbb{P}z_n, \\ Z^2 &= \mathbb{P}\left(z_n + \frac{H}{2}\phi(Z^1)\right), \\ Z^3 &= \mathbb{P}\left(z_n + \frac{H}{2}\phi(Z^2)\right), \\ Z^4 &= \mathbb{P}\big(z_n + H\phi(Z^3)\big) \end{aligned}$$

and replace (37) by

$$z_{n+1} = \mathbb{P}z_n + \frac{H}{6}\big(\phi(Z^1) + 2\phi(Z^2) + 2\phi(Z^3) + \phi(Z^4)\big). \tag{38}$$

Thus we only evaluate ϕ at points on the slow manifold. The modification requires using $\mathbb{P}$ four times per time step.

Results are reported in Figure 6 for step sizes $H = 1/2, 1/4, \ldots$. We first note that, as distinct from the other experiments based on integrating the stiff ODE, the integration now works satisfactorily with large step-lengths commensurate with the slowly varying character of the target solution. However, the savings afforded by the large values of the step-length are partly offset by the fact that each time step of the RK4 algorithm is very expensive due to the need to iterate the projection at each of the four stages. We also observe that now the errors in the q and p variables share the same behavior, as was the case when integrating the DAE, but not in the other experiments based on integrating the stiff ODE.

For $\omega = 10000$ the errors in the algorithm are mainly due to the RK4 integrator and follow an $\mathcal{O}(H^4)$ pattern. In fact the slopes in the figure are slightly higher than that associated with an order four method due to the fact that halving H less than doubles the computational effort because the average number of iterations of the projection algorithm at each stage decreases slightly. For $H = 1/2, \ldots, 1/16$ the behavior of the method with $\omega = 1000$ and $\omega = 10000$ is essentially the same; work and accuracy are ω-independent. However, with $\omega = 1000$ and $H = 1/32$ and $H = 1/64$ the $\mathcal{O}(\omega^{-4})$ projection errors become important; when H is small, halving the step-length leads to an increase in error as more projections are performed. For $\omega = 100$ the computational cost for fixed H is slightly smaller than for $\omega = 1000$ or 10000, thus reflecting that we used a less demanding tolerance to stop the iteration.

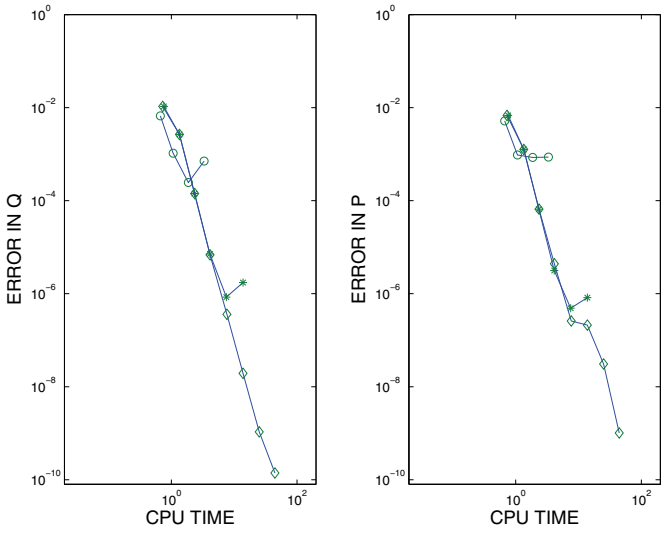

FIG. 6. *Errors for $\omega = 100$ (circles), $\omega = 1000$ (stars), $\omega = 10000$ (diamonds) when the slow solution is approximated by numerical integration with the classical RK4 algorithm with projection before each function evaluation.*

The RK4 integration provides a hard test for the accuracy of the iterative projection algorithm. For instance, at each time step, the point $Z^{2*} = z_n + (H/2)\phi(Z^1)$ to be projected is $\mathcal{O}(H^2)$ (uniformly in ω) away from the slow manifold; in view of the factor ω^2 in the oscillatory potential (3) this implies an oscillatory energy that grows unboundedly as ω increases. Several iterations (typically 3–5) are required to meet the stopping criterion (14) when projecting Z^{2*} to find Z^2. The analysis in section 4.3.3 predicts that those iterations introduce an $\mathcal{O}(H^4)$ tangential error in the projected Z^2; this error changes z_{n+1} by an $\mathcal{O}(H^5)$ amount, compatible with the overall algorithm being fourth-order accurate.

Comparing all of the results reported above, it appears that *for the problem at hand* and $\omega = 10000$ the DAE solver is the best option if high accuracy is not necessary; if it is, the projected RK4 is the most efficient choice.

In all the experiments in this section the aim has been the computation of the slow solution. However, as discussed in the theory in section 2 and borne out by the experiments above (see Figure 2), for ω suitably large there is a negligible difference between the slow and constrained solutions. This opens the possibility of obtaining approximations to solutions of the DAEs (2) by choosing a very large value of ω and finding the corresponding slow solution of the stiff system (1) with the IPA+RK4 technique described in this subsection. Note in this connection that, since the computational complexity of the IPA+RK4 numerical algorithm is *independent of* ω (cf. Figure 6), it is feasible to increase the stiffness in order that the slow solutions provide an extremely accurate approximation to the solutions of the DAEs. In those circumstances one may interpret the algorithm as a numerical method to solve the DAEs. Such a numerical method differs from standard DAE solvers in that these require at each step the solution of a system of algebraic equations and may need an amount of work that grows like a power of the dimension d (the details vary with the implicit solver and with the sparsity pattern of the problem).

7. Conclusions and future work. We have introduced and analyzed an algorithm that projects points in the phase space of a stiff mechanical system onto the associated slow manifold. The algorithm is based on ideas (micro-integration, filtering) from the field of multiscale simulation and is applicable to initializing integration algorithms for both stiff ODEs and DAEs, including the initialization of Lagrange multipliers.

We have shown that by applying the proposed iterated projection algorithm, conventional numerical integrators may enjoy the benefits of improved accuracy and reduced computational cost. Much future work is needed to identify the integrators that are best suited for this approach, to construct optimal filters, and to assess the merits of the new technique in realistic applications.

By performing the iterated projection before each function evaluation, it is possible to integrate DAEs with ODE solvers. While this approach—which does not require one to solve algebraic equations—is unlikely to be as efficient as standard DAE solvers for small problems, its application to large problems is worth considering.

Acknowledgment. The authors are thankful to M. P. Calvo for her assistance and to ICES, The University of Texas at Austin, for hosting a visit of Sanz-Serna that started this research.

REFERENCES

[1] G. Ariel, B. Engquist, and R. Tsai, *A multiscale method for highly oscillatory ordinary differential equations with resonance*, Math. Comp., 78 (2009), pp. 929–956.
[2] G. Ariel, B. Engquist, and R. Tsai, *Numerical multiscale methods for coupled oscillators*, Multiscale Model. Simul., 7 (2009), pp. 1387–1404.
[3] G. Ariel, B. Engquist, and R. Tsai, *A reversible multiscale integration method*, Commun. Math. Sci., 7 (2009), pp. 595–610.
[4] V. I. Arnold, *Mathematical Methods of Classical Mechanics*, 2nd ed., Springer, New York, 1989.
[5] F. Bornemann, *Homogenization in Time of Singularly Perturbed Mechanical Systems*, Lecture Notes in Math. 1687, Springer, Berlin, 1998.
[6] K. E. Brenan, S. L. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, North–Holland, New York, 1989.
[7] M. P. Calvo and J. M. Sanz-Serna, *Heterogeneous multiscale methods for mechanical systems with vibrations*, SIAM J. Sci. Comput., 32 (2010), pp. 2029–2046.
[8] Ph. Chartier, A. Murua, and J. M. Sanz-Serna, *Higher-order averaging, formal series and numerical integration* I: *B-series*, Found. Comput. Math., 10 (2010), pp. 695–727.
[9] W. E, *Analysis of the heterogeneous multiscale method for ordinary differential equations*, Commun. Math. Sci., 1 (2003), pp. 423–436.
[10] W. E and B. Engquist, *The heterogeneous multiscale methods*, Commun. Math. Sci., 1 (2003), pp. 87–132.
[11] W. E, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden, *Heterogeneous multiscale methods: A review*, Commun. Comput. Phys., 2 (2007), pp. 367–450.
[12] B. Engquist and Y. H. Tsai, *Heterogeneous multiscale methods for stiff ordinary differential equations*, Math. Comp., 74 (2005), pp. 1707–1742.
[13] C. W. Gear, T. J. Kaper, I. G. Kevrekidis, and A. Zagaris, *Projecting to a slow manifold: Singularly perturbed systems and legacy codes*, SIAM J. Appl. Dyn. Syst., 4 (2005), pp. 711–732.
[14] E. Hairer, Ch. Lubich, and G. Wanner, *Geometric Numerical Integration*, 2nd ed., Springer, Berlin, 2006.
[15] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations* II: *Stiff and Differential-Algebraic Problems*, 2nd ed., Springer, Berlin, 1996.
[16] H.-O. Kreiss, *Problems with different time scales*, Acta Numer., 1 (1992), pp. 101–139.
[17] H.-O. Kreiss and J. Lorenz, *Manifolds of slow solutions for highly oscillatory problems*, Indiana Univ. Math. J., 42 (1993), pp. 1169–1191.
[18] B. Leimkuhler and S. Reich, *Simulating Hamiltonian Dynamics*, Cambridge University Press, Cambridge, UK, 2005.

[19] Ch. Lubich, *Integration of stiff mechanical systems by Runge-Kutta methods*, Z. Angew. Math. Phys., 44 (1993), pp. 1022–1053.

[20] R. S. MacKay, *Slow manifolds*, in Energy Localization and Transfer, T. Dauxois, A. Litvak-Hinezon, R. S. MacKay, and A. Spanaudaki, eds., World Scientific, River Edge, NJ, 2004, pp. 149–192.

[21] J. M. Sanz-Serna, *Modulated Fourier expansions and heterogeneous multiscale methods*, IMA J. Numer. Anal., 29 (2009), pp. 595–605.

[22] T. Schlick, *Molecular Modelling and Simulation: An Interdisciplinary Guide*, 2nd ed., Springer, New York, 2010.

[23] A.-K. Tornberg and B. Engquist, *Numerical approximations of singular source terms in differential equations*, J. Comput. Phys., 200 (2004), pp. 462–488.

[24] A. Zagaris, C. W. Gear, T. J. Tasso, and Y. G. Kevrekidis, *Analysis of the accuracy and convergence of equation-free projection to a slow manifold*, M2AN Math. Model. Numer. Anal., 43 (2009), pp. 757–784.