A MINIMUM ACTION METHOD FOR DYNAMICAL SYSTEMS WITH CONSTANT TIME DELAYS

XIAOLIANG WAN* AND JIAYU ZHAI†

Abstract. In this work, we construct a minimum action method for dynamical systems with constant time delays. Minimum action method (MAM) plays an important role in seeking the most probable transition pathway induced by small noise. There exist two formulations of minimum action method: one is the geometric formulation based on the Maupertuis principle; and the other one is the temporal formulation. The geometric formulation relies on the conservation of Hamiltonian corresponding to the Freidlin-Wentzell action functional. For systems with time delays, the Hamiltonian does not conserve due to the explicit dependence on the time delay, which implies that the geometric MAM is not applicable. We work with the temporal formulation of MAM for problems with time delays. By defining an auxiliary path, we remove the optimization with respect to time through the optimal linear time scaling. The pointwise correspondence between the auxiliary path and the delayed transition path is dealt with by a penalty term included into the action functional. The action functional is then discretized by the finite element method, and strategies for h-adaptive mesh refinement have been developed. Numerical examples have been presented to demonstrate the effectiveness of our algorithm.

1. Introduction. As differential equations are used to model the dynamics in the real world, scientists and engineers want to make their models more realistic. Noting that the imperfect environment makes random perturbations ubiquitous in physical, chemical, biological and engineering applications, we may consider stochastic differential equations (SDEs) instead of deterministic ones by including random noise. One critical phenomenon beyond the deterministic models is the transition in the configuration space despite of the small noise amplitude. Such a transition may rarely occur but have extreme impact. Many important application problems can be considered as a small-noise-induced transition, e.g., non-equilibrium interface growth [7, 24], regime change in climate [35], switching in biophysical network [33], hydrodynamic instability [30, 31], wetting transitions on patterned surfaces [36] etc. Another way for model generalization is to include time delays into the system, which means that the dynamics may depend on not only the current state but also the past ones. A typical example is a mathematical model that regulates the self-driving vehicles [18]. Other applications include communication networks [3, 4], networked control systems [15, 34], traffic model and control [19], etc. We also note that model reduction results in low-order time-delay systems [13, 14]. In this paper, we seek numerically the most probable transition pathway induced by small noise in a time-delay system. This technique can be applied to study phase transitions in physical and biological applications. For example, experimental evidence of an absorbing phase transition was given recently for a bistable semiconductor laser with long delayed optoelectronic feedback and multiplicative noise [6].

To study the small-noise-induced transitions in dynamical systems, Freidlin and Wentzell introduced the large deviations theory for differential equations [8]. It gives a rigorous mathematical framework to quantify the probability of these rare events and to find the most possible transition path, which correspond to the minimum and the minimizer, respectively, of the so called Freidlin-Wentzell (F-W) action functional. Due to the lack of analytical solution, minimizing the F-W action functional numer-

*Department of Mathematics, Center for Computation and Technology, Louisiana State University, Baton Rouge 70803 (xlwan@math.lsu.edu)

†Department of Mathematics and Statistics, University of Massachusetts Amherst, Amherst, 01003 (zhai@math.umass.edu).

ically becomes critical from the application point of view. For a general dynamical system perturbed by small noise

$$d\boldsymbol{X}_t = \boldsymbol{b}(\boldsymbol{X}_t)\,dt + \sqrt{\varepsilon}\,d\boldsymbol{W}_t, \tag{1.1}$$

where ε is a small positive number and $\boldsymbol{W}_t$ is a standard Wiener process in $\mathbb{R}^n$, we have the following optimization problem:

$$S_{T^*}(\boldsymbol{\phi}_t^*) = \inf_{T\in\mathbb{R}^+} \inf_{\substack{\boldsymbol{\phi}_0=\boldsymbol{x}_1,\\ \boldsymbol{\phi}_T=\boldsymbol{x}_2}} S_T(\boldsymbol{\phi}_t), \tag{1.2}$$

where

$$S_T(\boldsymbol{\phi}_t) = \frac{1}{2}\int_0^T |\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t)|^2\,dt \tag{1.3}$$

is the F-W action functional and $\boldsymbol{\phi}_t^*$ defined on $[0,T^*]$ is the minimizer among all transition paths $\boldsymbol{\phi}_t$ connecting the two states $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ on the time interval $[0,T]$. The optimization problem (1.2) corresponds to the quasi-potential defined in equation (2.2). $\boldsymbol{\phi}_t^*$ is often called the minimal action path (MAP), and numerical algorithms that approximate $\boldsymbol{\phi}_t^*$ are in general called minimum action method (MAM) [5]. Available MAMs include: adaptive MAM (aMAM) [37, 26, 27, 25], geometric MAM (gMAM) [16, 9, 10], and MAM with optimal linear time scaling (tMAM) [28, 29, 32].

Consider the stochastic differential equation with a discrete time delay $0 < \tau < \infty$

$$\begin{cases} d\boldsymbol{X}_t = \boldsymbol{b}(\boldsymbol{X}_t, \boldsymbol{X}_{t-\tau})dt + \sqrt{\varepsilon}d\boldsymbol{W}_t, & t\in(0,T],\\ \boldsymbol{X}_t = \boldsymbol{\varphi}(t), & t\in[-\tau,0]. \end{cases} \tag{1.4}$$

Some results on large deviation of SDEs with constant time delays can be found in [1, 20, 21]. The F-W action functional for equation (1.4) is defined as

$$S_{\tau,T}(\boldsymbol{\phi}_t) = \frac{1}{2}\int_0^T \left|\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t, \boldsymbol{\phi}_{t-\tau})\right|^2 dt. \tag{1.5}$$

In this work, we focus on the optimization problem (1.2) with respect to the F-W action functional (1.5), i.e.,

$$S_{\tau,T^*}(\boldsymbol{\phi}_t^*) = \inf_{T\in\mathbb{R}^+} \inf_{\substack{\boldsymbol{\phi}_0=\boldsymbol{x}_1,\\ \boldsymbol{\phi}_T=\boldsymbol{x}_2}} S_{\tau,T}(\boldsymbol{\phi}_t), \tag{1.6}$$

It is not straightforward to generalize the available MAMs to deal with the time-delay systems. First of all, gMAM is not applicable. We note that gMAM is based on the Maupertuis principle, which means that a geodesic metric on the surface of constant Hamiltonian can be used to represent the action functional. However, the existence of an explicit time delay implies that the Hamiltonian is not conservative any more, meaning that the assumption of gMAM is not valid. We then need to work with time as the parameterization parameter. Both aMAM and tMAM can be employed. Considering that tMAM is more general than aMAM in the sense that aMAM is not able to deal with the case that T^* is finite, we focus on the generalization of tMAM in this paper.

In addition to the numerical difficulties for systems without time delays (see [32]), we need to pay attention to some extra difficulties induced by the time delay. First, the dynamical behavior of a time-delay system can be significantly different compared to a system without time delays. This implies that the initial guess of the optimization problem (1.6) should also depend on τ. Second, the change in the regularity of the solution of a time-delay system needs to be taken into account when we choose the approximation space and adaptivity strategy. Third, the optimal linear time scaling for time-delay systems is the root of a highly nonlinear equation, meaning that the uniqueness of the solution is not guaranteed such that a straightforward application of tMAM is not robust. Fourth, the time delay makes the problem nonlocal, meaning that the efficiency deserves some attention.

The main trick we use is the introduction of an auxiliary path $\boldsymbol{\psi}_t$ such that we can consider the minimization of

$$S_{\tau,T}(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t) = \frac{1}{2}\int_0^T \left|\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t)\right|^2 dt$$

subject to the constraint $\boldsymbol{\psi}_t = \boldsymbol{\phi}_{t-\tau}$. With respect to $\boldsymbol{\phi}_t$ and $\boldsymbol{\psi}_t$, the time delay does not show explicitly in the F-W action functional, meaning that the procedure of tMAM can be readily applied. To deal with the pointwise constraint $\boldsymbol{\psi}_t = \boldsymbol{\phi}_{t-\tau}$, we will include a penalty term in the action functional. Generally speaking, we decrease the complexity of the problem by increasing the dimensionality, where the number of unknowns is doubled. We then use finite elements to discretize the action functional, and an a posteriori error estimator based on the derivative recovery technique to guide the h-adaptivity. For now we do not look into the p-adaptivity because of the possible low regularity of the MAP, although such a low regularity might be local. Since the dynamical behavior may change significantly with respect to τ, we propose to increase the time delay from zero, where we assume a good initial guess is known for the minimization of $S_{\tau,T}$ with $\tau = 0$. Then the minimizer of $S_{\tau,T}$ will be used as the initial guess for the minimization of $S_{\tau+\delta\tau,T}$ such that the algorithm will be more robust. Furthermore, we will interweave the increment of τ and the mesh refinement to increase the efficiency.

The rest of this paper is organized as follows. We recall the tMAM in Section 2. The penalty method for the time-delay systems combined with some analysis is developed in Section 3. In Section 4, we provide a detailed discussion on the finite element discretization and the adaptivity strategy. Numerical results are given in Section 5 followed by a discussion section.

2. Minimum action method with optimal linear time scaling (tMAM).

We briefly recall the tMAM for dynamical systems perturbed by small noise [28]. Consider the following stochastic ODE:

$$d\boldsymbol{X}_t = \boldsymbol{b}(\boldsymbol{X}_t) + \sqrt{\varepsilon}d\boldsymbol{W}_t, \tag{2.1}$$

where ε is a small positive number, and $\boldsymbol{W}_t \in \mathbb{R}^n$ is a standard Wiener process. To address the most probable transition path from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$ induced by the small perturbations, we consider the quasi-potential:

$$V(\boldsymbol{x}_1, \boldsymbol{x}_2) = \inf_{T>0} \inf_{\substack{\boldsymbol{\phi}_0=\boldsymbol{x}_1,\\ \boldsymbol{\phi}_T=\boldsymbol{x}_2.}} \left[S_T = \frac{1}{2}\int_0^T |\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t)|^2 dt\right], \tag{2.2}$$

where S_T is called Freidlin-Wentzell action functional and the minimizer of S_T is called the minimal action pathway (MAP). According to the large deviation principle

(LDP), we know that

$$\Pr(\text{transition from } \boldsymbol{x}_1 \text{ to the vicinity of } \boldsymbol{x}_2) \approx Ce^{-\frac{V(\boldsymbol{x}_1,\boldsymbol{x}_2)}{\varepsilon}}, \tag{2.3}$$

when ε is small enough. The LDP also implies that the MAP is the most probable transition pathway (MPP), which is also called the maximum likelihood transition pathway (MLP). The tMAM was introduced in [28] to deal with the optimization problem in equation (2.2) required by the quasi-potential. The basic idea of tMAM is to remove the optimization parameter T by replacing it with an optimal linear time scaling

$$\hat{T}(\bar{\boldsymbol{\phi}}_s) = \frac{\|\bar{\boldsymbol{\phi}}_s'\|_{L^2(\Gamma_1)}}{\|b(\bar{\boldsymbol{\phi}}_s)\|_{L^2(\Gamma_1)}}. \tag{2.4}$$

where $\bar{\boldsymbol{\phi}}_s = \boldsymbol{\phi}_{t=sT}$, i.e., the time is mapped linearly from $\Gamma_T = [0, T]$ to $\Gamma_1 = [0, 1]$, and $'$ indicates the derivative with respect to the rescaled parameterization parameter s. The most straightforward way to obtain $\hat{T}(\bar{\boldsymbol{\phi}}_s)$ is to solve the following subproblem for any given $\bar{\boldsymbol{\phi}}_s$ with $s \in [0, 1]$

$$\hat{T}(\bar{\boldsymbol{\phi}}_s) = \arg\min_{T>0} \frac{T}{2} \int_0^1 |T^{-1}\bar{\boldsymbol{\phi}}_s - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s)|^2 ds,$$

which admits a unique solution given by $\hat{T}(\bar{\boldsymbol{\phi}}_s)$. Another way to obtain $\hat{T}$ is the zero-Hamiltonian constraint used in geometric minimum action method (gMAM). Taking the Legendre transform of the integrand of S_T with respect to $\dot{\boldsymbol{\phi}}_t$, we obtain the Hamiltonian

$$H(\boldsymbol{\phi}, \boldsymbol{p}) = \boldsymbol{b}^{\mathsf{T}}\boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^{\mathsf{T}}\boldsymbol{p}. \tag{2.5}$$

The conservation $H \equiv 0$ yields the following pointwise constraint on the transition path [16]

$$|\dot{\boldsymbol{\phi}}_t| = |\boldsymbol{b}(\boldsymbol{\phi}_t)|, \quad \forall t. \tag{2.6}$$

In terms of the variable s, the zero-Hamiltonian constraint becomes

$$|\bar{\boldsymbol{\phi}}_s'|T^{-1} = |\boldsymbol{b}(\bar{\boldsymbol{\phi}}_s)|, \quad \forall s \in [0, 1].$$

Integrating the above equation, we also obtain equation (2.4). The zero-Hamiltonian constraint (2.6) actually defines a nonlinear mapping between time and the geodesic metric on the surface $H \equiv 0$.

Replacing T in equation (2.2) with the optimal linear time scaling $\hat{T}(\bar{\boldsymbol{\phi}}_s)$, the optimization problem for the quasi-potential is reformulated as

$$\min_{\substack{\bar{\boldsymbol{\phi}}_0=\boldsymbol{x}_1,\\ \bar{\boldsymbol{\phi}}_1=\boldsymbol{x}_2.}} \left[S_{\hat{T}} = \frac{\hat{T}}{2} \int_0^1 |\hat{T}^{-1}\bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s)|^2 ds \right]. \tag{2.7}$$

If the optimal transition time is finite, this rescaled optimization problem is equivalent to the original one; If the optimal transition time is infinite, the rescaled optimization problem can still be used in the sense that the discrete version of the rescaled optimization problem is always well-posed. When there exists at least one critical point on the MAP, the optimal transition time is ∞. However, the discretization of the action functional can introduce a natural regularization such that the optimal transition time for the discrete action functional is always finite. Then an optimal linear time scaling always exists for the discrete action functional. The convergence analysis of a finite element discretization of $S_{\hat{T}}$ can be found in [32].

The main numerical difficulty for the optimization problem (2.7) can be explained by the Euler-Lagrange (E-L) equation associated with $S_{\hat{T}}$:

$$\hat{T}^{-2}(\bar{\boldsymbol{\phi}}_s)\bar{\boldsymbol{\phi}}_s'' + \hat{T}^{-1}(\bar{\boldsymbol{\phi}}_s)\left((\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T} - \nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b}\right)\bar{\boldsymbol{\phi}}_s' - (\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T}\boldsymbol{b} = 0. \tag{2.8}$$

When the optimal transition time is large, the E-L equation can be regarded as a singularly perturbed problem. In other words, the solution has boundary/internal layers, which means that adaptive discretization is necessary for numerical approximation. We have developed an *hp*-adaptive minimum action method based on a posteriori error estimate in [29] to approximate the optimization problem (2.7), where the optimal convergence rate of the finite element approximation has been recovered.

3. Penalty method for a dynamical system with time delays. We now consider the following stochastic ODE subject to a constant time delay

$$\begin{cases} d\boldsymbol{X}_t = \boldsymbol{b}(\boldsymbol{X}_t, \boldsymbol{X}_{t-\tau})dt + \sqrt{\varepsilon}d\boldsymbol{W}_t, & t \in (0, T], \\ \boldsymbol{X}_t = \boldsymbol{\varphi}(t), & t \in [-\tau, 0], \end{cases} \tag{3.1}$$

where $0 < \tau < \infty$ indicates the time delay. The solution of a time-delay system is not uniquely defined by the sole knowledge of the pointwise initial condition at $t = 0$ but by a functional initial condition $\boldsymbol{\varphi}(\cdot)$ defined over the interval $[-\tau, 0]$ [12]. In some literature, this is also referred to as a memory effect. Due to the dependence on a function instead of a point, equation (1.4) is not a finite-dimensional system, but an infinite-dimensional one. The Freidlin-Wentzell action functional for problem (1.4) is defined as [20, 21].

$$S_{\tau,T}(\boldsymbol{\phi}_t) = \frac{1}{2}\int_0^T \left|\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t, \boldsymbol{\phi}_{t-\tau})\right|^2 dt. \tag{3.2}$$

We intend to consider the following double-layered optimization problem

$$\inf_{T>0} \inf_{\substack{\boldsymbol{\phi}(0)=\boldsymbol{x}_1,\\ \boldsymbol{\phi}(T)=\boldsymbol{x}_2}} S_{\tau,T}(\boldsymbol{\phi}_t) \tag{3.3}$$

to seek the most probable transition in the sense of large deviation. Due to the explicit dependence on τ, the Hamiltonian will not be conservative, implying that the gMAM is not applicable for this problem.

We work with the temporal formulation of MAM. In particular, we intend to generalize the tMAM described in the previous section to deal with the optimization problem (3.3). Letting $t = sT$, we rewrite $S_{\tau,T}$ as

$$S_{\tau,T}(\boldsymbol{\phi}_t) = S_\tau(T, \bar{\boldsymbol{\phi}}_s) = \frac{T}{2}\int_0^1 |T^{-1}\bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\phi}}_{s-\tau/T})|^2 ds. \tag{3.4}$$

We will use $\langle \boldsymbol{v}, \boldsymbol{w}\rangle$ to indicate the inner product of vectors $\boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^n$, and $\langle \boldsymbol{g}_1(s), \boldsymbol{g}_2(s)\rangle_s$ to indicate the inner product of vector functions $\boldsymbol{g}_1(s), \boldsymbol{g}_2(s) \in \mathbb{R}^n$ defined for $s \in [0, 1]$. More specifically,

$$\langle \boldsymbol{g}_1(s), \boldsymbol{g}_2(s)\rangle_s = \int_0^1 \langle \boldsymbol{g}_1(s), \boldsymbol{g}_2(s)\rangle ds.$$

For any given $\boldsymbol{\phi}_s$, the optimization of $S_\tau(T, \bar{\boldsymbol{\phi}}_s)$ with respect to T, i.e., $\partial_T S_\tau(T, \bar{\boldsymbol{\phi}}_s) = 0$, yields that

$$\partial_T S_\tau(T, \bar{\boldsymbol{\phi}}_s) = \frac{1}{2}\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s - \frac{1}{2}T^{-2}\langle \bar{\boldsymbol{\phi}}_s', \bar{\boldsymbol{\phi}}_s'\rangle_s - \langle \widehat{\nabla}\boldsymbol{b}\bar{\boldsymbol{\phi}}_{\hat{s}}' T^{-1}\tau, T^{-1}\bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}\rangle_s = 0, \tag{3.5}$$

where we write $\bar{\boldsymbol{\phi}}_{s-\tau/T} = \bar{\boldsymbol{\phi}}_{\hat{s}}$ and let $\widehat{\nabla}\boldsymbol{b}$ indicate the gradient with respect to $\bar{\boldsymbol{\phi}}_{\hat{s}}$. It is seen that this is a nonlinear equation of T for any given $\bar{\boldsymbol{\phi}}_s$. In particular, the subscript $\hat{s}$ is a function of T. In contrast to the systems without time delays, the optimal linear time scaling given by equation (3.5) might not be unique. Although a root-finding algorithm is always possible, it is difficult to clarify the robustness of such a strategy.

To define a unique optimal linear time scaling for time-delay systems, we introduce an auxiliary path $\bar{\boldsymbol{\psi}}_s$, which is also defined on $[0,1]$ and satisfies the following pointwise constraint:

$$\bar{\boldsymbol{\psi}}_s = \bar{\boldsymbol{\phi}}_{\hat{s}} = \bar{\boldsymbol{\phi}}_{s-\tau/T}. \tag{3.6}$$

The action functional is rewritten as

$$S_\tau(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s) = \frac{T}{2}\int_0^1 |T^{-1}\bar{\boldsymbol{\phi}}_s' - b(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s)|^2 ds. \tag{3.7}$$

Assuming that $\bar{\boldsymbol{\phi}}_s$ and $\bar{\boldsymbol{\psi}}_s$ are independent, there exists a unique optimal linear time scaling satisfying $\partial_T S_\tau(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s) = 0$, i.e.,

$$\hat{T}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s) = \frac{\langle \bar{\boldsymbol{\phi}}_s', \bar{\boldsymbol{\phi}}_s'\rangle_s^{1/2}}{\langle \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s), \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s)\rangle_s^{1/2}} \tag{3.8}$$

for any given $\bar{\boldsymbol{\phi}}_s$ and $\bar{\boldsymbol{\psi}}_s$, which actually shares the same form as $\hat{T}$ defined in equation (2.4) for dynamical systems without time delays. To deal with the constraint (3.6), we add a penalty term into the action functional and define

$$\hat{S}_\tau(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s) = \frac{\hat{T}}{2}\int_0^1 |\hat{T}^{-1}\bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s)|^2 ds + \frac{\beta^2}{2}\int_0^1 |\bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}|^2 ds, \tag{3.9}$$

where $0 \neq \beta \in \mathbb{R}$ and $\bar{\boldsymbol{\phi}}_{\hat{s}} = \bar{\boldsymbol{\phi}}_{s-\tau/\hat{T}}$. Instead of minimizing the original action functional, we will work with its penalized form $\hat{S}_\tau(\boldsymbol{\phi}_s, \boldsymbol{\psi}_s)$. More specifically, we will consider the following optimization problem:

$$\min_{\substack{\bar{\boldsymbol{\phi}}_s \in H^1_{\Gamma_1}, \bar{\boldsymbol{\psi}}_s \in L^2_{\Gamma_1}, \\ \bar{\boldsymbol{\phi}}_0 = \boldsymbol{x}_1,\ \bar{\boldsymbol{\phi}}_1 = \boldsymbol{x}_2}} \hat{S}_\tau(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s). \tag{3.10}$$

3.1. Calculus of variation for $\hat{S}_\tau$.

For convenience, we split $\hat{S}_\tau$ to two parts:

$$\hat{S}_\tau(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s) = J^a(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\phi}}_s', \bar{\boldsymbol{\psi}}_s) + J^p(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\phi}}_{\hat{s}}, \bar{\boldsymbol{\psi}}_s),$$

corresponding to the two integrals respectively in equation (3.9), i.e.,

$$J^a = \frac{\hat{T}}{2}\int_0^1 |\hat{T}^{-1}\bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s)|^2 ds, \quad J^p = \frac{\beta^2}{2}\int_0^1 |\bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}|^2 ds,$$

where the dependence of J^p on $\bar{\boldsymbol{\phi}}_s$ is reflected through the relation $\hat{s} = s - \tau/\hat{T}$. Consider two test functions $\delta\bar{\boldsymbol{\phi}}_s \in H^1_{\Gamma_1}$ with $\delta\bar{\boldsymbol{\phi}}_s|_{s=0} = \delta\bar{\boldsymbol{\phi}}_s|_{s=1} = 0$, and $\delta\bar{\boldsymbol{\psi}}_s \in L^2_{\Gamma_1}$.

We first look at J^a. Note that we can treat $\hat{T}$ in J^a as a constant because $\partial_{\hat{T}} J^a = 0$ by the definition of $\hat{T}$. Then δJ^a can be easily obtained as

$$
\begin{aligned}
\left\langle \frac{\delta J^a}{\delta \bar{\boldsymbol{\phi}}'_s}, \delta\bar{\boldsymbol{\phi}}'_s \right\rangle_s &= \langle \hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}, \delta\bar{\boldsymbol{\phi}}'_s\rangle_s, \\
\left\langle \frac{\delta J^a}{\delta \bar{\boldsymbol{\phi}}_s}, \delta\bar{\boldsymbol{\phi}}_s \right\rangle_s &= -\hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T}(\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}), \delta\bar{\boldsymbol{\phi}}_s\rangle_s, \\
\left\langle \frac{\delta J^a}{\delta \bar{\boldsymbol{\psi}}_s}, \delta\bar{\boldsymbol{\psi}}_s \right\rangle_s &= -\hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_s}\boldsymbol{b})^\mathsf{T}(\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}), \delta\bar{\boldsymbol{\psi}}_s\rangle_s.
\end{aligned}
$$

We now look at J^p. In contrast to J^a, we need to take into account the contribution from the first-order variation of $\hat{T}$ for J^p, which is

$$
\partial_{\hat{T}} J^p \delta\hat{T} = -\frac{\tau\beta^2}{\hat{T}}\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \bar{\boldsymbol{\phi}}'_{\hat{s}}\rangle_s \delta\hat{T} = B\delta\hat{T},
$$

where

$$
B = -\frac{\tau\beta^2}{\hat{T}}\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \bar{\boldsymbol{\phi}}'_{\hat{s}}\rangle_s, \tag{3.11}
$$

and $\delta\hat{T}$ can be obtained from equation (3.8) as

$$
\delta\hat{T} = \frac{\langle \bar{\boldsymbol{\phi}}'_s, \delta\bar{\boldsymbol{\phi}}'_s\rangle_s}{\hat{T}\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} - \frac{\hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \delta\bar{\boldsymbol{\phi}}_s\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} - \frac{\hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_s}\boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \delta\bar{\boldsymbol{\psi}}_s\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s}. \tag{3.12}
$$

Fixing $\hat{T}$, we have

$$
\begin{aligned}
\left\langle \frac{\delta J^p}{\delta \bar{\boldsymbol{\psi}}_s}, \delta\bar{\boldsymbol{\psi}}_s \right\rangle_s \bigg|_{\hat{T}} &= \beta^2\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \delta\bar{\boldsymbol{\psi}}_s\rangle_s\big|_{\hat{T}}, \\
\left\langle \frac{\delta J^p}{\delta \bar{\boldsymbol{\phi}}_{\hat{s}}}, \delta\bar{\boldsymbol{\phi}}_{\hat{s}} \right\rangle_s \bigg|_{\hat{T}} &= -\beta^2\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \delta\bar{\boldsymbol{\phi}}_{\hat{s}}\rangle_s\big|_{\hat{T}}.
\end{aligned}
$$

Combining all the above information, we obtain the first-order variation of $\hat{S}_\tau$ as

$$
\begin{aligned}
&\delta\hat{S}_\tau(\delta\bar{\boldsymbol{\phi}}_s, \delta\bar{\boldsymbol{\psi}}_s) \\
=&\langle \hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}, \delta\boldsymbol{\phi}'_s\rangle_s + \frac{B}{\hat{T}\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s}\langle \bar{\boldsymbol{\phi}}'_s, \delta\bar{\boldsymbol{\phi}}'_s\rangle_s \\
&- \hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T}(\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}), \delta\bar{\boldsymbol{\phi}}_s\rangle_s - \frac{B\hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_s}\boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \delta\bar{\boldsymbol{\phi}}_s\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} \\
&- \hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_s}\boldsymbol{b})^\mathsf{T}(\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_s - \boldsymbol{b}), \delta\bar{\boldsymbol{\psi}}_s\rangle_s - \frac{B\hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_s}\boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \delta\bar{\boldsymbol{\psi}}_s\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} + \beta^2\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \delta\bar{\boldsymbol{\psi}}_s\rangle_s \\
&- \beta^2\langle \bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}, \delta\bar{\boldsymbol{\phi}}_{\hat{s}}\rangle_s.
\end{aligned} \tag{3.13}
$$

Choosing the test functions $\delta\bar{\boldsymbol{\phi}}_s$ and $\delta\bar{\boldsymbol{\psi}}_s$ from a finite element space, we will develop a numerical solver for problem (3.10) in section 4.

3.2. Change of variable. Although the constraint that the Hamiltonian is conservative does not hold for time-delay systems, a related constraint can be found through the change of variable. We look at the following formulation of the action functional

$$S_\tau(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t) = \frac{1}{2}\int_0^T |\dot{\boldsymbol{\phi}}_t - \boldsymbol{b}(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t)|^2 dt, \tag{3.14}$$

subject to the constraint $\boldsymbol{\psi}_t = \boldsymbol{\phi}_{t-\tau}$. Consider a change of variable $\alpha = \alpha(t)$. We have

$$\begin{aligned} S_\tau(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t) &= \frac{1}{2}\int_{\alpha(0)}^{\alpha(T)} |\boldsymbol{\phi}'_\alpha t'(\alpha)^{-1} - \boldsymbol{b}(\boldsymbol{\phi}_\alpha, \boldsymbol{\psi}_\alpha)|^2 t'(\alpha) d\alpha \\ &= \frac{1}{2}\int_{\alpha(0)}^{\alpha(T)} (|\boldsymbol{b}(\boldsymbol{\phi}_\alpha, \boldsymbol{\psi}_\alpha)|^2 t' + |\boldsymbol{\phi}'_\alpha|^2 (t')^{-1}) d\alpha - \int_{\alpha(0)}^{\alpha(T)} \langle \boldsymbol{b}, \boldsymbol{\phi}'_\alpha \rangle d\alpha \\ &\geq \int_{\alpha(0)}^{\alpha(T)} |\boldsymbol{b}(\boldsymbol{\phi}_\alpha, \boldsymbol{\psi}_\alpha)||\boldsymbol{\phi}'_\alpha| d\alpha - \int_{\alpha(0)}^{\alpha(T)} \langle \boldsymbol{b}, \boldsymbol{\phi}'_\alpha \rangle d\alpha, \end{aligned} \tag{3.15}$$

where $'$ indicates the derivative with respect to α. To achieve the lower bound of S_τ, the equality in the last step will hold when

$$|\boldsymbol{\phi}'_\alpha| = t'(\alpha)|\boldsymbol{b}(\boldsymbol{\phi}_\alpha, \boldsymbol{\psi}_\alpha)|, \quad \forall \alpha \tag{3.16}$$

or

$$|\dot{\boldsymbol{\phi}}| = |\boldsymbol{b}(\boldsymbol{\phi}_t, \boldsymbol{\psi}_t)|, \quad \forall t. \tag{3.17}$$

Taking into account the constraint $\boldsymbol{\psi}_t = \boldsymbol{\phi}_{t-\tau}$, the function $t(\alpha)$ is given by the following differential equation

$$\frac{dt}{d\alpha} = \frac{|\boldsymbol{\phi}'_{\alpha(t)}|}{|\boldsymbol{b}(\boldsymbol{\phi}_{\alpha(t)}, \boldsymbol{\phi}_{\alpha(t-\tau)})|}. \tag{3.18}$$

Without loss of generality, we assume that α indicates the arc length. Starting from $\boldsymbol{\phi}_\alpha$ there exist many different ways to define $t(\alpha)$, since a particle can travel along the curve at a varying speed. However, the condition (3.17) yields a particular way to parameterize the path with respect to time such that the action functional can reach its lower bound in equation (3.15). Let

$$\hat{\alpha}(t) = \int_t^0 |\dot{\boldsymbol{\varphi}}_t| dt, \quad \forall t \in [-\tau, 0]. \tag{3.19}$$

The initial condition of equation (3.18) can be defined as

$$t = \hat{\alpha}^{-1}(\alpha).$$

for $\alpha \in [-\int_{-\tau}^0 |\dot{\boldsymbol{\varphi}}| dt, 0]$. Note that for any $\alpha_1 > \alpha_2$, $t_1 = \hat{\alpha}^{-1}(\alpha_1) \geq \hat{\alpha}^{-1}(\alpha_2) = t_2$, which implies that

$$\alpha(t-\tau) < \alpha(t).$$

Thus in terms of α, equation (3.18) is a delayed differential equation when $\tau > 0$. The delay given by $\alpha(t) - \alpha(t-\tau)$ is time dependent although τ is a constant.

The constraint (3.17) is a necessary condition satisfied by the minimizer of the action functional, which specifies the relation between the time and a more effective parameterization for the MAP. With respect to α, there also exist infinitely many curves connecting $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, among which the MAP will be sought. However, due to the existence of time delay τ, we are not able to obtain a closed formulation of the action functional with respect to α. This is the main reason that gMAM is not applicable.

4. Finite element approximation. For dynamical systems without time delays, we provided in [32] a finite element approximation framework for the discretization of the action functional, where the well-posedness of optimizing S_T and the convergence of the linear finite element approximation of the MAP have been analyzed. We also showed in [29] that the tMAM based on the adaptive finite element approximation is able to recover the optimal convergence rate for both h-refinement and hp-refinement (see Section 4.3) no matter that the optimal transition time is finite or infinite. In this work, we will use finite elements to discretize $\hat{S}_\tau$, where we pay particular attention to the effectiveness of the penalty method that deals with the time delays.

4.1. Approximation spaces. Consider a partition of the interval $\Gamma_1 = [0, 1]$:

$$\mathcal{T}_h : 0 = s_0 < s_1 < \cdots < s_N = 1.$$

Let $R = [-1, 1]$ be a reference element and F_{e_i} an affine mapping from the element $e_i = [s_i, s_{i+1}]$, $i = 0, 1, \ldots, N-1$, to the reference element R. Then in each element e_i, we can define a linear space spanned by polynomials

$$W^p_{e_i} = \{v : v \circ F^{-1}_{e_i} \in \mathscr{P}_p(R)\}, \tag{4.1}$$

where $\mathscr{P}_p(R)$ denotes the set of polynomials of degree up to p over R. In particular, we choose $\mathscr{P}_p(R) = \mathrm{span}\{\tilde{\theta}_i(\tilde{s})\}_{i=0}^m$, where

$$\tilde{\theta}_i(\tilde{s}) = \begin{cases} \frac{1-\tilde{s}}{2}, & i = 0, \\ \frac{1+\tilde{s}}{2}, & i = 1, \\ \frac{1-\tilde{s}}{2}\frac{1+\tilde{s}}{2} P^{1,1}_{i-2}(\tilde{s}), & 2 \le i \le m, \end{cases} \tag{4.2}$$

where $P^{1,1}_i(\tilde{s})$ denotes orthogonal Jacobi polynomials of degree i with respect to the weight function $(1-\tilde{s})(1+\tilde{s})$ [17]. The polynomial order of $\tilde{\theta}_i$ is equal to i for $i \ge 2$. Let us call $\tilde{\theta}_0$ the left boundary mode and $\tilde{\theta}_1$ the right boundary mode. All interior modes with $i \ge 2$ are equal to zero at the element boundaries.

With the partition $\mathcal{T}_h$, we define the following finite element approximation space for $\bar{\boldsymbol{\phi}}_s$:

$$W^p_h = \left\{\boldsymbol{v} : \boldsymbol{v} \in \mathbb{R}^n,\, v_i \in H^1(\Gamma_1),\, v_i|_{e_j} \in W^{(p)}_{e_j},\, \boldsymbol{v}(0) = \boldsymbol{x}_1,\, \boldsymbol{v}(1) = \boldsymbol{x}_2\right\} \subset H^1(\Gamma_1; \mathbb{R}^n),$$

where $i = 1, \ldots, n$, and $j = 0, \ldots, N-1$. For $\bar{\boldsymbol{\psi}}_s$, we use the same approximate space by removing the constraints at the starting and ending points:

$$V^p_h = \left\{\boldsymbol{v} : \boldsymbol{v} \in \mathbb{R}^n,\, v_i \in L^2(\Gamma_1),\, v_i|_{e_j} \in W^{(p)}_{e_j}\right\} \subset L^2(\Gamma_1; \mathbb{R}^n).$$

We then discretize problem (3.10) as

$$\min_{(\bar{\boldsymbol{\phi}}_{h,s}, \bar{\boldsymbol{\psi}}_{h,s}) \in W^p_h \otimes V^p_h} \hat{S}_{\tau,h} = \hat{S}_\tau(\bar{\boldsymbol{\phi}}_{h,s}, \bar{\boldsymbol{\psi}}_{h,s}). \tag{4.3}$$

Let us order the finite element basis functions defined on $\mathcal{T}_h$ from 0 to $M+1$, and let $\bar{\boldsymbol{\phi}}_{h,s}$ and $\bar{\boldsymbol{\psi}}_{h,s}$ have the following representations in W_h^p and V_h^p, respectively,

$$\bar{\boldsymbol{\phi}}_{h,s} = \sum_{i=1}^{M} \boldsymbol{\phi}_i \theta_{W,i,s} + \boldsymbol{x}_1 \theta_{W,0,s} + \boldsymbol{x}_2 \theta_{W,M+1,s}, \quad \bar{\boldsymbol{\psi}}_{h,s} = \sum_{i=0}^{M+1} \boldsymbol{\psi}_i \theta_{V,i,s},$$

where $\theta_{W,0,s}$ and $\theta_{W,M+1,s}$ for $\bar{\boldsymbol{\phi}}_{h,s}$ indicate the left boundary mode in element e_0 and the right boundary mode in element e_{N-1}, respectively. Although we use the same finite element basis functions to define W_h^p and V_h^p, we still differentiate them for clarity by adding subscripts $\cdot_W$ and $\cdot_V$. The first-order variation of $\hat{S}_\tau$ given in equation (3.13) gives the gradient of the discrete action functional $\hat{S}_{\tau,h}$. More specifically,

$$\frac{\partial \hat{S}_{\tau,h}}{\partial \phi_{i,j}} = \delta \hat{S}_\tau(\theta_{W,i,s}\mathbf{e}_j, \mathbf{0}), \quad \frac{\partial \hat{S}_{\tau,h}}{\partial \psi_{i,j}} = \delta \hat{S}_\tau(\mathbf{0}, \theta_{V,i,s}\mathbf{e}_j), \tag{4.4}$$

where $\delta \hat{S}_\tau$ is given in equation (3.13), and $\mathbf{e}_j$ is the unit vector in $\mathbb{R}^n$ with its jth component being 1 and the rest being 0. To this end, we obtain an unconstrained optimization problem, for which a gradient-type optimization algorithm such as L-BFGS, nonlinear conjugate gradient method, etc., can be employed to seek the approximate MAP.

Remark 4.1. *One popular strategy to reduce the possibility of ill conditioning induced by the penalty term in equation* (3.9) *is the augmented Lagrangian method, which introduces explicit Lagrange multiplier estimates for the constraint [22]. In this work, we do not employ the augmented Lagrangian method not only for simplicity but also due to the observation that the pointwise constraint* (3.6) *cannot be achieved exactly in the finite element space $W_h^p \otimes V_h^p$, where the same mesh is used for both W_h^p and V_h^p.*

Remark 4.2. *We include more details about equation* (4.4)*:*

$$\begin{aligned}\frac{\partial \hat{S}_{\tau,h}}{\partial \phi_{i,j}} =& \langle \hat{T}^{-1} \bar{\boldsymbol{\phi}}'_{h,s} - \boldsymbol{b}, \theta'_{W,i,s}\mathbf{e}_j \rangle_s - \frac{B}{\hat{T}\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} \langle \boldsymbol{\phi}'_{h,s}, \theta'_{W,i,s}\mathbf{e}_j\rangle_s \\ & - \hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_{h,s}} \boldsymbol{b})^\mathsf{T} (\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_{h,s} - \boldsymbol{b}), \theta_{W,i,s}\mathbf{e}_j\rangle_s - \frac{B\hat{T}\langle (\nabla_{\bar{\boldsymbol{\phi}}_{h,s}} \boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \theta_{W,i,s}\mathbf{e}_j\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} \\ & - \beta^2 \langle \bar{\boldsymbol{\psi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,\hat{s}}, \theta_{W,i,\hat{s}}\mathbf{e}_j\rangle_s, \quad i = 1, \dots, M, \end{aligned} \tag{4.5}$$

and

$$\begin{aligned}\frac{\partial \hat{S}_{\tau,h}}{\partial \psi_{i,j}} =& - \hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_{h,s}} \boldsymbol{b})^\mathsf{T} (\hat{T}^{-1}\bar{\boldsymbol{\phi}}'_{h,s} - \boldsymbol{b}), \theta_{V,i,s}\mathbf{e}_j\rangle_s - \frac{B\hat{T}\langle (\nabla_{\bar{\boldsymbol{\psi}}_{h,s}} \boldsymbol{b})^\mathsf{T}\boldsymbol{b}, \theta_{V,i,s}\mathbf{e}_j\rangle_s}{\langle \boldsymbol{b}, \boldsymbol{b}\rangle_s} \\ & + \beta^2 \langle \bar{\boldsymbol{\psi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,\hat{s}}, \theta_{V,i,s}\mathbf{e}_j\rangle_s, \quad i = 0, \dots, M+1, \end{aligned} \tag{4.6}$$

where B is given in equation (3.11).

Remark 4.3. *From the optimization point of view, we should increase the value of the penalty parameter gradually to achieve a better approximation. Since the pointwise constraint* (3.6) *cannot be exactly satisfied in the approximation space, the penalty parameter cannot be too large, otherwise, the action term may be overwhelmed by the penalty term. In other words, a lower bound of the penalty parameter is expected to achieve the convergence of the numerical solution. This problem will be left for future* *study. In this work we simply increase the penalty parameter to examine the possible improvement. See remark 4.6 and more discussions about adaptivity in section 4.4.2.*

4.2. The computation of gradient. The time delay introduces some complexities for the computation of the gradient $\nabla\hat{S}_{\tau,h}$. It is seen in equations (4.5) and (4.6) that there exist some terms, such as $\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s$, that may not be achieved within one element due to the existence of time delay no matter that the mesh is uniform or not. Among all the inner products needed for the gradient, we only look at two cases that are related to time delay: 1) The time delay is in the transition path $\bar{\boldsymbol{\phi}}_{h,s}$ or $\bar{\boldsymbol{\psi}}_{h,s}$, e.g., $\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s$, and 2) the time delay is in the basis functions, e.g., $\langle\bar{\boldsymbol{\psi}}_{h,s},\theta_{W,i,\hat{s}}\mathbf{e}_j\rangle_s$. For these two cases, information from different regions is requested for integration. These two cases are illustrated by figures 4.1 and 4.2, where we use two identical horizontal lines to indicate the mesh shared by the transition path $\bar{\boldsymbol{\phi}}_{h,s}$ and the basis function $\theta_{V,i,s}$.

Let us first assume that the delay exists in the transition path and consider $\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s$. The basis function $\theta_{V,i,s}$ is defined on a certain element, say e_k. For integration, we need the information of the path on $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$. First of all, $\hat{T}$ depends on $\bar{\boldsymbol{\phi}}_{h,s}$ and $\bar{\boldsymbol{\psi}}_{h,s}$, meaning that interval $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$ varies at each optimization iteration. Second, the boundaries of the interval $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$ are, in general, not grid points, see the illustration in figure 4.1. To achieve the integration, we need to know how the interval $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$ overlaps with previous elements. For the scenario in figure 4.1, it is seen that the interval has overlap with three elements. In contrast to the case without time delays, the inner product involves three elements instead of one. This means on each element, we may need to compute the Gauss-type quadrature points for a subinterval, one of whose boundaries is an interior point of this element. These information cannot be pre-computed on the reference element. In reality, we can maintain a list for each element $[s_k,s_{k+1}]$, which contains all the elements that have overlap with $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$, and will be updated for each optimization iteration after $\hat{T}$ is updated. For example, suppose that $\theta_{V,i,s}$ is located in element e_k. Let `s_backward[i][j]` be a two-dimensional array, where $i=0,\dots,N-1$ indicates the element index and j indicates the elements that have overlap with $[s_i-\tau/\hat{T},s_{i+1}-\tau/\hat{T}]$. In figure 4.1, we have `s_backward[k][j]=k-1-j`, $j=0,1,2$, such that

$$\begin{aligned}\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s &= \langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s\big|_{\hat{s}\in e_{k-1}}+\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s\big|_{\hat{s}\in e_{k-2}}\\ &\quad+\langle\bar{\boldsymbol{\phi}}_{h,\hat{s}},\theta_{V,i,s}\mathbf{e}_j\rangle_s\big|_{\hat{s}\in e_{k-3}}.\end{aligned}$$

We now assume that delay exists in the basis function, illustrated by figure 4.2, and consider the inner product $\langle\bar{\boldsymbol{\phi}}_{h,s},\theta_{V,i,\hat{s}}\mathbf{e}_j\rangle_s$. We let $\theta_{V,i,s}$ be a bubble function, i.e., nonzero on one element and zero elsewhere. It is seen that although $\theta_{V,i,s}=0$ on the element $[s_k,s_{k+1}]$ in which we take information of $\bar{\boldsymbol{\phi}}_{h,s}$, the inner product is not zero due to the time delay, i.e., $\theta_{V,i,\hat{s}}\neq 0$ on $[s_k-\tau/\hat{T},s_{k+1}-\tau/\hat{T}]$. In particular, due to the compact support of $\theta_{V,i,s}$, the valid part for integration, given by the thicker line in figure 4.2, is the only part of the element on which the non-zero part of $\theta_{V,i,s}$ is defined. If we use $\theta_{V,i,s}$ as a reference instead of $\bar{\boldsymbol{\phi}}_{h,s}$, we need to know that what elements have overlap with $[s_k+\hat{T},s_{k+1}+\hat{T}]$. Similar to the previous case, we can maintain a list for each element and update it as soon as $\hat{T}$ is updated. We still assume that $\theta_{V,i,s}$ is located in element e_k. This time we define a two-dimensional array `s_forward[i][j]` where i indicates the element index while j indicates the elements that have overlap

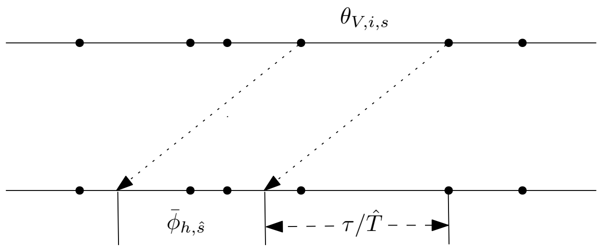

FIG. 4.1. *The inner product of a basis function and a delayed path.*

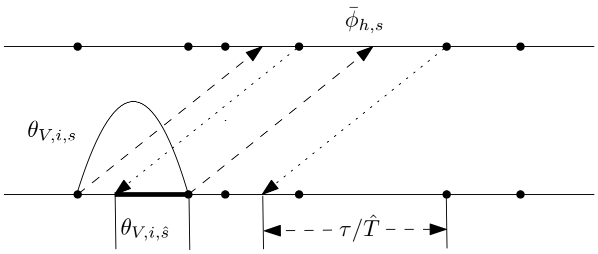

FIG. 4.2. *The inner product of a delayed basis function and the path.*

with $[s_i + \tau/\hat{T}, s_{i+1} + \tau/\hat{T}]$. In figure 4.2, we have `s_forward[k][j]=k+2+j`, $j = 0, 1$. Thus

$$\langle \bar{\boldsymbol{\phi}}_{h,s}, \theta_{V,i,\hat{s}}\mathbf{e}_j\rangle_s = \langle \bar{\boldsymbol{\phi}}_{h,s}, \theta_{V,i,\hat{s}}\mathbf{e}_j\rangle_s\big|_{s\in e_{k+2}} + \langle \bar{\boldsymbol{\phi}}_{h,s}, \theta_{V,i,\hat{s}}\mathbf{e}_j\rangle_s\big|_{s\in e_{k+3}}.$$

REMARK 4.4. *It is seen that due to the variation and the time delay, the computation of the gradient is much more complicated than the cases without time delays [28]. On the other hand, we note that the finite element basis $\theta_{V,i,s}$ is much more flexible to deal with the time delay than other types of discretization, such as the finite difference method, in the sense that the basis function itself is able to carry the effect of time delay.*

4.3. Mesh refinement. Mesh refinement is an important issue for MAM formulated with respect to time. Due to the existence of both slow and fast dynamics, the non-uniform mesh is a necessity for an accurate approximation. Simply speaking, the mesh for the transition path $\bar{\boldsymbol{\phi}}_s$ should be consistent with the dynamics [28]. In the region of slow dynamics, the element size can be larger while in the region of fast dynamics, the element size should be small. For problems without time delays, this physically-based adaptivity criterion was further refined by a regularity-consistent a posteriori error estimator in [29].

In our penalized action functional for time-delay systems, we define an auxiliary path $\bar{\boldsymbol{\psi}}_s = \bar{\boldsymbol{\phi}}_{\hat{s}=s-\tau/\hat{T}}$. From the approximation point of view, the mesh for $\bar{\boldsymbol{\phi}}_s$ on $[0, 1-\tau/\hat{T}]$ should be comparable to the mesh of $\bar{\boldsymbol{\psi}}_s$ on $[\tau/\hat{T}, 1]$. If one non-uniform mesh is used for both $\bar{\boldsymbol{\phi}}_s$ and $\bar{\boldsymbol{\psi}}_s$, it is difficult to achieve such a translation invariance. The simplest solution is to use different meshes for $\bar{\boldsymbol{\phi}}_s$ and $\bar{\boldsymbol{\psi}}_s$, which certainly introduces more computation cost. We note that if $\tau \ll \hat{T}$, it may still be reasonable to use the same mesh for both $\bar{\boldsymbol{\phi}}_s$ and $\bar{\boldsymbol{\psi}}_s$.

4.3.1. A posteriori error estimator. We have two choices here for mesh refinement: 1) $\bar{\boldsymbol{\phi}}_{h,s}$ and $\bar{\boldsymbol{\psi}}_{h,s}$ use the same mesh, and 2) $\bar{\boldsymbol{\phi}}_{h,s}$ and $\bar{\boldsymbol{\psi}}_{h,s}$ use different meshes. For both choices, we can use the derivative-recovery technique, which was developed in [29] for tMAM and dynamical systems without time delays, to obtain a posteriori error estimate for $\bar{\boldsymbol{\phi}}_{h,s}$. The reason we can achieve this is that the a posteriori error estimator in [29] only depends on the regularity of the path, in other words, it does not depend explicitly on the problem itself. For the first choice, we can construct an element-wise error indicator as follows. Suppose that we have an estimated solution $\hat{\boldsymbol{\phi}}_{h,s}$ given by the derivative-recovery technique, which is more accurate than $\bar{\boldsymbol{\phi}}_{h,s}$ in a certain sense. We define an error estimator η_{e_k} on element $e_k = [s_k, s_{k+1}]$:

$$\eta_{1,e_k} = |\hat{\boldsymbol{\phi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s}|_{H^1(D)}\Big|_{e_k} + |\bar{\boldsymbol{\psi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s-\tau/\hat{T}}|_{L^2(D)}\Big|_{e_k}, \tag{4.7}$$

where the first term in η_{e_k} is the estimated error of $\bar{\boldsymbol{\phi}}_{h,s}$ on e_k, and the second term measures the deviation from the point-wise constraint (3.6). For the second choice, we first update the mesh for $\bar{\boldsymbol{\phi}}_{h,s}$ using the error indicator

$$\eta_{2,e_k} = |\hat{\boldsymbol{\phi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s}|_{H^1(D)}\Big|_{e_k}, \tag{4.8}$$

where we only keep the first term in η_{1,e_k}, and then generate the mesh for $\bar{\boldsymbol{\psi}}_{h,s}(s)$ according to the constraint $\bar{\boldsymbol{\psi}}_s = \bar{\boldsymbol{\phi}}_{\hat{s}}$. More specifically, we can use the mesh of $[0, 1-\tau/\hat{T}]$ for $\bar{\boldsymbol{\phi}}_{h,s}$ as the mesh of $[\tau/\hat{T}, 1]$ for $\bar{\boldsymbol{\psi}}_{h,s}(s)$. The mesh of $[0, \tau/\hat{T}]$ for $\bar{\boldsymbol{\psi}}_{h,s}$ can be easily generated according to the initial condition. Compared to the first choice, the second choice is more expensive since a global operation is needed to project $\bar{\boldsymbol{\psi}}_{h,s}$ from the old mesh to the new one. In this work, we will only consider the first choice, where only local projection is needed after the mesh is refined.

We now outline the computation of η_{1,e_k}, and more details can be found in [29]. For robustness, we only consider h-refinement, meaning that we split one element to two equal elements if it is associated with a relatively large error estimate η_{1,e_k}. Assume that $\bar{\boldsymbol{\phi}}_{h,s} \in W_h^p$. Then the pth order derivative $\bar{\boldsymbol{\phi}}_{h,s}^{(p)} \in \mathbb{R}^n$ is a piecewise constant vector. The derivative recovery with respect to $\bar{\boldsymbol{\phi}}_{h,s}^{(p)}$ consists of two steps. The first step is a projection step, where we define a projection operator $\mathcal{Q}_h$ such that

$$\langle \mathcal{Q}_h \bar{\boldsymbol{\phi}}_{h,s}^{(p)}, \bar{\boldsymbol{\varphi}}_{h,s} \rangle_s = \langle \bar{\boldsymbol{\phi}}_{h,s}^{(p)}, \bar{\boldsymbol{\varphi}}_{h,s} \rangle_s, \quad \forall \boldsymbol{\varphi}_{h,s} \in W_h^1. \tag{4.9}$$

In other words, we project a piecewise constant function onto the linear finite element space for each component of $\bar{\boldsymbol{\phi}}_{h,s}^{(p)}$. The second step is a smoothing step using the operator $\mathcal{S}_h = \mathcal{I} - \lambda^{-1}\mathcal{A}_h$, where $\mathcal{I}$ is an identity operator, $\mathcal{A}_h : W_h^1 \to W_h^1$ is uniquely determined by

$$\langle \mathcal{A}_h \bar{\boldsymbol{\varphi}}_{h,s}, \bar{\boldsymbol{\xi}}_{h,s} \rangle_s = \langle \bar{\boldsymbol{\varphi}}_{h,s}', \bar{\boldsymbol{\xi}}_{h,s}' \rangle_s + \langle \bar{\boldsymbol{\varphi}}_{h,s}, \bar{\boldsymbol{\xi}}_{h,s} \rangle_s, \quad \forall \bar{\boldsymbol{\varphi}}_{h,s}, \bar{\boldsymbol{\xi}}_{h,s} \in W_h^1, \tag{4.10}$$

and $\lambda = \rho(\mathcal{A}_h) \simeq h^{-2}$ with h being the element size. We then have the recovered pth-order derivative $\mathcal{R}\bar{\boldsymbol{\phi}}_{h,s}^{(p)} = \mathcal{S}_h^m \mathcal{Q}_h \bar{\boldsymbol{\phi}}_{h,s}^{(p)}$, where m is the number of smoothing steps. Roughly speaking, we will use $\mathcal{R}\bar{\boldsymbol{\phi}}_{h,s}^{(p)}$ to replace the p-th order derivative $\bar{\boldsymbol{\phi}}_s^{*(p)}$ of the exact or reference solution $\bar{\boldsymbol{\phi}}_s^*$.

We now use $\mathcal{R}\bar{\boldsymbol{\phi}}_{h,s}^{(p)}$ to construct a piecewise polynomial $\tilde{\bar{\boldsymbol{\phi}}}_{h,s}$ of degree $p+1$ such that

$$\tilde{\bar{\boldsymbol{\phi}}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s}\Big|_{e_k} = \mathrm{diag}(\boldsymbol{c})(\mathcal{I} - \mathcal{P}_p)\bar{\boldsymbol{\varphi}}_{h,s}^{e_k,p+1}, \tag{4.11}$$

where $\boldsymbol{c} \in \mathbb{R}^n$, $\bar{\boldsymbol{\varphi}}_{h,s}^{e_k,p+1} = \tilde{\theta}_{p+1} \circ F_{e_k}^{-1}(s)[1,1,\ldots,1]^{\mathsf{T}} \in \mathbb{R}^n$, and $\tilde{\theta}_{p+1} \circ F_{e_k}^{-1}(s)$ is the local polynomial basis of degree $p+1$ defined on element e_k, and $\mathcal{P}_p$ indicates a L_2 projection operator onto the space span$\{\tilde{\theta}_{p+1} \circ F_{e_k}^{-1}(s)\}_{i=0}^{p}$ since the local basis functions are not mutually orthogonal. We then use the approximation $\tilde{\bar{\boldsymbol{\phi}}}_{h,s}^{(p+1)} \approx (\mathcal{R}\bar{\boldsymbol{\phi}}_{h,s}^{(p)})'$ to determine the coefficient vector $\boldsymbol{c}$. To this end, we can define the error indicator

$$\eta_{1,e_k}^2 = \alpha_{e_k}^2 |\tilde{\bar{\boldsymbol{\phi}}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s}|_{H^1(D)}^2\Big|_{e_k} + \beta^2 |\bar{\boldsymbol{\psi}}_{h,s} - \bar{\boldsymbol{\phi}}_{h,s-\tau/\hat{T}}|_{L^2(D)}^2\Big|_{e_k}, \tag{4.12}$$

where the coefficient α_{e_k} satisfies

$$\alpha_{e_k} = \frac{\|(\mathcal{I} - \mathcal{R})\bar{\boldsymbol{\phi}}_{h,s}^{(p)}\|_{L^2(D)}\Big|_{e_k}}{\|\tilde{\bar{\boldsymbol{\phi}}}_{h,s}^{(p)} - \bar{\boldsymbol{\phi}}_{h,s}^{(p)}\|_{L^2(D)}\Big|_{e_k}}.$$

The total error is defined as

$$\eta_1 = \left(\sum_{k=0}^{N-1} \eta_{e_k}^2\right)^{1/2}. \tag{4.13}$$

Let $J = \{i | 0 \le i \le N-1\}$ be the set of indices of all finite elements. We look for a subset $\hat{J} \subset J$ such that for $r_\eta \in (0,1]$,

$$r_\eta \sum_{i \in J} \eta_{e_i}^2 \le \sum_{i \in \hat{J}} \eta_{e_i}^2. \tag{4.14}$$

To uniquely specify $\hat{J}$, we choose the elements that have the largest estimated error, i.e.,

$$\min_{i \in \hat{J}} \eta_{e_i} \ge \max_{i \in J \setminus \hat{J}} \eta_{e_i}.$$

This is sometimes referred to as Dörfler's marking strategy. Then all elements whose indices belong to $\hat{J}$ will be refined to two equidistant elements, i.e., h-refinement. Let M_{old} be the number of degrees of freedom (DOFs) of the old mesh, and M_D the number of DOFs after h-refinement based on the Döfler's marking strategy.

4.3.2. Maintaining constraint (3.17)**.** The constraint (3.17) is a necessary condition satisfied by the MAP. To measure the deviation from this constraint, we define the following elementwise indicator as in [29]:

$$\begin{aligned}\theta_{e_i}^2 &= \int_{\hat{T}s_i}^{\hat{T}s_{i+1}} (|\dot{\boldsymbol{\phi}}_{h,t}| - |b|)^2 dt \\ &= \hat{T}\int_{s_i}^{s_{i+1}} (\hat{T}^{-1}|\bar{\boldsymbol{\phi}}_{h,s}'| - |b|)^2 ds, \quad i = 0, 1, \ldots, N-1.\end{aligned} \tag{4.15}$$

Let $\theta_{\max}$ and $\theta_{\min}$ be the maximum and minimum values of θ_{e_i} respectively. If the raito $\theta_{\max}/\theta_{\min}$ is larger than a threshold θ_c, we will implement h-refinement in elements with large θ_i such that the deviation from the constraint (3.17) is not too skewed. More specifically, we will refine the element with the largest θ_{e_i} until $(M - M_D) \ge r_M(M_D - M_{\mathsf{old}})$. In other words, after refining the mesh according to η_{e_i}, we add $r_M(M_D - M_{\mathsf{old}})$ more DOFs by refining the mesh according to θ_{e_i}. We usually choose $r_M = 10\%$ [29].

To this end, we can define an h-adaptive tMAM for time-delay systems, see Algorithm 1.

Algorithm 1 h-adaptive tMAM for time-delay systems.

Solve problem (3.10) to obtain $\bar{\boldsymbol{\phi}}_{h,s}^{*,0}$ and $\bar{\boldsymbol{\psi}}_{h,s}^{*,0}$ on the initial partition $\mathcal{T}_h^0$.
while $\epsilon > \epsilon_{\text{tol}}$ **do**
  Compute η_{e_i}, α_{e_i}.
  Define the set $\hat{J}$ in equation (4.14).
  for e_i with $i \in \hat{J}$ **do**
    Refine element e_i to two equidistant elements.
  end for
  if $\theta_{\max}/\theta_{\min} > \theta_c$ **then**
    while $M - M_D \le r_M(M_D - M_{\mathsf{old}})$ **do**
      Do h-refinement for the element with largest θ_{e_i}.
      Set the local indicator $\theta_{e_i} = 0$ for child elements.
    end while
  end if
  Solve problem (3.10) using the new partition $\mathcal{T}_h^{k+1}$ to obtain MAP $\bar{\boldsymbol{\phi}}_{h,s}^{*,k+1}$ and $\bar{\boldsymbol{\psi}}_{h,s}^{*,k+1}$.
  $\epsilon \leftarrow \left(S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,k}, \bar{\boldsymbol{\psi}}_{h,s}^{*,k}) - S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,k+1}, \bar{\boldsymbol{\psi}}_{h,s}^{*,k+1})\right) / S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,k+1}, \bar{\boldsymbol{\psi}}_{h,s}^{*,k+1})$.
end while

4.4. The delay parameter. Intuitively, when the memory goes further to the past, i.e., τ is larger, the problem itself will become more nonlinear. One obvious effect of a larger τ on the computation is that the computation of gradient is more expensive since one element is correlated to more other elements. More importantly, the delay can significantly change the dynamical behavior, which makes the optimization problem (3.10) more ill-conditioned.

4.4.1. The effect of delay on stability. We illustrate the effect of delay on stability using the following linear system:

$$\begin{cases} \dot{\boldsymbol{x}}_t = A\boldsymbol{x}_t + B\boldsymbol{x}_{t-\tau}, & t \in [0, T], \\ \boldsymbol{x}_t = \theta(t), & t \in [-\tau, 0], \end{cases} \tag{4.16}$$

where we assume that the linear system is stable when the time delay $\tau = 0$. In other words, we assume that $(A+B)$ is normal and $(A+B)+(A+B)^{\mathsf{T}}$ is negative definite such that when $\tau = 0$

$$|\boldsymbol{x}_t|^2 = \langle e^{Ct}\boldsymbol{x}_0, e^{Ct}\boldsymbol{x}_0\rangle = \langle e^{(C+C^{\mathsf{T}})t}\boldsymbol{x}_0, \boldsymbol{x}_0\rangle \le |\boldsymbol{x}_0| \left| e^{(C+C^{\mathsf{T}})t}\boldsymbol{x}_0 \right| \to 0, \quad \text{as } t \to \infty,$$

where $C = A + B$. Equation (4.16) can be solved by the method of steps, where the solution is obtained on the time intervals $[i\tau, (i+1)\tau]$ with $i = 0, 1, \ldots$ using the information in the previous interval as the initial condition. For example, for $t \in [0, \tau]$, we can integrate equation (4.16) to obtain

$$\boldsymbol{x}_t = e^{At}\boldsymbol{x}_0 + \int_0^t e^{A(t-q)} B\boldsymbol{x}_{q-\tau} dq. \tag{4.17}$$

Once we obtain $\boldsymbol{x}_t$ with $t \in [0, \tau]$, we can use the same formula to compute $\boldsymbol{x}_t$ with $t \in [\tau, 2\tau]$. This process can be repeated to obtain $\boldsymbol{x}_t$ with $t \in [-\tau, \infty)$. Consider the

Laplace transform of equation (4.16):

$$\tilde{s}\tilde{\boldsymbol{X}}_{\tilde{s}} - \theta(0) = A\tilde{\boldsymbol{X}}_{\tilde{s}} + B\left[e^{-\tilde{s}\tau}\tilde{\boldsymbol{X}}_{\tilde{s}} + \int_{-\tau}^{0} e^{-\hat{s}(q+\tau)}\theta(q)dq\right],$$

where $\tilde{\boldsymbol{X}}_{\tilde{s}}$ is the Laplace transform of $\boldsymbol{x}_t$. We have

$$\tilde{\boldsymbol{X}}_{\tilde{s}} = (\tilde{s}I - A - e^{-\tilde{s}\tau}B)^{-1}\left[\theta(0) + B\int_{-\tau}^{0} e^{-\hat{s}(q+\tau)}\theta(q)dq\right]. \tag{4.18}$$

We define the following characteristic function

$$g(\tilde{s}; e^{-\tau\tilde{s}}) = (\tilde{s}I - A - e^{-\tilde{s}\tau}B). \tag{4.19}$$

For a certain delay τ, if

$$g(\tilde{s}; e^{-\tau\tilde{s}}) \neq 0, \quad \forall \tilde{s} \in \bar{\mathbb{C}}_+, \tag{4.20}$$

where $\bar{\mathbb{C}}_+$ is the closed right half complex plane, we say the system is stable (see definition 2.1 in [11]). When τ is beyond a certain threshold, the condition (4.20) may fail and the system loses its stability. Although the main numerical difficulties for approximation remain similar no matter that the system is stable nor not, the dynamics may change significantly as τ increases, which makes it challenging to propose a good initial path for the optimization iteration. Let us illustrate this issue using an example.

Example 4.5. *Consider*

$$A = \begin{bmatrix} -2 & 0 \\ 0 & -0.9 \end{bmatrix}, \quad B = \begin{bmatrix} -1 & 0 \\ -1 & -1 \end{bmatrix}.$$

Apparently when $\tau = 0$, the system is stable. We now increase the time delay with the following initial conditions:

$$x_{1,t} = t^2 + 0.1, \quad x_{2,t} = -t^2 + 0.1.$$

It can be verified through equation (4.20), $(0,0)$ will lose its stability when $\tau \gtrapprox 6.1725$. In figure 4.3, we compared the dynamics given by different time delays. It is seen that as τ increases the trajectory of the delayed system changes significantly. If we use the points $(0.1, 0.1)$ and $(0,0)$ as the starting and ending points for the minimum action method, the minimizer should be consistent with the trajectory. For the case $\tau = 0$, we can use a linear path as the initial guess to obtain the trajectory. However, for the case $\tau = 0.8$, we are not able to obtain the trajectory starting from a linear initial guess.

4.4.2. Growing the MAP. To alleviate the possible difficulties in the initialization of tMAM for time-delay systems, we propose a simple strategy: growing the MAP of a time-delay system from the case that $\tau = 0$. The strategy is illustrated in figure 4.3. Let $(0,0)$ indicate the coarsest mesh with zero time delay and $(1,1)$ indicates the finest mesh with the desired time delay, where the coordinates are understood as a degree for the corresponding task. We then need to select a pathway from $(0,0)$ to $(1,1)$. There exist many choices for such a purpose. Two simplest choices include: 1) fully refine the mesh first for $\tau = 0$ and then increase the time delay from 0 to τ, and 2) increase the time delay from 0 to τ on the coarse mesh and

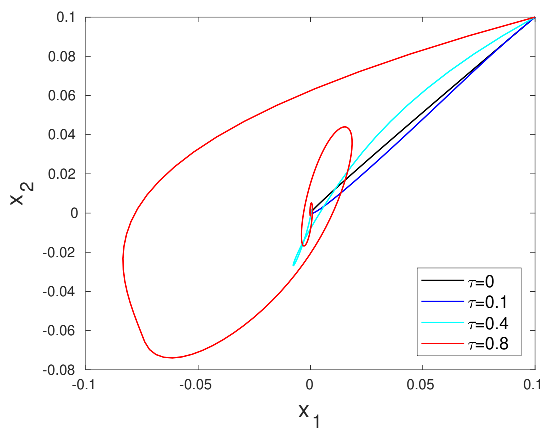

FIG. 4.3. *The comparison of dynamics given by different time delays for the problem defined in example 4.5.*

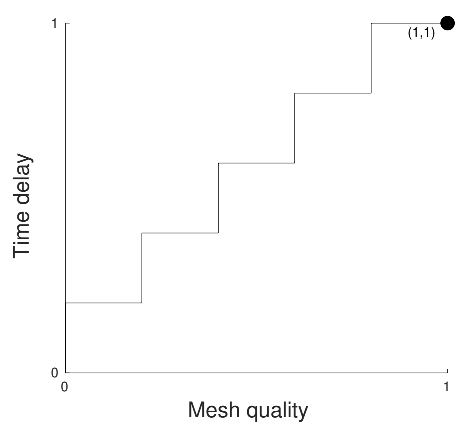

FIG. 4.4. *Our adaptivity strategy is illustrated by the "stairs", where each mesh refinement is followed by an increase of time delay. The starting point* $(0, 0)$ *indicates a coarse mesh with zero time delay, and the ending point* $(1, 1)$ *indicates the finest mesh with the desired time delay.*

then implement mesh refinement. Both choices are not effective. For the first choice, we do not know if the fine mesh for $\tau = 0$ is sufficient for $\tau \neq 0$; For the second choice, a coarse mesh is obviously not able to handle the possible complexity induced by the time delay (see figure 4.3). In this work, we pick a zigzag pathway close to the straight line from $(0, 0)$ to $(1, 1)$, which interweaves the mesh refinement and the increasing of the time delay, see figure 4.4 and Algorithm 2.

REMARK 4.6. *The idea of Algorithm 2 can also be applied to the penalty parameter* β *such that we can interweave the mesh refinement and the increment of* β *to obtain more accuracy and efficiency.*

5. Numerical experiments.

In this section, we present some numerical experiments to demonstrate the effectiveness of our algorithm. For verification, we mainly use the MAM to approximate the trajectory of an unperturbed system, along which the action functional is zero. Considering the regularity of the solution of ordinary

Algorithm 2 Adaptive tMAM for time-delay systems by interweaving h-refinement and the increment in time delay.

Choose an initial partition $\mathcal{T}_h^{\texttt{old}} = \mathcal{T}_h^{\texttt{new}}$, and a step size $\Delta\tau$. Let $N = \tau_{\texttt{final}}/\Delta\tau$, $\tau = 0$ and $\epsilon = 1$.
for $k \leftarrow 1$ to N **do**
  $\tau \leftarrow \tau + \Delta\tau$
  Solve problem (3.10) to obtain $\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{old}}$ and $\bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{old}}$ on partition $\mathcal{T}_h^{\texttt{old}}$.
  if $\epsilon > \epsilon_{\text{tol}}$ **then**
    Refine the partition using Algorithm 1 to obtain new partition $\mathcal{T}_h^{\texttt{update}}$.
    Solve problem (3.10) using the partition $\mathcal{T}_h^{\texttt{update}}$ to obtain $\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{update}}$ and $\bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{update}}$.
    $\epsilon \leftarrow \left| S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{update}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{update}}) - S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{old}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{old}}) \right| / S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{update}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{update}})$.
    $\mathcal{T}_h^{\texttt{old}} \leftarrow \mathcal{T}_h^{\texttt{new}}$.
    $\mathcal{T}_h^{\texttt{new}} \leftarrow \mathcal{T}_h^{\texttt{update}}$.
  else
    Solve problem (3.10) using the new partition $\mathcal{T}_h^{\texttt{new}}$ to obtain $\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{new}}$ and $\bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{new}}$.
    $\epsilon \leftarrow \left| S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{new}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{new}}) - S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{old}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{old}}) \right| / S_\tau(\bar{\boldsymbol{\phi}}_{h,s}^{*,\texttt{new}}, \bar{\boldsymbol{\psi}}_{h,s}^{*,\texttt{new}})$.
  end if
end for
while $\epsilon > \epsilon_{\text{tol}}$ **do**
  Implement Algorithm 1 to refine the mesh.
end while

differential equations with constant time delays [2], the main characteristics is the propagation of discontinuities at time $i\tau$, $i = 0, 1, 2, \ldots$. At $t = 0$, we usually have $\dot{\theta}(0)^- \neq \dot{\boldsymbol{x}}_0^+$ (see equation (4.16)), where $^-$ and $^+$ indicate the left and right derivative respectively. At $t = \tau$, the jump in $\dot{\boldsymbol{x}}_0$ will induce a jump in $\ddot{\boldsymbol{x}}_\tau$ although $\dot{\boldsymbol{x}}_\tau$ is continuous. In general, the derivative jump at $t = 0$ will propagate along the integration interval and give rise to subsequent discontinuity points at $t = i\tau$ where the solution is smoothed out more and more. As a consequence, even the force term is C^∞, the solution $\boldsymbol{x}_t$ is simply C^1-continuous. Based on such an observation, we only consider linear finite elements in the numerical experiments if the convergence rate is needed. However, since the regularity of the solution is improved as the evolution time t increases, high-order finite elements are in general more efficient. In MATLAB, the trajectory ϕ_t^{d} can be computed by the subroutine `dde23()` [23].

5.1. Adaptivity behavior. We consider a simple linear system with time delays:

$$\begin{cases} d\boldsymbol{X}_t = A\boldsymbol{X}_t + B\boldsymbol{X}_{t-\tau}dt + \sqrt{\varepsilon}d\boldsymbol{W}_t, & t \in [0, T], \\ \boldsymbol{X}_t = \theta(t), & t \in [-\tau, 0]. \end{cases} \tag{5.1}$$

Let

$$A = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} a & b \\ -b & a \end{bmatrix}, \quad B = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix},$$

with $a = 1/3$, $b = \sqrt{8}/3$, $\lambda_1 = -5$, and $\lambda_2 = -1$. We use the MATLAB solver `dde23` to compute a trajectory ϕ_t^{d} for the unperturbed system.

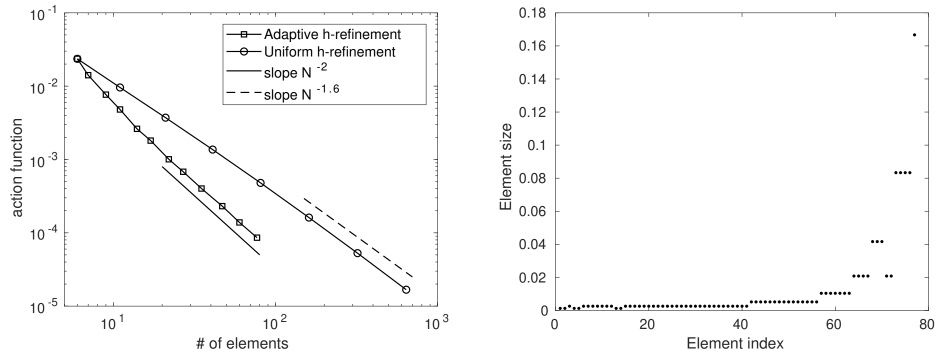

FIG. 5.1. *The convergence of tMAM with adaptive h-refinement and uniform h-refinement. Linear finite elements are used for discretization. The penalty parameter is fixed as $\beta = 1.0$. The bulk parameter for adaptivity is $r_\eta = 0.4$. The optimal convergence rate is N^{-2}. The initial coarse mesh is given by six equidistant linear finite elements. Left: convergence rates of adaptive tMAMs; Right: the distribution of element size of the adaptive mesh.*

5.1.1. Small time delay. We first look at the case that the time delay is relatively small using Algorithm 1. We consider equation (5.1) with the following initial conditions:

$$\theta(t) = [0.5e^t, 0.5e^t]^\mathsf{T}. \tag{5.2}$$

Let $\tau = 0.05$. Let the starting point be $\phi^\mathsf{d}_{t=0} = (0.5, 0.5)^\mathsf{T}$ and the ending point be $\phi^\mathsf{d}_{t=10} \approx (9.5 \times 10^{-8}, 2.5 \times 10^{-7})^\mathsf{T}$ such that the minimizer of the action function is ϕ^d_t with $t \in [0, T^* = 10]$. Note that $\phi^\mathsf{d}_{t=\infty} = (0, 0)$ is a stable fixed point for the unperturbed system. Due to the fact that $\phi^\mathsf{d}_{t=10} \approx \phi^\mathsf{d}_{t=\infty}$, seeking the minimizer ϕ^d_t with $t \in [0, T^* = 10]$ shares similar difficulties to the case that $T^* = \infty$. For this case, we simply use a linear path as the initial guess. In figure 5.1, we plot the convergence behavior of tMAM with adaptive h-refinement and uniform h-refinement on the left, and the distribution of element size of the adaptive mesh on the right. First, the uniform refinement achieves algebraic convergence with a rate that is smaller than the optimal one $O(N^{-2p})$. Since $\boldsymbol{x}_t$ is C^1-continuous, the optimal convergence rate is achievable for $p = 1$. This is similar to the results for systems without time delays [32]. More specifically, equation (2.8) becomes degenerate as the optimal integration time goes to infinity, and uniform refinement is not able to achieve the optimal convergence rate for this kind of problems. Note that this issue is independent of the time delay. Second, the adaptive h-refinement based on the a posteriori error estimate can significantly improve the convergence rate. For the problem studied, the optimal rate has actually been recovered. Third, the element size $|e_i| = |s_i - s_{i-1}|$ becomes larger as the path approaches the stable fixed point $(0, 0)$, which means that the a posteriori error estimator effectively captures the fact that the regularity is low in the region of fast dynamics [29].

5.1.2. Large time delay. We now look at the application of Algorithm 2 to the case that the time delay is relatively large. The initial condition $\theta(t)$ is the same as the previous case except that the time delay changes from $\tau = 0.05$ to $\tau = 1$. In figure 5.2, we plot the trajectories ϕ^d_t for $\tau = 0.05, 1$. Compared to a small time delay, the large time delay $\tau = 1$ introduces dramatic oscillations when the trajectory converges to $(0, 0)$, which makes the linear path not effective as an initial guess for

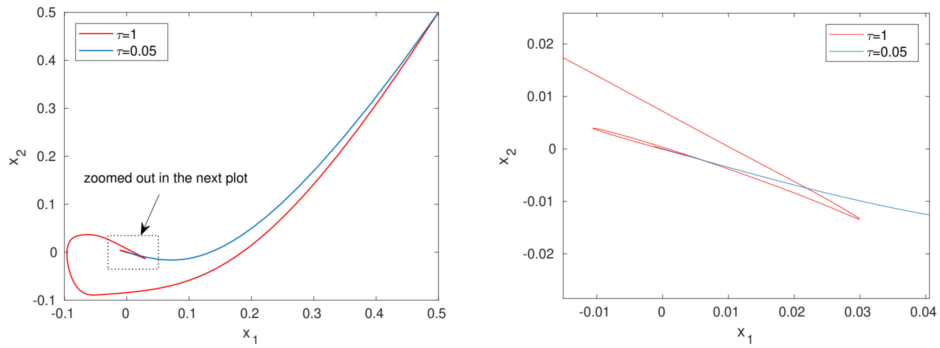

FIG. 5.2. *The trajectories ϕ_t^{d} for equation* (5.1) *subject to initial condition* (5.2)*. Left: ϕ_t^{d} on $t \in [0, 10]$ for $\tau = 0.05, 1$. Right: Close-up view of the region enclosed by the rectangle in the previous plot.*

the minimum action method. In other words, starting from the linear path, the optimization solver will converge to a local minimizer that is not the solution. Let $\phi_{t=10}^{\mathsf{d}}$ be the ending point. Starting from $\tau = 0.05$ and an initial linear path, we increase τ by $\frac{1-0.05}{10} = 0.095$ after each mesh refinement. After $\tau = 1$ is reached, we keep refining the mesh until the prescribed tolerance in action function is achieved. In figure 5.3, we compare the exact solution and the approximate solution given by Algorithm 2. It is seen that Algorithm 2 works effectively, which captures not only the overall path but also the details around $(0, 0)$. From plot (a) to (d), the characteristic scale of the path decays approximately from $O(1)$ to $O(10^{-3})$, where all abrupt turns in the path, except the last one shown in plot (d), have been well captured.

5.2. Phase transition problem. We add a pair of time-delay terms to a classical physical model to look at the effect of time delay on phase transition. We consider the following modified Maier-Stein model [16]:

$$\begin{cases} dX_t &= (X_t - X_t^3 - \beta X_t Y_t^2 - \frac{1}{2}(X_{t-\tau} - X_t))dt + \sqrt{\varepsilon} dW_t^x \\ dY_t &= -(Y_t + X_t^2 Y_t + \frac{1}{2}(Y_{t-\tau} - Y_t))dt + \sqrt{\varepsilon} dW_t^y \end{cases}, \tag{5.3}$$

where W_t^x and W_t^y are independent Wiener processes and $\beta > 0$ is a parameter. When $\tau = 0$, the original Maier-Stein (MS) model will be recovered. In this work, the delayed terms are only added for numerical purpose without any physical motivations.

The original Maier-Stein model has two stable fixed points: $\boldsymbol{a}_1 = (-1, 0)^{\mathsf{T}}$ and $\boldsymbol{a}_2 = (1, 0)^{\mathsf{T}}$, and one saddle point $\boldsymbol{a}_3 = (0, 0)^{\mathsf{T}}$. We choose τ such that the stability of $\boldsymbol{a}_i$, $i = 1, 2, 3$, remains the same. For numerical experiments, we set $\beta = 10$. We start with a coarse mesh with 6 quadratic elements, and increase the time delay τ from 0.05 to 1, and increase the penalty parameter β from 10 to 200. We increase τ and β at the same time for each mesh refinement, where the maximum values of both τ and β are reached in 10 steps. Let us write $\hat{S}_\tau(\bar{\phi}_s, \bar{\psi}_s) = \hat{S}_{\tau,\mathsf{action}} + \hat{S}_{\tau,\mathsf{penalty}}$, where

$$\hat{S}_{\tau,\mathsf{action}} = \frac{\hat{T}}{2} \int_0^1 |\hat{T}^{-1} \bar{\boldsymbol{\phi}}_s' - \boldsymbol{b}(\bar{\boldsymbol{\phi}}_s, \bar{\boldsymbol{\psi}}_s)|^2 ds, \quad \hat{S}_{\tau,\mathsf{penalty}} = \frac{\beta^2}{2} \int_0^1 |\bar{\boldsymbol{\psi}}_s - \bar{\boldsymbol{\phi}}_{\hat{s}}|^2 ds.$$

For the approximated MAP, we have $\frac{\hat{S}_{\tau,\mathsf{penalty}}}{\hat{S}_{\tau,\mathsf{action}}} \approx 10^{-4}$, meaning that the constraint is sufficiently enforced. In figure 5.4, we compare the most probable transition paths

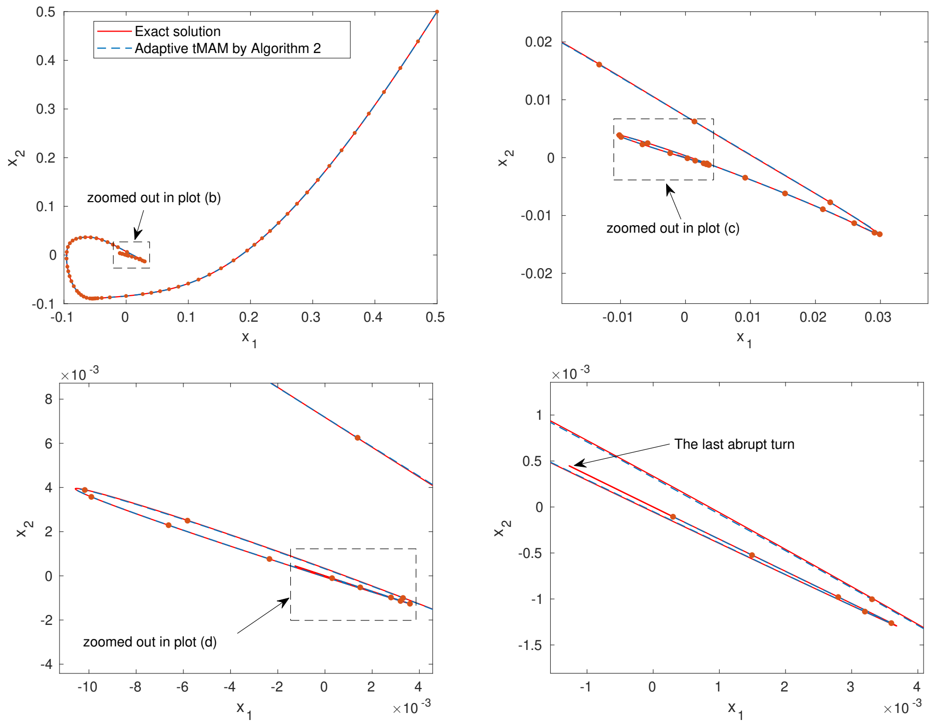

FIG. 5.3. *The approximate solution is computed by Algorithm 2, where the value of the action function is* 1.34×10^{-8}. *The final mesh has 79 quadratic finite elements, and the grid points for the finite element mesh are indicated by the red dots.* $\beta = 1$. *(a). Compare the exact solution and the approximate one. (b)-(c). Close-up view of the region enclosed by the rectangle in the previous plot.*

of the MS model and the modified MS model with $\tau = 1$, where the grid points correspond to the finite element mesh. On the one hand, the transition mechanism is similar for both cases, where both MAPs approach the saddle point first and then follow the unstable manifold to the other fixed point; on the other hand, the effect of the time delay is substantial, where the actions of the MAPs are 0.34 and 0.18, respectively, for the cases without and with time delays although it seems that the MAPs do not differentiate that much. It is seen that in the right plot of figure 5.4, the MAP does not exactly reach the saddle point $(0, 0)$, which is mainly due to the fact that the number of finite elements is relatively small. The saddle point will be captured better by setting the tolerance ϵ_{tol} in Algorithms 1 and 2 smaller such that more elements will be constructed around the saddle point. More discussions about the approximation around unknown critical points can be found in [25, 29]. The relation between the action of the MAP and the time delay has been plotted in figure 5.5. For the problem studied, as the time delay increases, the action of the MAP decreases, meaning that the time delay makes the transition easier for the problem studied, see equation (2.3). The relation between different forms of time delay and the action of the MAP is in general an open question, which deserves further studies.

In figure 5.6, we plotted the convergence behavior of h-adaptive tMAM with linear elements, i.e., $p = 1$, for the MS model. The reference solution is computed by h-adaptive tMAM with 2584 linear elements. The initial coarse mesh has 3 elements.

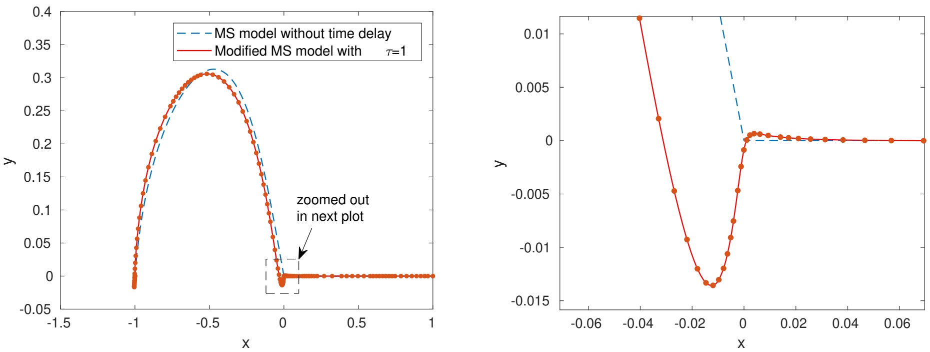

FIG. 5.4. *The MAP has been computed by Algorithm 2. The final mesh has 56 quadratic finite elements with $\beta = 200$. The ratio between the penalty term and the action term is about 10^{-4}. Left: The comparison between the MAPs of the MS model without time delay and the modified MS model with $\tau = 1$. Right: Close-up view of the region enclosed by the rectangle in the previous plot.*

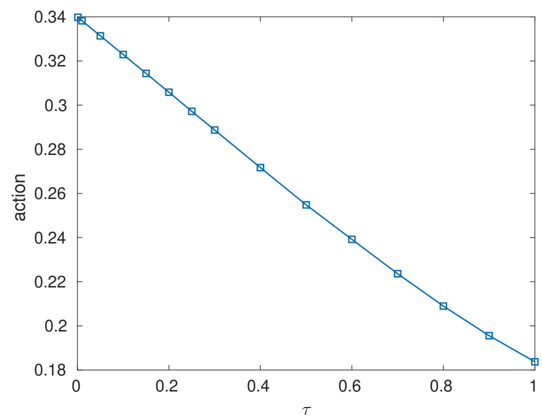

FIG. 5.5. *The time delay versus the action of the MAP for the modified MS model.*

Starting from $\tau = 0$ and $\beta = 10$, we increase τ by $\frac{1-0}{10}$ and β by $\frac{300-10}{10}$ before each mesh refinement until the desired values $\tau = 1$ and $\beta = 300$ are reached. The bulk parameter is $r_\eta = 0.5$ for mesh refinement. Data have been collected when $\tau = 1$ and $\beta = 300$. It is seen in figure 5.6 that the overall convergence rate agrees well with the optimal rate $O(N^{-2})$ in terms of the error of the action functional.

6. Summary and discussions. In this work, we have developed a minimum action method to seek the most probable transition path in systems with constant time delays. Since the Hamiltonian is not conservative any more, the Maupertuis principle does not apply, and we need to work with the action functional formulated with respect to time. We define an auxiliary path $\psi_t = \phi_{t-\tau}$ such that the action functional will not depend on τ explicitly, which means that we can use a simple optimal linear scaling to remove the optimization with respect to T. The constraints $\psi_t = \phi_{t-\tau}$ will be enforced through a quadratic penalty term included in the original action functional. Adaptive discretization is necessary for the minimum action method formulated with respect to time. We have adapted a posteriori error estimate, developed in [29] for systems

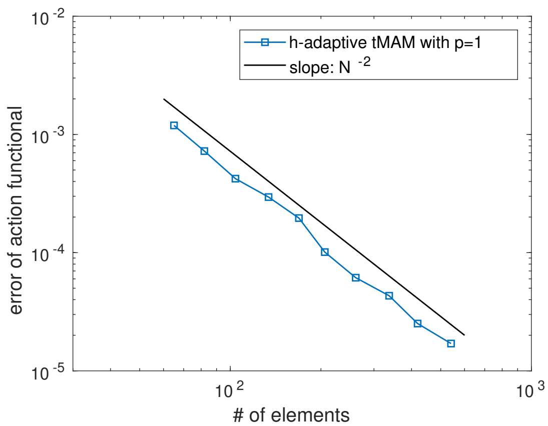

Fig. 5.6. *The convergence behavior of the h-adaptive tMAM with linear finite elements for the MS model. The penalty parameter is $\beta = 300$. The bulk parameter for adaptivity is $r_\eta = 0.5$. Starting from $\tau = 0$ and $\beta = 10$, we increase τ by $\frac{1-0}{10}$ and β by $\frac{300-10}{10}$ before each mesh refinement until the desired $\tau = 1$ and $\beta = 300$ are achieved.*

without time delays, for our problem by including the difference $\psi_t - \phi_{t-\tau}$ into the error indicator. Another difficulty comes from large time delays, which may significantly change the dynamics. For the optimization iteration in the minimum action method, the initial guess that is valid for the systems without time delays may not work any more. To deal with this issue, we consider a sequence of time delays, where the time delay increases gradually. More specifically, we interweave the mesh refinement and the increment of time delay such that the MAP will grow from a coarse mesh for a system without time delays to a fine adaptive mesh for a system with a desired time delay. Preliminary numerical results have verified the effectiveness of the proposed strategy. Many possible improvements can be made, e.g., the augmented Lagrangian method can be employed for the optimization, and different meshes can be used for ϕ_t and ψ_t, etc. Theoretical issues, such as the convergence of the approximated solution and the choice of the penalty parameter, etc., need to be analyzed. The study on these issues will be reported elsewhere.

Acknowledgement. This work is supported by NSF Grants DMS-1620026 and DMS-1913163. Zhai was supported in part by an appointment with the NSF Mathematical Sciences Summer Internship Program sponsored by the National Science Foundation, Division of Mathematical Sciences (DMS). This program is administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the U.S. Department of Energy (DOE) and NSF. ORISE is managed by ORAU under DOE contract number DE-SC0014664.

REFERENCES

[1] R. Azencott, B. Geiger and W. Ott, *Large Deviations for Gaussian Diffusions with Delay*, J. Stat. Phys. **170(2)** (2018) 254-285.

[2] A. Bellen and M. Zennaro *Numerical Methods for Delay Differential Equations*, Oxford University Press, 2003.

[3] C. Briat, H. Hjalmarsson, K. H. Johansson, G. Karlsson, U. T. Jönsson and H. Sandberg, *Nonlinear state-dependent delay modeling and stability analysis of internet conges-*

tion control, 49th IEEE Conference on Decision and Control, Atlanta, USA, 2010, 1484–1491.

[4] C. Briat, E. A. Yavuz and G. Karlsson, *A conservation-law-based modular fluid-flow model for network congestion modeling*, 31st IEEE International Conference on Computer Communications (INFOCOM), Orlando, Florida, USA, 2012, 2050–2058.

[5] W. E, W. Ren and E. Vanden-Eijnden, *Minimum action method for the study of rare events*, Commun. Pure Appl. Math., **57** (2004), 637-565.

[6] M. Faggian, F. Ginelli, F. Marino and G. Giacomelli, *Evidence of a critical phase transition in a purely temporal dynamics with long-delayed feedback* Phys. Rev. Lett. **120**, 173901 (2018)

[7] H.C. Fogedby and W. Ren, *Minimum action method for the Kardar-Parisi-Zhang equation*, Phys. Rev. E, **80** (2009), 041116.

[8] M. Freidlin and A. Wentzell, *Random Perturbations of Dynamical Systems*, second ed., Springer-Verlag, New York, 1998.

[9] T. Grafke, R. Grauer, T. Schäfer and E. Vanden-Eijnden, *Arclength parametrized Hamilton's equations for the calculation of instantons*, SIAM, Multiscale Model. Simul., **12(2)** (2014), 566–580.

[10] T. Grafke, T. Schäfer and E. Vanden-Eijnden, *Long-term effects of small random perturbations on dynamical systems: theoretical and computational tools*, In: R. Melnik, R. Makarov and J. Belair (eds), Recent Progress and Modern Challenges in Applied Mathematics, Modeling and Computational Science, Fields Institute Communications, vol 79. Springer, New York, NY.

[11] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*, Birkhäuser, 2003.

[12] J. K. Hale and S. M. Verduyn Lunel, *Introduction to Functional Differential Equations*, Springer, New York, 1991.

[13] Y. Halevi, *Optimal reduced order models with delay*, Proceedings of the 30th Conference on Decision and Control, Brighton, England, 602-607, 1991.

[14] Y. Halevi, *Reduced-order models with delay*, International Journal of Control, **64**, 733-744, 1996.

[15] J. P. Hespanha, P. Naghshtabrizi and Y. Xu, *A survey of recent results in networked control systems*, Proc. IEEE, 95(1) (2007), 138–162.

[16] M. Heymann and E. Vanden-Eijnden, *The geometric minimum action method: A least action principle on the space of curves*, Commun. Pure Appl. Math., **61** (2008), 1052-1117.

[17] G. Karniadakis and S. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*, Oxford University Press, 2nd Edition, 2005.

[18] C. Katrakaza, M. Quddus, W.-H. Chen, and L. Deka, *Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions*, Transport. Res. Part C: Emerg. Technol., **60** (2015), 416–442.

[19] W. Michiels, C.-I. Morarescu and S.-I. Niculescu, *Consensus problems with distributed delays, with application to traffic flow models*, SIAM J. Control Optim., 48(1) (2009), 77–101.

[20] C. Mo and J. Luo, *Large deviations for stochastic differential delay equations*. Nonlinear Analysis. **80** (2013), 202-210.

[21] S.-E. A. Mohammed, T. Zhang, *Large deviations for stochastic systems with memory*. Discrete & Continuous Dynamical Systems - B, **6(4)** (2006), 881-893.

[22] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, 1999.

[23] L. F. Shampine and S. Thompson, *Solving DDEs in MATLAB*, Appl. Numer. Math., **37** (2001), 441–458.

[24] N. R. Smith, B. Meerson and P. V. Sasorov, *Local average height distribution of fluctuating interfaces*, Phys. Rev. E, **95** (2017), 012134.

[25] Y. Sun and X. Zhou, *An improved adaptive minimum action method for the calculation of transition path in non-gradient systems*, Commun. Compt. Phys., **24(1)** (2017), 44–68.

[26] X. Wan, *An adaptive high-order minimum action method*, J. Compt. Phys., **230** (2011), 8669-8682.

[27] X. Wan and G. Lin, *Hybrid parallel computing of minimum action method*, Parallel Computing, **39** (2013), 638–651.

[28] X. Wan, *A minimum action method with optimal linear time scaling*, Comm. Compt. Phys., **18(5)** (2015), 1352–1379.

[29] X. Wan, B. Zheng and G. Lin, *An hp adaptive minimum action method based on a posteriori error estimate*, Comm. Compt. Phys., **23(2)** (2018), 408–439.

[30] X. Wan, H. Yu and W. E, *Model the nonlinear instability of wall-bouned shear flows as a rare*

event: A study on two-dimensional Poiseuille flows, Nonlinearity, **28** (2015), 1409–1440.

[31] X. Wan and H. Yu, *A dynamic-solver-consistent minimum action method: With an application to 2D Navier-Stokes equations*, J. Compt. Phys., **331** (2017), 209–226.

[32] X. Wan, H. Yu and J. Zhai, *Convergence analysis of a finite element approximation of minimum action method*, SIAM J. Numer. Anal., **56(3)** (2018), 1597–1620.

[33] D. K. Wells, W. L. Kath and A. E. Motter, *Control of stochastic and induced switching in biophysical networks*, Phys. Rev. X **5** (2015), 031036.

[34] E. Witrant, C. Canudas-de-Wit, D. Gerorges and M. Alamir, *Remote stabilization via communication networkswith a distributed control law*, IEEE Trans. Autom. Control, 52(8) (2007), 1480–1485.

[35] W. Yao and W. Ren, *Noise-induced transition in barotropic flow over topography and application to Kuroshio*, J. Comput. Phys. **300** (2015), 352-364.

[36] W. Yao and W. Ren, *Liquid-vapor transition on patterned solid surfaces in a shear flow*, J. Chem. Phys. **143** (2015), 244701.

[37] X. Zhou, W. Ren and W. E, *Adaptive minimum action method for the study of rare events*, J. Chem. Phys., **128** (2008), 104111.