SIAM J. NUMER. ANAL.
Vol. 63, No. 1, pp. 360–395
© 2025 Society for Industrial and Applied Mathematics

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

PARAMETERIZED WASSERSTEIN HAMILTONIAN FLOW*

HAO WU†, SHU LIU‡, XIAOJING YE§, AND HAOMIN ZHOU†

Abstract. In this work, we propose a numerical method to compute the Wasserstein Hamiltonian flow (WHF), which is a Hamiltonian system on the probability density manifold. Many well-known PDE systems can be reformulated as WHFs. We use the parameterized function as a push-forward map to characterize the solution of WHF, and convert the PDE to a finite-dimensional ODE system, which is a Hamiltonian system in the phase space of the parameter manifold. We establish theoretical error bounds for the continuous time approximation scheme in the Wasserstein metric. For the numerical implementation, neural networks are used as push-forward maps. We design an effective symplectic scheme to solve the derived Hamiltonian ODE system so that the method preserves some important quantities such as Hamiltonian. The computation is done by a fully deterministic symplectic integrator without any neural network training. Thus, our method does not involve direct optimization over network parameters and hence can avoid errors introduced by the stochastic gradient descent or similar methods, which are usually hard to quantify and measure in practice. The proposed algorithm is a sampling-based approach that scales well to higher dimensional problems. In addition, the method also provides an alternative connection between the Lagrangian and Eulerian perspectives of the original WHF through the parameterized ODE dynamics.

Key words. Hamiltonian dynamics, Wasserstein Hamiltonian flow, deep learning, symplectic Euler scheme, numerical analysis

MSC codes. 65M15, 68T07, 49Q22

DOI. 10.1137/23M159281X

1. Introduction.

Wasserstein Hamiltonian flow (WHF) describes the time evolution of a Hamiltonian system on a Wasserstein manifold. It can be formulated as the following first-order Hamiltonian system of dual coordinates on the Wasserstein manifold, which is the space of probability densities equipped with optimal transport distance [49]:

$$\partial_t \rho = \frac{\delta}{\delta \Phi} \mathcal{H}(\rho, \Phi), \tag{1.1a}$$

$$\partial_t \Phi = -\frac{\delta}{\delta \rho} \mathcal{H}(\rho, \Phi), \tag{1.1b}$$

with given initial values ρ_0 and Ψ_0 at all $x \in \mathbb{R}^d$,

$$\rho(0, x) = \rho_0(x) \qquad \text{and} \qquad \Phi(0, x) = \Phi_0(x). \tag{1.2}$$

*Received by the editors August 9, 2023; accepted for publication (in revised form) September 5, 2024; published electronically February 14, 2025.
https://doi.org/10.1137/23M159281X
Funding: The work of the authors was partially supported by National Science Foundation grants DMS-1925263, DMS-2152960, DMS-2307465, and DMS-2307466 and by ONR grant N00014-21-1-2891.
†School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332 USA (hwu406@gmail.com, hmzhou@gatech.edu).
‡Department of Mathematics, University of California, Los Angeles, CA 90095 USA (shuliu@math.ucla.edu).
§Department of Mathematics and Statistics, Georgia State University, Atlanta, GA 30303 USA (xye@gsu.edu).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

In (1.1), $\frac{\delta}{\delta\rho}$ is the L^2 first variation, ρ is the probability density on the state space $\mathbb{R}^d$, i.e., ρ is a nonnegative function with $\int_{\mathbb{R}^d}\rho(x)dx=1$, Φ is called the dual function, whose gradient provides the vector field transporting ρ on the Wasserstein manifold, and ρ_0 and Φ_0 are their initial values, respectively. We note that our results throughout this work apply to the case where the state space of x is any Riemannian manifold M without boundary, but for notation simplicity, we only consider the case $M=\mathbb{R}^d$ in this work. We may also write $\rho(t,\cdot)$ as $\rho_t(\cdot)$ or ρ_t for notation simplicity, and similarly for $\Phi(t,\cdot)$. We consider the following general class of Hamiltonian:

$$\mathcal{H}(\rho,\Phi)=\int_{\mathbb{R}^d}\frac{1}{2}|\nabla\Phi(x)|^2\rho(x)dx+\mathcal{F}(\rho), \tag{1.3}$$

where the first term is the kinetic energy associated with the 2-Wasserstein metric, and $\mathcal{F}(\rho)$ is a potential functional defined on the Wasserstein manifold, which typically is one or a combination of the three terms appeared in the following formula:

$$\mathcal{F}(\rho)=\int_{\mathbb{R}^d}V(x)\rho(x)dx+\int_{\mathbb{R}^d}U(\rho,x)dx+\iint_{\mathbb{R}^d\times\mathbb{R}^d}W(x-y)\rho(x)\rho(y)dxdy. \tag{1.4}$$

Here the first term is determined by the linear potential V, the second one is a nonlinear functional U of ρ such as entropy or Fisher information, and the third is an interactive potential W between particles whose population density is given by ρ. Recent work [12] reveals that WHF has deep connections to many well-known partial differential equations (PDEs), such as the Wasserstein geodesic, Vlasov, and Schrödinger equations, just to name a few. WHF provides an alternative framework and a set of new tools originated from optimal transport that potentially can be used to study those PDEs and relevant applications. However, computation of WHF remains a challenging problem, especially in higher dimensions. In this work, we develop a computational framework to solve WHF by leveraging several techniques together, including generative models, neural networks, the symplectic integrator, and the Wasserstein metric on density manifold. In particular, our method is readily scalable to solve WHFs in high-dimensional spaces.

There are two main objectives in this paper. The first one is to derive an effective finite-dimensional approximation of WHF (1.1) by using reduced-order models. While the theory developed in this paper is applicable to general reduced-order models, we select a special class known as neural networks due to their excellent empirical approximation power in this study. Our derivation is conducted on a parameter space and a submanifold of the probability density space where a density function ρ is determined by a push-forward map parameterized by a neural network. The parameter space is equipped with an induced metric called the *pullback Wasserstein metric.* The resulting Hamiltonian system is a set of coupled ordinary differential equations (ODEs) for the neural network parameter and its dual, which is a finite-dimensional approximation to the infinite-dimensional WHF (1.1). We call this new Hamiltonian system the *parameterized* WHF (PWHF).

Our second objective is to develop a symplectic numerical scheme to solve the PWHF. This is accomplished by introducing an approximation to the pullback Wasserstein metric, which can be efficiently computed by recently developed machine learning techniques, such as residual neural networks [28, 29] or continuous normalizing flows [10, 24]. Moreover, the proposed method allows the computation of both particle motion of the Hamiltonian system in classical phase space and the density evolution on a Wasserstein manifold simultaneously.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Here we highlight several main features of the proposed method:

- (Dimension reduction) The PDEs (1.1), which can also be viewed as an infinite-dimensional dynamical system, are effectively approximated by a finite-dimensional ODE system and solved by a customized symplectic numerical scheme.
- (Computation effectiveness) A simplified Wasserstein metric is introduced to greatly reduce the computational cost when compared to that of the pullback Wasserstein metric on the parameter space.
- (Training free) The proposed method does *not* involve nonconvex optimization algorithms like stochastic gradient descent methods which are commonly adopted in machine learning. This avoids errors introduced by those optimization methods that are usually difficult to control and analyze.
- (Symplectic structure preservation) The proposed scheme preserves the symplectic structure of PWHF. Thus the Hamiltonian is conserved, even for large time horizon.
- (Error estimation) The convergence of the proposed scheme is guaranteed by error estimates obtained in the Wasserstein metric.
- (Eulerian and Lagrangian formulations) PWHF provides a natural connection bridging the Eulerian and Lagrangian formulations of the underlying Hamiltonian system.

The remainder of the paper is organized as follows. In section 2, we describe work related to this study. We briefly introduce the WHF and its equivalent formulations in section 3.1. We derive the PWHF and its simplified dynamics in sections 3.2 and 3.3, respectively. In section 3.4, we show that the density ρ_θ obtained by PWHF is a good approximation to the true solution with provable error bound. Then we provide a numerical algorithm to effectively solve the PWHF, with details about the simplified pullback Wasserstein metric tensor in section 4. Numerical results are given in section 5. We provide a discussion about potential applications of PWHF on other types of problems in section 6 and conclude the paper in section 7.

2. Related work. The formulation of WHF studied here is first introduced in the paper [12], where a derivation framework based on the Lagrangian functional for general WHF on a density manifold is proposed. This work also reveals the connections between WHF and several well-known PDEs through examples. Numerical methods have been developed for solving the WHF in recent works [13, 14], in which the classical finite difference and shooting techniques are used to solve WHF in lower dimensions.

We note that the idea of introducing the metric defined on a probability manifold to parameter space originates from [3], in which the Fisher metric is discussed. Later, the case of the Wasserstein metric was introduced and studied in [36] and [11]. Soon after, the Wasserstein gradient flows defined on the parameter space of the generative model were introduced in [34, 35, 39, 40].

The present study is mostly inspired by a recent work on the parametric Fokker–Planck equation (PFPE) [35, 40] which establishes a finite-dimensional approximation of the Fokker–Planck equation (FPE) by using push-forward maps, neural networks, and the Wasserstein metric. Leveraging the viewpoint that the FPE is the gradient flow of the relative entropy functional on a Wasserstein manifold [30, 41], PFPE is derived by taking the gradient flow of relative entropy projected onto the parameter space equipped with the pullback Wasserstein metric. The resulting PFPE is a system of ODEs for the parameters. Our work follows a similar strategy. We use the same

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

parameter space defined by push-forward maps and neural networks, and a similar pullback Wasserstein metric on the parameter space. Different from PFPE, our aim is to establish PWHF on the parameterized Wasserstein submanifold. In addition, we introduce a new metric, which can be viewed as a close approximation to the one introduced in [40]. Such a new metric does not require ρ_θ-weighted Helmholtz projection of vector fields $\partial_\theta T_\theta$ in [40]. As a result, using the new metric enables us to directly compute PWHF via a customized symplectic scheme with provable accuracy and significantly reduced computational cost.

Since introduced in the seminal works [31, 32] to describe the limiting behavior of stochastic differential games, mean field games (MFGs) have been studied extensively, including numerical methods [1, 2, 8] and machine learning–based approaches [15, 33, 44]. The works reported in [38, 44] provide methods for computing MFG in high-dimensional cases. The WHF is closely related to MFG systems at least in their mathematical forms, i.e., the MFG systems with quadratic kinetic energy can be treated as WHFs with boundary conditions. However, the WHF considered in this work and a typical MFG are different in the setting of initial and terminal conditions in their original form: WHF assumes knowledge of the initial (ρ_0, Φ_0) while MFG has a set of given initial condition ρ_0 and terminal condition Φ_τ at some terminal time τ.

Hamiltonian Monte Carlo algorithms introduced in [17] aim at generating samples from a given probability distribution by evolving an associated Hamiltonian system in the phase space. We refer readers to [6, 21] and the references therein for more details. Instead of sampling from a fixed terminal distribution, our research in the paper computes the entire density evolution of Hamiltonian flow.

There are also numerous pieces of research [9, 25, 46, 48] focusing on recovering the Hamiltonian and predicting the dynamics of certain physical systems based on observed trajectories. This is called the inverse problem in computing a Hamiltonian system in some literature. Neural networks are widely utilized in those studies to make the computation scalable for high-dimensional settings. Nevertheless, there are significant differences between our problem and theirs, with the most prominent one being that we aim at solving for the entire probability flow while the aforementioned research always focuses on particlewise computation.

In a broader sense, PWHF and the proposed numerical method provide an alternative approach that can potentially be applied to solve some PDEs in higher dimension by using neural networks. In the past few years, various machine learning methods have been developed for solving PDEs. For example, a deep learning method based on backward stochastic differential equations has been designed to solve high-dimensional parabolic PDEs in [18]. The deep Ritz method is studied to solve PDEs whose solutions can be reformulated as the minimizers of variational forms [19]. A physics-informed neural network is proposed as a general framework to solve PDEs by minimizing the residual in a least squares sense [26, 42]. The weak adversarial network solves PDEs in weak forms through a minimax approach [5, 51]. More recently, a Fourier neural operator [37], DeepONet [50], and Neural control [20] have been constructed to approximate the solution operators by neural networks so that the computation can be carried out more efficiently when the same PDEs must be solved repetitively with different initial or boundary conditions. Those and many more studies have shown that deep neural networks possess great potential in handling high-dimensional PDEs with various nonlinearities.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3. Parameterization of Wasserstein Hamiltonian flow. In this section, we first briefly review the Wasserstein metric and WHF, and then we derive the parameterization of WHF and suggest a strategy to speed up its computation by using an approximate Wasserstein metric. We provide a comprehensive error analysis of PWHF in the end.

3.1. Formulation of Wasserstein Hamiltonian flow. The review here follows the formulation detailed in [12]. For simplicity, let M be a smooth manifold without boundary. As mentioned in section 1, our results throughout this work can be applied to such M, but for simplicity we only consider the case $M = \mathbb{R}^d$ in theoretical and numerical demonstrations. Let us consider the density manifold,

$$\mathcal{P}(M) = \left\{ \rho \in C^\infty(M) \,:\, \rho \geq 0, \int_M \rho \, dx = 1, \ \int_M |x|^2 \rho \, dx < \infty \right\}, \tag{3.1}$$

and its tangent space at $\rho \in \mathcal{P}(M)$,

$$T_\rho \mathcal{P}(M) = \left\{ \sigma \in C^\infty(M) : \int_M \sigma \, dx = 0 \right\}. \tag{3.2}$$

We introduce the tangent and cotangent bundles of $\mathcal{P}$ and denote them respectively by

$$\mathcal{T}\mathcal{P} = \bigcup_{\rho \in \mathcal{P}} \{\rho\} \times T_\rho \mathcal{P}, \qquad \mathcal{T}^*\mathcal{P} = \bigcup_{\rho \in \mathcal{P}} \{\rho\} \times T_\rho^* \mathcal{P}. \tag{3.3}$$

Here for each ρ the cotangent space $T_\rho^* \mathcal{P}$ is taken as

$$C^\infty(\mathbb{R}^d)/\mathbb{R} = \{[\Phi] : \Phi \in C^\infty(\mathbb{R}^d)\},$$

where $[\Phi]$ is the equivalent class of functions that are identical to Φ up to a constant, i.e., $[\Phi] = \{\Phi + c : c \in \mathbb{R}\}$. In the following discussion, we always write the equivalent class $[\Phi]$ as Φ for simplicity. It is clear that $\nabla \phi = \nabla \Phi$ for any $\phi \in [\Phi]$; thus we also denote $\nabla[\Phi]$ as $\nabla \Phi$ for convenience.

The space $\mathcal{P}(M)$ becomes a metric space when equipped with the Wasserstein distance. For any $\rho_1, \rho_2 \in \mathcal{P}(M)$, the 2-Wasserstein distance (we call it Wasserstein distance for short hereafter) between ρ_1 and ρ_2 is given by [49]

$$W_2(\rho_1, \rho_2) = \left(\inf_{\pi \in \Pi(\rho_1, \rho_2)} \iint |x - y|^2 d\pi(x, y) \right)^{1/2},$$

where $\Pi(\rho_1, \rho_2)$ is the set of joint distributions on $\mathbb{R}^d \times \mathbb{R}^d$ with ρ_1 and ρ_2 as the marginals. This distance naturally induces a metric on $\mathcal{P}(M)$. In fact, for any $\rho \in \mathcal{P}(M)$ and $\sigma \in T_\rho \mathcal{P}(M)$, let us denote $\Delta_\rho := \nabla \cdot (\rho \nabla)$ and let $\Delta_\rho^\dagger$ be its pseudoinverse operator, i.e., $\Phi = (-\Delta_\rho)^\dagger \sigma$ implies $\sigma = -\Delta_\rho \Phi$.

It is shown that Φ is unique up to a constant for any given σ [12]. Then the Wasserstein metric is defined by $g^W(\rho)(\cdot, \cdot) : T_\rho \mathcal{P}(M) \times T_\rho \mathcal{P}(M) \to \mathbb{R}$ as follows:

$$g^W(\rho)(\sigma_1, \sigma_2) = \int_M \sigma_1(x)(-\Delta_\rho)^\dagger \sigma_2(x) \, dx = \int_M \nabla \Phi_1(x) \cdot \nabla \Phi_2(x) \rho(x) \, dx, \tag{3.4}$$

where $-\Delta_\rho \Phi_i(x) = -\nabla \cdot (\rho(x) \nabla \Phi_i(x)) = \sigma_i(x)$ for any $x \in M$ and $i = 1, 2$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

It is known that, equipped with the Wasserstein distance, the density manifold $\mathcal{P}(M)$ becomes a Riemannian manifold on which various differential operators and geometric flows can be established. In particular, WHF is derived by considering the variational problem

$$\mathcal{I}(\rho) = \inf_{\rho} \left\{ \int_0^T \mathcal{L}(\rho, \partial_t \rho)\, dt : \rho(t,\cdot) = \rho_0(\cdot),\ \rho(T,\cdot) = \rho_T(\cdot) \right\}, \tag{3.5}$$

where $\mathcal{L}(\rho, \partial_t \rho) = \frac{1}{2} g^W(\rho)(\partial_t \rho, \partial_t \rho) - \mathcal{F}(\rho)$ is a functional defined on $\mathcal{TP}$ known as the Lagrangian, and ρ_0 and ρ_T are some given initial and terminal densities, respectively. The solution of (3.5) satisfies the Euler–Lagrange equation, which can be reformulated as the system of first-order PDEs (1.1) given in the following proposition [12, Proposition 1].

PROPOSITION 3.1 (Proposition 1 of [12], Hamiltonian flow in dual coordinates). *Consider the dual* $\Phi = (-\Delta_\rho)^\dagger \partial_t \rho \in \mathcal{T}_\rho^* \mathcal{P}$*; the Euler–Lagrange equation to* (3.5) *is equivalent to* (1.1) *which can be treated as a Hamiltonian system on* $\mathcal{T}^* \mathcal{P}$*.*

WHF (1.1) describes the evolution of density ρ as a function of space and time. This can be viewed as the Eulerian formulation if using the language of classical fluid mechanics. Likewise, the dynamics can be written in the Lagrangian formulation, which describes the particle motion, i.e., the evolution of particle position, $\boldsymbol{X}$ as a function of time t. $\boldsymbol{X}$ is a random variable whose distribution follows the density ρ governed by (1.1). The connections between two formulations are summarized in the following proposition.

PROPOSITION 3.2 (Proposition 2 of [12], WHF as density transition equation). *Let* $(X(t))_{0 \le t < t_0}$ *be a random process in* $\mathbb{T}^d$ *with density* ρ*. Suppose* $X(t)$ *satisfies*

$$\begin{aligned} &\frac{d^2}{dt^2} X(t) = -\nabla \frac{\delta}{\delta \rho} \mathcal{F}(\rho_t(X), X) \quad \text{for any } X_0 \in \mathbb{T}^d, \\ &\dot{X}(0) = \nabla \Phi_0(0, X_0). \end{aligned} \tag{3.6}$$

Then the density $\rho(t,\cdot)$ *of* X *is a solution of the WHF* (1.1)*.*

For simplicity, this proposition was presented with periodic boundary condition or the underlying manifold being $\mathbb{T}^d$ to avoid extra boundary formulations. It can be generalized to other manifold with nominal changes. By introducing a new momentum variable $v(t, X)$, (3.6) is converted into a system of first-order equations:

$$\frac{d}{dt} X = v(t, X), \quad X(0) = X_0, \tag{3.7a}$$

$$\frac{d}{dt} v(t, X) = -\nabla \frac{\delta}{\delta \rho} \mathcal{F}(\rho_t(X), X), \quad v(0, X_0) = \nabla \Phi(0, X_0). \tag{3.7b}$$

For convenience, we call the system (3.7) the *particle* WHF.

3.2. Parameterized WHF. As one of the main goals of this paper, we introduce the PWHF in this subsection. The adopted strategy is to project the Lagrangian $\mathcal{L}$ in (3.5) onto a parameter space defined by the push-forward maps, and then derive the corresponding Euler–Lagrange equation in the parameter space.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3.2.1. Parameter space defined by push-forward maps. Let $T : \mathbb{R}^d \to \mathbb{R}^d$ be a measurable map, also called a push-forward map in $\mathbb{R}^d$. Fixing any reference probability distribution λ that is absolutely continuous with respect to the standard Lebesgue measure μ on $\mathbb{R}^d$, we use $\varrho = d\lambda/d\mu$, the Radon–Nikodym deriviative of λ with respect to μ, to denote the reference density determined by λ. Then T induces a new probability density $T_\sharp \varrho$ on $\mathbb{R}^d$:

$$\begin{aligned}\int_E T_\sharp \varrho(x)\, d\mu(x) &= \int_{T^{-1}(E)} \varrho(z)\, d\mu(z) = \int_{T^{-1}(E)} d\lambda(z) \\ &= \lambda(T^{-1}(E)) \quad \text{for all measurable } E \subset \mathbb{R}^d,\end{aligned}$$

where $T^{-1}(E)$ is the preimage of E. Hereafter we use dx instead of $d\mu(x)$ to reduce notation complexity.

Let us take T as a parameterized map, namely for any $\theta \in \Theta$, $T_\theta : \mathbb{R}^d \to \mathbb{R}^d$ is a parametric function with parameter θ. Here Θ, as a subset of $\mathbb{R}^m$, is called the *parameter space,* where m is the number of parameters of T_θ (i.e., the dimension of θ). Typical examples of T_θ include Fourier expansion, finite element approximation, and (deep) neural networks.

The map $T_{\cdot\sharp} : \Theta \to \mathcal{P}$ given by $\theta \mapsto T_{\theta\sharp}\varrho$ naturally defines an immersion from Θ to the probability manifold $\mathcal{P}$. Collecting all parameterized distributions together, i.e.,

$$\mathcal{P}_\Theta = \{\rho_\theta = T_{\theta\sharp}\varrho \,:\, \theta \in \Theta\},$$

we obtain a finite-dimensional submanifold $\mathcal{P}_\Theta$ of $\mathcal{P}$. We can define the tangent space of $\mathcal{P}_\Theta$ at each $\theta = (\theta_1, \ldots, \theta_m) \in \Theta \subset \mathbb{R}^m$ as $T_{\rho_\theta}\mathcal{P}_\Theta = \operatorname{span}\{\frac{\partial \rho_\theta}{\partial \theta_1}, \ldots, \frac{\partial \rho_\theta}{\partial \theta_m}\}$. The tangent bundle is then $\mathcal{T}\mathcal{P}_\Theta = \cup_{\theta\in\Theta}\{\rho_\theta\} \times T_{\rho_\theta}\mathcal{P}_\Theta$. On the other hand, the cotangent space $T^*_{\rho_\theta}\mathcal{P}_\Theta$ is the dual space of $T_{\rho_\theta}\mathcal{P}_\Theta$, and the cotangent bundle is $\mathcal{T}^*\mathcal{P}_\Theta = \cup_{\theta\in\Theta}\{\rho_\theta\} \times T^*_{\rho_\theta}\mathcal{P}_\Theta$.

A counterpart to the Wasserstein metric defined on $\mathcal{P}$ can be introduced on the parameter space $\Theta \in \mathbb{R}^m$ by using the pullback operator through T_θ, i.e., $G(\theta) = {T_{\theta\sharp}}^* g^W$, where g^W is the Wasserstein metric tensor given in (3.4). This is the pullback Wasserstein metric on the parameter space. It turns out that $G(\theta)$ is an $m \times m$ positive semidefinite matrix which defines a bilinear form on the tangent space of Θ at θ, $\mathcal{T}_\theta\Theta \simeq \mathbb{R}^m$ (rigorously, $\mathcal{T}_\theta\Theta$ is a subspace of $\mathbb{R}^m$ depending on the choice of T_θ, as addressed in Remark 3.5). For any $\theta \in \Theta$ and $\xi_1, \xi_2 \in \mathcal{T}_\theta\Theta$, we have

$$G(\theta)(\xi_1, \xi_2) = g^W(\rho_\theta)((T_{\theta\sharp})_*\xi_1, (T_{\theta\sharp})_*\xi_2), \tag{3.8}$$

where $(T_{\theta\sharp})_*\xi_i$ is the tangent vector at $T_{\theta\sharp}\varrho$ on the Wasserstein manifold due to the push-forward of ξ_i by the map $T_{\theta\sharp}$ for $i = 1, 2$. It requires solving elliptic PDEs to evaluate $G(\theta)$. Interested readers are referred to section 3.1 of [40] for further details.

3.2.2. Parameterization of WHF. We introduce the parameterization of WHF in this section, which is the first contribution of this work. Our treatment is outlined in the following flowchart:

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Starting with a given Hamiltonian $\mathcal{H}(\rho, \Phi)$
$\Downarrow$
By taking Legendre transform of $\mathcal{H}$, we obtain Lagrangian $\mathcal{L}(\rho, \dot{\rho})$
$\Downarrow$
Using $\mathcal{L}$, we can define L on $\mathcal{T}\Theta$ as $L(\theta, \dot{\theta}) = \mathcal{L}((T_{\cdot\sharp}\lambda)(\theta), (T_{\theta\sharp}\varrho)_*\dot{\theta})$
$\Downarrow$
By applying Legendre transform to L, we obtain the Hamiltonian in parameter space $H(\theta, p)$
$\Downarrow$
We formulate the PWHF as
$\dot{\theta}(t) = \partial_p H(\theta(t), p(t))$,
$\dot{p}(t) = -\partial_\theta H(\theta(t), p(t))$.

Following the procedure in the above flowchart, we derive the PWHF by leveraging the perspective of the Lagrangian mechanics. To be more specific, as introduced in section 3.1, we consider the Lagrangian:

$$\mathcal{L}(\rho, \dot{\rho}) = \frac{1}{2} g^W(\dot{\rho}, \dot{\rho}) - \mathcal{F}(\rho),$$

where g^W is defined in (3.4) and $\mathcal{F}(\rho)$ takes the general form in (1.4). We define the Lagrangian L as the parameterized counterpart of $\mathcal{L}$ on $\mathcal{T}\Theta$ as

$$L(\theta, \dot{\theta}) = \mathcal{L}(T_{\theta\sharp}\varrho, (T_{\theta\sharp})_*\dot{\theta}).$$

More precisely, for any vectors $\theta \in \Theta$ and $\dot{\theta} \in \mathbb{R}^m$ (we assume all vectors are column vectors unless otherwise noted), we denote $\rho_\theta = T_{\theta\sharp}\varrho$, then $(T_{\theta\sharp})_*\dot{\theta} = \frac{\partial \rho_\theta}{\partial \theta}\dot{\theta}$, and the Lagrangian L takes the following form:

$$L(\theta, \dot{\theta}) = \mathcal{L}\left(\rho_\theta, \frac{\partial \rho_\theta}{\partial \theta}\dot{\theta}\right) = \frac{1}{2}\dot{\theta}^\top G(\theta)\dot{\theta} - F(\theta), \tag{3.9}$$

where $F(\theta) := \mathcal{F}(\rho_\theta)$ and $^\top$ stands for the transpose of a vector or matrix. Detailed calculation of (3.9) is given in section SM1.1 in the supplementary material.

THEOREM 3.3 (Euler–Lagrange equation in parameter space). *Consider the Lagrangian L defined in* (3.9) *and given initial density ρ_0 and terminal density ρ_T at some prescribed terminal time T, as well as the variational problem of a curve $\theta = \{\theta(t) \in \mathbb{R}^m : t \in [0, T]\} \subset \Theta$ and its time derivative $\dot{\theta} = \{\dot{\theta(t)} \in \mathbb{R}^m : t \in [0, T]\}$:*

$$\mathcal{I}^\Theta(\theta) = \inf_\theta \left\{ \int_0^T L(\theta(t), \dot{\theta}(t))dt : \rho_{\theta(0)} = \rho_0, \rho_{\theta(T)} = \rho_T \right\}. \tag{3.10}$$

Its Euler–Lagrange equation is the following second-order ODE:

$$G(\theta)\ddot{\theta} + \sum_{k=1}^m \dot{\theta}_k \partial_{\theta_k} G(\theta)\dot{\theta} - \frac{1}{2}[\dot{\theta}^\top \partial_{\theta_k} G(\theta)\dot{\theta}]_{k=1}^m = -\nabla_\theta F(\theta) \tag{3.11}$$

with $\dot{\theta} = [\dot{\theta}_k]_{k=1}^m$. Here $[a_k]_{k=1}^m$ represents an m-dimensional vector with a_k as its kth component. From (3.11) *and hereafter, we drop (t) and just write θ and $\dot{\theta}$ to indicate*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$\theta(t)$ and $\dot\theta(t)$ at any time point t for notation simplicity when no danger of confusion may occur.

The proof of Theorem 3.3 follows the standard procedure in Lagrangian mechanics, and it is given in supplementary material section SM1.2.

Let us assume that $G(\theta)$ is nonsingular for any $\theta \in \Theta$ (the more general case where $G(\theta)$ can be singular is discussed in Remark 3.5); then we can introduce the associated Hamiltonian via Legendre transform with any fixed θ. Specifically, we denote $\mathcal{T}^*\Theta$ the phase space (cotangent bundle) of Θ and then define $H(\cdot,\cdot): \mathcal{T}^*\Theta \to \mathbb{R}$ as

$$H(\theta,p) = \sup_{\dot\theta} \{\dot\theta^\top p - L(\theta,\dot\theta)\} = \frac{1}{2} p^\top G(\theta)^{-1} p + F(\theta). \tag{3.12}$$

Following the convention in classical mechanics [22], we introduce the momentum

$$p = \frac{\partial}{\partial \dot\theta} L(\theta,\dot\theta) = G(\theta)\dot\theta. \tag{3.13}$$

Then the Hamiltonian system associated with (3.11) can be formulated as

$$\dot\theta = \partial_p H(\theta,p) = G(\theta)^{-1} p, \tag{3.14a}$$

$$\dot p = -\partial_\theta H(\theta,p) = \frac{1}{2}[p^\top G(\theta)^{-1} \partial_{\theta_k} G(\theta) G(\theta)^{-1} p]_{k=1}^m - \nabla_\theta F(\theta). \tag{3.14b}$$

We call the ODE system (3.14) the *parameterized Wasserstein Hamiltonian flow* (PWHF).

Remark 3.4 (existence and uniqueness of PWHF). Under the assumption that $G(\theta)$ is nonsingular on Θ, one can verify that both $\frac{\partial H(\theta,p)}{\partial \theta}$ and $\frac{\partial H(\theta,p)}{\partial p}$ are locally Lipschitz. Thus by the standard ODE theory, the PWHF (3.14) must have a unique solution over a finite time interval $[0,t^*)$ for some $t^* > 0$ from any given initial value. However, determining t^* is a challenging problem due to the complex structure of T_θ and geometry of $\mathcal{P}_\Theta$. We leave this for future investigations.

Remark 3.5 (singular $G(\theta)$). In our derivation of the PWHF (3.14), the metric tensor G is assumed to be nonsingular. This assumption can be relaxed. If G is singular, the PWHF can be derived similarly with the following modifications: we restrict $\mathcal{T}_\theta\Theta = \mathcal{T}_\theta^*\Theta = \mathcal{R}(G(\theta)) \subset \mathbb{R}^m$ at each θ, where $\mathcal{R}(\cdot)$ denotes the range (i.e., column space) of its argument matrix. Then $G(\theta)$ is positive definite, hence a nondegenerate inner product, on the tangent and cotangent spaces with corresponding bundles denoted by $\{(\theta,\mathcal{T}_\theta\Theta) : \theta \in \Theta\}$ and $\{(\theta,\mathcal{T}_\theta^*\Theta) : \theta \in \Theta\}$, respectively. We can define $H(\cdot,\cdot): \mathcal{T}^*\Theta \to \mathbb{R}$ by

$$H(\theta,p) = \sup_{\dot\theta \in \mathcal{R}(G)} \{\dot\theta^\top p - L(\theta,\dot\theta)\} = \frac{1}{2} p^\top G(\theta)^\dagger p + F(\theta), \tag{3.15}$$

where $G(\theta)^\dagger$ is the Penrose–Moore pseudoinverse of $G(\theta)$. In this case, the Legendre transform is well-defined, because the maximizer $\dot\theta = G(\theta)^\dagger p$ can be attained as long as $p \in \mathcal{T}_\theta^*\Theta = \mathcal{R}(G(\theta))$, which is always true given the definition of momentum $p = G(\theta)\dot\theta \in \mathcal{R}(G(\theta))$. The resulting Hamiltonian system on (θ,p) is the same as (3.14a) and (3.14b) except for two modifications: G^{-1} is replaced by $G^\dagger$, and (3.14b) will have an additional term due to the derivative of the pseudoinverse $G(\theta)^\dagger$; see Proposition 3.12 below.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3.3. PWHF with a simplified metric. Theorem 3.3 reduces the PDE in density space to a parameterized system in finite-dimensional space, hence potentially providing a way to compute the WHF by numerical algorithms. However, the computational cost to solve (3.11), as well as (3.14), is still high. The main difficulty comes from the computation of the metric tensor G. More precisely, directly evaluating G requires solving m different elliptic PDEs where m is the number of parameters in the push-forward map T_θ, whereas m can be large if we choose T_θ to be neural networks. In [40], a bilevel minimization scheme is proposed to circumvent this challenge. By introducing several auxiliary functions, the term $G(\theta)^{-1}\theta$ is calculated as the critical point of a min-max problem. However, it may still be expensive to solve such optimization problems in general.

In this paper, we develop a new strategy by introducing a simplified metric $\widehat{G}$ and use it to replace G in the derivation. This new metric not only yields much simpler implementation and more effective computations, but also enables us to establish a theoretical estimate in the Wasserstein metric to quantify the error of the approximation. Furthermore, our investigation shows that both the computation and the theory can be extended to the general case where $\widehat{G}$ is positive semidefinite with constant rank but not necessarily invertible. Numerical results also demonstrate excellent approximation accuracy of this new metric, as we will show below.

DEFINITION 3.6 (simplified pullback Wasserstein metric in Θ). *Letting T_θ be the push-forward map and λ be the reference distribution, we define the simplified pullback Wasserstein metric on $\mathcal{P}_\Theta$ as*

$$\widehat{G}(\theta) = \int \partial_\theta T_\theta(z)^\top \partial_\theta T_\theta(z)\, d\lambda(z). \tag{3.16}$$

It is worth mentioning that this definition is directly inspired by $G(\theta)$. A more in-depth motivation is influenced by the work of Otto [41], where the Wasserstein metric is defined through an isometric submersion from the space of push-forward operators $\mathcal{O}$ onto the Wasserstein manifold of density $\mathcal{P}(\mathbb{R}^d)$. Here by the space of push-forward operators we mean that $\mathcal{O}$ is the set of smooth transformations T on $\mathbb{R}^d$. Let us consider the map $\mathscr{T} : \Theta \ni \theta \mapsto T_\theta \in \mathcal{O}$. We define the pullback metric on Θ as

$$\widehat{G}(\theta) = \mathscr{T}^* g_{L^2(\lambda)},$$

where $\mathscr{T}^*$ is the pullback operation induced by $\mathscr{T}$ and $g_{L^2(\lambda)}$ is the metric on the tangent space of $\mathcal{O}$. To evaluate $\widehat{G}(\theta)$, we consider any curve $\{\theta(t)\}_{-\epsilon\le t\le\epsilon}$ in Θ, and denote $\dot\theta(0) = \frac{d}{dt}\theta(t)|_{t=0}$. By the definition of pullback operation, we have

$$\begin{aligned}\widehat{G}(\dot\theta(0),\dot\theta(0)) &= g_{L^2(\lambda)}\left(\frac{d}{dt}T_{\theta(t)}|_{t=0}, \frac{d}{dt}T_{\theta(t)}|_{t=0}\right)\\ &= \dot\theta(0)^\top\left(\int_{\mathbb{R}^d}\partial_\theta T_\theta(z)^\top \partial_\theta T_\theta(z)\, d\lambda(z)\right)\dot\theta(0).\end{aligned}$$

Thus we obtain (3.16). More precisely, we have $\widehat{G}(\theta) = (\widehat{G}(\theta)_{ij})_{1\le i,j\le m}$, and the (i,j)th entry is

$$\widehat{G}(\theta)_{ij} = \sum_{k=1}^{d}\int_{\mathbb{R}^d}\partial_{\theta_i}T_\theta^{(k)}(z)\cdot\partial_{\theta_j}T_\theta^{(k)}(z)\, d\lambda(z),$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

where $T_\theta^{(k)}:\mathbb{R}^d\to\mathbb{R}$ is the kth component of the mapping $T_\theta:\mathbb{R}^d\to\mathbb{R}^d$. The form of $\widehat{G}(\theta)$ indicates it as an $m\times m$ semipositive definite matrix for any $\theta\in\Theta$. It is shown that $\widehat{G}$ coincides with G when $d=1$ [40], but not necessarily so when $d>1$. The following theorems state conditions for the matrix $\widehat{G}$ to be invertible.

THEOREM 3.7 (positive definiteness of $\widehat{G}$). *The metric $\widehat{G}(\theta)$ defined in* (3.16) *is positive definite if and only if the m vectors $\{\partial_{\theta_k}T_\theta : k=1,\dots,m\}$ are linearly independent in $L^2(\mathbb{R}^d;\mathbb{R}^d,\lambda)$.*

THEOREM 3.8. *If the metric $G(\theta)$ defined in* (3.8) *is positive definite, then $\widehat{G}(\theta)$ is positive definite.*

The proof of Theorem 3.7 is trivial and hence omitted. The proof of Theorem 3.8 is provided in the supplementary material section SM1.3. However, we remark that the converse of Theorem 3.8 is not necessarily true, and a counterexample is given in section SM3.1. Replacing G by $\widehat{G}$ in the expression of L, we can establish similar results as those stated in Theorem 3.3, as given in the following theorem.

THEOREM 3.9. *Consider the variational problem*

$$\widehat{\mathcal{I}}^{\Theta}(\theta)=\inf_{\theta}\left\{\int_0^T \widehat{L}(\theta(t),\dot\theta(t))\,dt : \rho_{\theta(0)}=\rho_0,\rho_{\theta(T)}=\rho_T\right\}, \tag{3.17}$$

where $\widehat{L}$ is defined as

$$\widehat{L}(\theta,\dot\theta)=\frac{1}{2}\dot\theta^\top\widehat{G}(\theta)\dot\theta-F(\theta). \tag{3.18}$$

The Euler–Lagrange equation for the variation formulation is

$$\widehat{G}(\theta)\ddot\theta+\sum_{k=1}^m\dot\theta_k\partial_{\theta_k}\widehat{G}(\theta)\dot\theta-\frac{1}{2}[\dot\theta^\top\partial_{\theta_k}\widehat{G}(\theta)\dot\theta]_{k=1}^m=-\nabla_\theta F(\theta). \tag{3.19}$$

3.4. Error bound for the continuous time PWHF. In this subsection, we give error estimates on the continuous time dynamics (3.19). We assume that the matrix $\widehat{G}$ has a constant rank (could be less than m) over Θ. The main results are Theorem 3.23 (error bound on ρ) and Theorem 3.25 (error bound on Φ). Before doing so, we shall first express (3.19) as a Hamiltonian system.

Let us begin by recalling some properties of the pseudoinverse operator for a positive semidefinite matrix. As introduced before, $\widehat{G}(\theta)^\dagger$ is the Penrose–Moore pseudoinverse of the matrix $\widehat{G}(\theta)$. We write them as $\widehat{G}^\dagger$ and $\widehat{G}$, respectively, for notation simplicity below. The Penrose–Moore pseudoinverse operator is a well-defined, one-to-one linear mapping. In addition, by [23, Theorem 4.3], we know that if $\widehat{G}$ has constant rank, then

$$\partial_{\theta_k}\widehat{G}^\dagger=-\widehat{G}^\dagger(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger+\widehat{G}^\dagger\widehat{G}^\dagger(\partial_{\theta_k}\widehat{G})(I-\widehat{G}\widehat{G}^\dagger)+(I-\widehat{G}^\dagger\widehat{G})(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger\widehat{G}^\dagger. \tag{3.20}$$

Moreover, $\widehat{G}^\dagger$ satisfies the following estimate for any $\eta\in\mathbb{R}^m$:

$$|\widehat{G}(\theta)^\dagger\eta|\le\frac{1}{\lambda_{\min}(\widehat{G}(\theta))}|\eta|, \tag{3.21}$$

where $|\cdot|$ is the standard Euclidean norm of vectors and $\lambda_{\min}(\widehat{G}(\theta))$ is the *smallest positive eigenvalue* of $\widehat{G}(\theta)$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

3.4.1. Simplified Hamiltonian dynamics with the pseudoinverse operator. In this part, we study properties of the parameterized dynamics (3.19) in detail. From the equivalence between the Lagrange and Hamiltonian mechanics, the second-order ODE (3.19) is equivalent to a first-order Hamiltonian system with Hamiltonian

$$H(\theta,p)=\frac{1}{2}p^\top \widehat{G}^\dagger(\theta)p+F(\theta). \tag{3.22}$$

Remark 3.10. To derive the Hamiltonian system from the Lagrange mechanics, we require $\dot\theta\in\mathcal{R}(\widehat{G})$. This condition can be verified as long as the initial value $\dot\theta(0)$ lies in the range $\widehat{G}(\theta(0))$, which will be justified in Lemma 3.17 below.

In the remainder of this section, we make the following assumption.

Assumption 1. Assume that $\widehat{G}(\theta)$ is a smooth function of θ and

$$\lambda_{\min,\Theta}:=\inf_{\theta\in\Theta}\lambda_{\min}(\widehat{G}(\theta))>0,$$

where $\lambda_{\min}(\widehat{G}(\theta))$ is the smallest positive eigenvalue of matrix $\widehat{G}(\theta)$. We also assume

$$C_\Theta:=\sup_{\theta\in\Theta}\max_k\|\partial_{\theta_k}\widehat{G}(\theta)\|<\infty,$$

where $\|\partial_{\theta_k}\widehat{G}(\theta)\|$ is the standard matrix 2-norm of $\partial_{\theta_k}\widehat{G}(\theta)$.

Remark 3.11. Under Assumption 1, the smallest positive eigenvalue of $\widehat{G}(\theta)$ is a smooth function of θ, and $\lambda_{\min,\Theta}>0$ implies that $\widehat{G}(\theta)$ is constant rank; hence (3.20) holds.

PROPOSITION 3.12. *Under Assumption* 1*, the following is the Hamiltonian dynamics of* (3.22)*:*

$$\dot\theta=\widehat{G}^\dagger p, \tag{3.23a}$$

$$\dot p=\frac{1}{2}[(\widehat{G}^\dagger p)^\top(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger p]_{k=1}^m-\nabla_\theta F(\theta)-S(\theta,p), \tag{3.23b}$$

where

$$S(\theta,p)=\frac{1}{2}\left[p^\top\widehat{G}^\dagger\widehat{G}^\dagger(\partial_{\theta_k}\widehat{G})(I-\widehat{G}\widehat{G}^\dagger)p+p^\top(I-\widehat{G}^\dagger\widehat{G})(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger\widehat{G}^\dagger p\right]_{k=1}^m. \tag{3.24}$$

Moreover, (3.23) *is equivalent to* (3.19)*.*

Similarly, we can also prove that the dynamics (3.11) is equivalent to the Hamiltonian system (3.23) (with every $\widehat{G}/\widehat{G}^\dagger$ replaced by $G/G^\dagger$ in (3.23), assuming G has a constant rank on Θ). We note that the term $S(\theta,p)$ in (3.24) (again, with every $\widehat{G}/\widehat{G}^\dagger$ replaced by $G/G^\dagger$ in (3.24), assuming G has a constant rank on Θ) is also the extra term mentioned in Remark 3.5: it is due to using pseudoinverse $G^\dagger$ in place of G^{-1} when the rank of G is not full. Note that S in (3.24) vanishes if $\widehat{G}$ is invertible (similarly, if we use G in place of $\widehat{G}$ in (3.24), then S also vanishes when G is invertible). Moreover, the Hamiltonian structure guarantees the boundedness of $|\dot\theta|$, as shown in the following lemma, which further allows us to give a prior estimation on the error.

LEMMA 3.13. *Assume the potential* $F(\theta)$ *can be bounded from below, i.e.,* $F_{\min}:=\inf_{\theta\in\Theta}F(\theta)>-\infty$*. Suppose* (θ,p) *is solved from* (3.23) *with initial values* $(\theta(0),p(0))$*,*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

and we denote $H_0 := H(\theta(0), p(0))$. *Under Assumption* 1, $|\dot{\theta}|$ *can be uniformly upper bounded by*

$$|\dot{\theta}| \le \sqrt{\frac{2(H_0 - F_{\min})}{\lambda_{\min,\Theta}}}. \tag{3.25}$$

Proof. Since the value of the Hamiltonian $H(\theta, p)$ is conserved for any time t when (θ, p) solves the Hamiltonian system (3.23), we have

$$\frac{1}{2}\dot{\theta}^\top \widehat{G}(\theta)\dot{\theta} + F(\theta) = H(\theta(0), p(0)) = H_0 \quad \text{for any } t \ge 0. \tag{3.26}$$

By (3.21) and the fact that $\dot{\theta} = \widehat{G}(\theta)^\dagger p \in \mathcal{R}(\widehat{G})$, we have

$$\frac{1}{2}\dot{\theta}^\top \widehat{G}(\theta)\dot{\theta} + F(\theta) \ge \frac{1}{2}\lambda_{\min}(\widehat{G}(\theta))|\dot{\theta}|^2 + F_{\min} \ge \frac{1}{2}\lambda_{\min,\Theta}|\dot{\theta}|^2 + F_{\min}. \tag{3.27}$$

Combining (3.26) and (3.27) yields (3.25). ∎

3.4.2. Parameterization of the potential energy. To obtain the error estimates, we need to express the potential energy in terms of the push-forward map, which is given in this subsection. If we have $\rho = T_\sharp \varrho$ for some reference density ϱ and push-forward map T, the connection between ρ and T is known to be explicitly given by

$$\rho(x) = (\varrho \circ T^{-1})(x) \det\left(\frac{d}{dx} T^{-1}(x)\right). \tag{3.28}$$

Considering the parameterized push-forward map T_θ as well as the density function $\rho_\theta = T_{\theta\sharp}\varrho$, we can verify

$$\nabla_\theta F(\theta) = \int \partial_\theta T_\theta(z)^\top \nabla \frac{\delta}{\delta \rho} \mathcal{F}(T_{\theta\sharp}\varrho(\cdot), \cdot) \circ T_\theta(z)\, d\lambda(z). \tag{3.29}$$

Details of the derivation can be found in supplementary material section SM1.4.

3.4.3. Simplification of PWHFs. In this subsection, we show that the Hamiltonian system (3.23) can be simplified even when $\widehat{G}$ is not invertible. Specifically, we shall show that $\widehat{G}(\theta(t))\dot{\theta}(t) - p(t)$ remains zero for all t as long as its initial value is zero at $t = 0$. In this case, we can also show that the term $S(\theta(t), p(t)) = 0$ for all t. To this end, we need to investigate properties of the metric $\widehat{G}$. From its definition (3.16), we can see that $\widehat{G}$ is the inner product matrix for the functions $\{\partial_{\theta_k} T_\theta : k = 1, \dots, m\}$ in the $L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$ space. We define $\mathcal{Q}^\theta := \operatorname{span}\{\partial_{\theta_k} T_\theta(\cdot) : k = 1, \dots, m\}$ to be the subspace of $L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$. The following lemma gives the orthogonal projection operator from $L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$ onto $\mathcal{Q}^\theta$.

LEMMA 3.14. *Define kernel* $K_\theta(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathcal{M}(\mathbb{R}^d)$, *where* $\mathcal{M}(\mathbb{R}^d)$ *is the space of* $d \times d$ *matrices,*

$$K_\theta(z', z) = \partial_\theta T_\theta(z') \widehat{G}^\dagger(\theta) \partial_\theta T_\theta(z)^\top, \tag{3.30}$$

and the linear operator $\mathcal{K}_\theta$ *on* $L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$ *as*

$$\mathcal{K}_\theta[f](\cdot) := \partial_\theta T_\theta(\cdot) \widehat{G}^\dagger(\theta) \int \partial_\theta T_\theta(z)^\top f(z)\, d\lambda(z) = \int K_\theta(\cdot, z) f(z)\, d\lambda(z), \tag{3.31}$$

where $f \in L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$. *Then the operator* $\mathcal{K}_\theta$ *is the orthogonal projection from* $L^2(\mathbb{R}^d; \mathbb{R}^d, \lambda)$ *onto* $\mathcal{Q}^\theta$.

The following proposition is a direct result of Lemma 3.14.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

PROPOSITION 3.15. *For any $f \in L^2(\mathbb{R}^d;\mathbb{R}^d,\lambda)$, we have*

$$\int \partial_\theta T_\theta(z)^\top f(z)\, d\lambda(z) \in \mathcal{R}(\widehat{G}). \tag{3.32}$$

Lemma 3.14 and Proposition 3.15 can be proved through standard linear algebra arguments, and we put the details in sections SM1.5 and SM1.6. With Proposition 3.15 and the integral form (3.29) of $\nabla_\theta F(\theta)$, we conclude with the following proposition.

PROPOSITION 3.16. *For energy $\mathcal{F}$ with smooth L^2 first variation, we always have $\nabla_\theta F(\theta) \in \mathcal{R}(\widehat{G}(\theta))$.*

Under Assumption 1, $\{\theta(t)\}$ determined by (3.23) is a C^2 curve on Θ; hence we can define the following function $\Gamma^\theta(\cdot,t) \in L^2(\lambda)$ associated to θ:

$$\Gamma^\theta(z,t) = \sum_{k=1}^m \dot\theta_k \partial_{\theta_k}\partial_\theta T_\theta(z)\dot\theta + \partial_\theta T_\theta(z)\ddot\theta. \tag{3.33}$$

The following lemma provides an explicit expression on the time derivative of $\widehat{G}(\theta)\dot\theta - p$, and the proof can be found in section SM1.7.

LEMMA 3.17. *If (θ,p) solves the parameterized system* (3.23)*, then there is*

$$\frac{d}{dt}[\widehat{G}(\theta)\dot\theta - p] = \int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z) + S(\theta,p). \tag{3.34}$$

We are ready to establish the estimation on the magnitude of $\widehat{G}(\theta(t))\dot\theta(t) - p(t)$.

THEOREM 3.18. *Let (θ,p) be the solution of* (3.23) *and denote $r(t) := \widehat{G}(\theta(t))\dot\theta(t) - p(t)$ and $R(t) := |r(t)|^2$; then there is*

$$R(t) \le R(0)e^{C_S t}, \tag{3.35}$$

where $C_S := 2\sqrt{2m(H_0 - F_{\min})}C_\Theta \lambda_{\min,\Theta}^{-3/2}$.

Proof. Taking the time derivative of $R(t)$, we obtain (we temperorily write $\widehat{G}(\theta)$ and $\widehat{G}(\theta)^\dagger$ as $\widehat{G}$ and $\widehat{G}^\dagger$, respectively, for notation simplicity in this proof)

$$\begin{aligned}
\frac{d}{dt}R(t) &= 2\left\langle r(t), \frac{d}{dt}r(t)\right\rangle \\
&= 2\left\langle \widehat{G}\dot\theta - p, \int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z) + S(\theta,p)\right\rangle \\
&= 2\left\langle \widehat{G}\dot\theta - p, \int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z)\right\rangle \\
&\quad + 2\langle \widehat{G}\dot\theta - p, S(\theta,p)\rangle.
\end{aligned} \tag{3.36}$$

By Proposition 3.15, we know that $\int \partial_\theta T_\theta(z)^\top[\Gamma^\theta(z,t) + \nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)]\, d\lambda(z) \in \mathcal{R}(\widehat{G}(\theta))$. Due to the property of a pseudoinverse operator, $\widehat{G}(\theta(t))\dot\theta(t) - p(t)$ is orthogonal to the subspace $\mathcal{R}(\widehat{G}(\theta))$; hence the first inner product on the right-hand side of (3.36) vanishes. As a result, we can simply write

$$\frac{d}{dt}R(t) = 2\langle \widehat{G}\dot\theta - p, S(\theta,p)\rangle. \tag{3.37}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

By Lemma 3.13, we know that $\dot{\theta}$ is uniformly bounded, so we can show that

$$\left|[p^\top \widehat{G}^\dagger \widehat{G}^\dagger (\partial_{\theta_k}\widehat{G})(I-\widehat{G}\widehat{G}^\dagger)p]_{k=1}^m\right| = \left|[\dot{\theta}\widehat{G}^\dagger(\partial_{\theta_k}\widehat{G})(p-\widehat{G}\dot{\theta})]_{k=1}^m\right| \le \frac{\sqrt{m}|\dot{\theta}|}{\lambda_{\min}(\widehat{G})}\max_k \|\partial_{\theta_k}\widehat{G}\|\cdot|\widehat{G}\dot{\theta}-p| \le \frac{1}{2}C_S\cdot|\widehat{G}\dot{\theta}-p|.$$

Similarly, we can also bound $|[p^\top(I-\widehat{G}^\dagger\widehat{G})(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger\widehat{G}^\dagger p]_{k=1}^m|$ by $\frac{1}{2}C_S\cdot|\widehat{G}\dot{\theta}-p|$. These two bounds and the definition of $S(\theta,p)$ in (3.24) yield

$$|S(\theta,p)| \le \frac{1}{2}C_S\cdot|\widehat{G}\dot{\theta}-p|. \tag{3.38}$$

Combining (3.37) and (3.38), we get

$$\frac{d}{dt}R(t) \le 2|\widehat{G}\dot{\theta}-p|\cdot|S(\theta,p)| \le C_S\cdot|\widehat{G}\dot{\theta}-p|^2 = C_S R(t).$$

Applying Grönwall's inequality yields (3.35). ∎

A consequence of Theorem 3.18 is that $\widehat{G}(\theta(t))\dot{\theta}(t)-p(t)=0$ for all t as long as $p(0)=\widehat{G}(\theta(0))\dot{\theta}(0)$, which implies $R(0)=0$. This result is summarized in the following proposition, whose proof is omitted.

PROPOSITION 3.19. *Under the same assumption as in Theorem* 3.18*, and setting* $p(0)=\widehat{G}(\theta(0))\dot{\theta}(0)$ *for any* $\dot{\theta}(0)\in\mathbb{R}^m$*, system* (3.23) *is equivalent to*

$$\dot{\theta} = \widehat{G}^\dagger p, \tag{3.39a}$$

$$\dot{p} = \frac{1}{2}[(\widehat{G}^\dagger p)^\top(\partial_{\theta_k}\widehat{G})\widehat{G}^\dagger p]_{k=1}^m - \nabla_\theta F(\theta). \tag{3.39b}$$

We denote the ODE system (3.39) as parameterized Wasserstein Hamiltonian flow with simplified metric, or PWHF.

3.4.4. Error analysis on the PWHF. In section 3.1, we discussed the particle WHF. Here we consider its counterpart for the PWHF, i.e., the particle level dynamics induced by our parameterized dynamics in parameter space:

$$\ddot{\boldsymbol{Y}}^\Theta = \frac{d}{dt}\left(\partial_\theta T_\theta(z_0)\dot{\theta}\right) = \sum_{k=1}^m \dot{\theta}_k\partial_{\theta_k}\partial_\theta T_\theta(z)\dot{\theta} + \partial_\theta T_\theta(z)\ddot{\theta} = \Gamma^\theta(T_\theta^{-1}(\boldsymbol{Y}^\Theta),t), \tag{3.40}$$

where θ satisfies the ODE (3.39). This Lagrangian perspective enables us to carry out the error analysis results of PWHF. To be more specific, for a fixed initial position z_0, we first estimate the difference between the vector fields that drive $\boldsymbol{X}$ and $\boldsymbol{Y}^\Theta$. The result leads to an estimation on the l^2 distance between $\boldsymbol{X}$ and $\boldsymbol{Y}^\Theta$ for $t>0$. To achieve this goal, we introduce several useful lemmas. Here we assume the function $\frac{\delta}{\delta\rho}\mathcal{F}$ to be Lipschitz continuous in the $L^2(\mathbb{R}^d;\mathbb{R}^d,\lambda)$ sense as follows.

Assumption 2. *There exists a constant* $C_{\mathcal{F}}$ *such that for any two push-forward maps* T *and* $\tilde{T}$ *there is*

$$\begin{aligned}&\int\left|\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_\sharp\lambda(\cdot),\cdot)\circ T(z)-\nabla\frac{\delta}{\delta\rho}\mathcal{F}(\tilde{T}_\sharp\lambda(\cdot),\cdot)\circ\tilde{T}(z)\right|^2 d\lambda(z)\\ &\le C_{\mathcal{F}}\int|T(z)-\tilde{T}(z)|^2\,d\lambda(z).\end{aligned} \tag{3.41}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Remark 3.20. Assumption 2 can be verified if $\nabla \frac{\delta}{\delta \rho}\mathcal{F}$ is Lipschitz. As a special case, for the linear potential $\mathcal{F}(\rho) = \int V(x)\, d\rho(x)$, Assumption 2 holds true if ∇V is Lipschitz continuous.

LEMMA 3.21 (second-order dynamic of $\boldsymbol{Y}^\Theta$). *Let $\{\theta\}$ be a C^2 curve on Θ. Assume*

$$\ddot{\boldsymbol{Y}}^\Theta(t) = \Gamma^\theta(T^{-1}_{\theta(t)}(\boldsymbol{Y}^\Theta(t)), t), \quad \boldsymbol{Y}^\Theta(0) = T_{\theta(0)}(z_0), \quad \dot{\boldsymbol{Y}}^\Theta(0) = \partial T_{\theta(0)}(z_0)\dot{\theta}(0) \tag{3.42}$$

admits a unique solution for any $z_0 \in \mathbb{R}^d$. Then $\boldsymbol{Y}^\Theta(t) = T_{\theta(t)}(z_0)$ for $t \geq 0$.

Proof of Lemma 3.21. We denote $\boldsymbol{X}^\Theta(t) = T_{\theta(t)}(z_0)$. Taking the time derivative of $\boldsymbol{X}^\Theta$ twice, we obtain

$$\ddot{\boldsymbol{X}}^\Theta = \frac{d}{dt}\left(\partial_\theta T_{\theta(t)}(z_0)\dot{\theta}(t)\right) = \Gamma^\theta(T^{-1}_{\theta(t)}(\boldsymbol{X}^\Theta), t). \tag{3.43}$$

We can verify that $\boldsymbol{X}^\Theta(0) = T_{\theta(0)}(z_0)$ and $\dot{\boldsymbol{X}}^\Theta(0) = \partial_\theta T_{\theta(0)}(z_0)\dot{\theta}(0)$. According to the uniqueness of the ODE solution, we know $\boldsymbol{Y}^\Theta(t) = \boldsymbol{X}^\Theta(t) = T_{\theta(t)}(z_0)$ for any $t \geq 0$. ∎

The following lemma decomposes the dynamics (3.42) into three parts, which are used to estimate the difference between the particle level dynamics (3.7) and (3.42).

LEMMA 3.22. *Under the same assumptions as in Theorem* 3.18 *with initial $p(0) = \widehat{G}(\theta(0))\dot{\theta}(0)$, we can decompose the second-order particle level dynamics* (3.42) *as*

$$\begin{aligned}
\ddot{\boldsymbol{Y}}^\Theta &= \left(\Gamma^\theta(T_\theta^{-1}(\boldsymbol{Y}^\Theta), t) - \mathcal{K}_\theta[\Gamma^\theta(\cdot, t)](T_\theta^{-1}(\boldsymbol{Y}^\Theta))\right) \\
&\quad + \left(\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}^\Theta), \boldsymbol{Y}^\Theta) - \mathcal{K}_\theta\Big[\frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot), \cdot) \circ T_\theta\Big](T_\theta^{-1}(\boldsymbol{Y}^\Theta))\right) \\
&\quad - \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}^\Theta), \boldsymbol{Y}^\Theta),
\end{aligned} \tag{3.44}$$

which can also be written as a second-order system

$$\begin{aligned}
\dot{\boldsymbol{Y}}^\Theta &= \boldsymbol{P}^\Theta, \\
\dot{\boldsymbol{P}}^\Theta &= (Id - \mathcal{K}_\theta)\left[\Gamma^\theta(\cdot, t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot), \cdot) \circ T_\theta(\cdot)\right](T_\theta^{-1}(\boldsymbol{Y}^\Theta)) \\
&\quad - \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}^\Theta), \boldsymbol{Y}^\Theta).
\end{aligned} \tag{3.45}$$

Proof. We can rewrite $\ddot{\boldsymbol{Y}}^\Theta$ as

$$\begin{aligned}
\ddot{\boldsymbol{Y}}^\Theta &= \left(\Gamma^\theta(T_\theta^{-1}(\boldsymbol{Y}^\Theta), t) - \mathcal{K}_\theta[\Gamma^\theta(\cdot, t)](T_\theta^{-1}(\boldsymbol{Y}^\Theta))\right) \\
&\quad + \left(\mathcal{K}_\theta\left[\Gamma^\theta(\cdot, t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot), \cdot) \circ T_\theta\right](T_\theta^{-1}(\boldsymbol{Y}^\Theta))\right) \\
&\quad + \left(\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}^\Theta), \boldsymbol{Y}^\Theta) - \mathcal{K}_\theta\left[\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot), \cdot) \circ T_\theta\right](T_\theta^{-1}(\boldsymbol{Y}^\Theta))\right) \\
&\quad - \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}^\Theta), \boldsymbol{Y}^\Theta).
\end{aligned} \tag{3.46}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

By Lemma 3.17, we know that

$$\int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z) = \frac{d}{dt}[\widehat{G}(\theta)\dot{\theta} - p] - S(\theta,p).$$

By Theorem 3.18, we have that both $\frac{d}{dt}[\widehat{G}(\theta)\dot{\theta} - p]$ and $S(\theta,p)$ equal to zero. Thus,

$$\int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z) = 0.$$

Then we can compute

$$\begin{aligned}
&\mathcal{K}_\theta\left[\Gamma^\theta(\cdot,t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta\right](T_\theta^{-1}(\boldsymbol{Y}^\Theta)) \\
&= \partial_\theta T_\theta(T_\theta^{-1}(\boldsymbol{Y}^\Theta))\widehat{G}^\dagger \int \partial_\theta T_\theta(z)^\top \left[\Gamma^\theta(z,t) + \nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)\right] d\lambda(z) = 0.
\end{aligned}$$

Plugging the above identity into (3.46), we obtain (3.44). ∎

To measure the first two terms in (3.44), we introduce two quantities that characterize the approximation power of the push-forward map T_θ:

$$\begin{aligned}
\delta_0 &= \sup_{\theta\in\Theta}\min_{\zeta\in\mathcal{T}_\theta^*\Theta}\left\{\int \left|\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z) - \partial_\theta T_\theta(z)\zeta\right|^2 d\lambda(z)\right\} \\
&= \sup_{\theta\in\Theta}\left\{\int \left|\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z) - \mathcal{K}_\theta\left[\nabla \frac{\delta}{\delta \rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta\right](z)\right|^2 d\lambda(z)\right\}
\end{aligned} \tag{3.47}$$

and

$$\begin{aligned}
\delta_1 &= \sum_{1\le i,j\le m}\sup_{\theta\in\Theta}\min_{\zeta\in\mathcal{T}_\theta^*\Theta}\left\{\int |\partial_{\theta_i}\partial_{\theta_j}T_\theta(z) - \partial_\theta T_\theta(z)\zeta|^2 d\lambda(z)\right\} \\
&= \sum_{1\le i,j\le m}\sup_{\theta\in\Theta}\left\{\int |\partial_{\theta_i}\partial_{\theta_j}T_\theta(z) - \mathcal{K}_\theta[\partial_{\theta_i}\partial_{\theta_j}T_\theta](z)|^2 d\lambda(z)\right\}.
\end{aligned} \tag{3.48}$$

Another quantity δ_2 measures how well the initial tangent space $\mathcal{Q}^\theta$ approximates the initial velocity:

$$\begin{aligned}
\delta_2 &= \min_{\zeta\in\mathcal{T}_\theta^*\Theta}\int \left|\nabla\Phi(0,T_{\theta(0)}(z)) - \partial_\theta T_{\theta(0)}(z)\zeta\right|^2 d\lambda(z) \\
&= \int \left|\nabla\Phi(0,T_{\theta(0)}(z)) - \mathcal{K}_{\theta(0)}[\nabla\Phi(0,T_{\theta(0)}(\cdot))](z)\right|^2 d\lambda(z).
\end{aligned} \tag{3.49}$$

Now we are ready to provide an upper bound on the difference between ρ_θ of the PWHF and ρ of the original WHF in the W_2 sense based on the values of δ_0, δ_1, and δ_2.

THEOREM 3.23 (error estimation on ρ). *Let (ρ,Φ) be the solution of WHF (1.1) with given initial value (ρ_0,Φ_0) on time interval $[0,t_0)$ for any time horizon $t_0>0$. Suppose $p(0) = \int \partial_\theta T_{\theta(0)}^\top(z)\nabla\Phi(0,T_{\theta(0)}(z))d\lambda(z)$ and (θ,p) is the solution of PWHF*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

(3.23) *with initial value* $(\theta(0), p(0))$. *Assume* $C_{\nabla\Phi_0} := \mathrm{Lip}(\nabla\Phi_0) < \infty$ *and denote* $\epsilon_\rho := W_2^2(\rho_{\theta(0)}, \rho_0)$ *as the initial approximation error. Then under Assumptions* 1 *and* 2, *there is*

$$\begin{aligned} W_2^2(\rho_{\theta(t)}, \rho_t) &\le e^{Ct}\left((1 + 2C_{\nabla\Phi_0}^2)\epsilon_\rho + 2\delta_2\right) \\ &+ \left(\frac{3\delta_0}{C} + \frac{12\delta_1(H_0 - F_{\min})^2}{C\lambda_{\min,\Theta}^2}\right)(e^{Ct} - 1) \quad \textit{for } 0 \le t < t_0, \end{aligned} \tag{3.50}$$

where $C := 2 + 3C_{\mathcal{F}}^2$, δ_0 *and* δ_1 *are defined in* (3.47) *and* (3.48), *respectively, and* H_0 *and* $F_{\min}$ *are defined in Lemma* 3.13.

Proof. Let $\boldsymbol{X}$ be the process satisfying the system (3.7), i.e., $\boldsymbol{X}$ solves the ODE

$$\begin{aligned} \ddot{\boldsymbol{X}} &= -\nabla_X \frac{\delta}{\delta\rho(t, \boldsymbol{X})}\mathcal{F}(\rho), \quad \boldsymbol{X}(0) \sim \rho_0, \\ \dot{\boldsymbol{X}}(0) &= \nabla_p H(\boldsymbol{X}(0), \nabla\Phi(0, \boldsymbol{X}(0))) = \nabla\Phi(0, \boldsymbol{X}(0)), \end{aligned} \tag{3.51}$$

where $\rho_0(\cdot)$ and $\Phi_0(\cdot) = \Phi(0, \cdot)$ are the initial conditions of the WHF. We can verify that $\mathrm{Law}(\boldsymbol{X}(t)) = \rho(t, \cdot)$ for all t.

On the other hand, for $\{\theta\}_{t\in[0,t_0)}$, we consider the vector field $\Gamma^\theta(\cdot, t)$ defined in (3.33) and another dynamic

$$\ddot{\boldsymbol{Y}}(t) = \Gamma^\theta(T_{\theta(t)}^{-1}(\boldsymbol{Y}(t)), t), \quad \boldsymbol{Y}(0) = T_{\theta(0)}(x_0), \quad \dot{\boldsymbol{Y}}(0) = \partial_\theta T_{\theta(0)(x_0)}\dot{\theta}(0), \tag{3.52}$$

where $x_0 = \boldsymbol{X}(0)$. From the definition of $p(0)$ and Proposition 3.15, we know that $p(0) \in \mathcal{R}(\widehat{G}(\theta(0)))$; hence we can apply the decomposition (3.44) for $\boldsymbol{Y}$.

Suppose the Monge map from $\rho_{\theta(0)} = T_{\theta(0)\sharp}\lambda$ to ρ_0, which exists and is unique under the 2-Wasserstein metric, is given by ω and we assume the random variables $\boldsymbol{X}, \boldsymbol{Y}$ are coupled via $\boldsymbol{X}(0) = \omega(\boldsymbol{Y}(0))$.

Now consider the expected l^2 distance between $(\boldsymbol{X}, \dot{\boldsymbol{X}})$ and $(\boldsymbol{Y}, \dot{\boldsymbol{Y}})$,

$$E(t) := \mathbb{E}|(\boldsymbol{X}, \dot{\boldsymbol{X}}) - (\boldsymbol{Y}, \dot{\boldsymbol{Y}})|^2. \tag{3.53}$$

Taking the time derivative gives us

$$\begin{aligned} \frac{d}{dt}E(t) &= 2\mathbb{E}((\boldsymbol{X}, \dot{\boldsymbol{X}}) - (\boldsymbol{Y}, \dot{\boldsymbol{Y}})) \cdot ((\dot{\boldsymbol{X}}, \ddot{\boldsymbol{X}}) - (\dot{\boldsymbol{Y}}, \ddot{\boldsymbol{Y}})) \\ &\le 2\sqrt{\mathbb{E}|(\boldsymbol{X}, \dot{\boldsymbol{X}}) - (\boldsymbol{Y}, \dot{\boldsymbol{Y}})|^2}\sqrt{\mathbb{E}|\dot{\boldsymbol{X}} - \dot{\boldsymbol{Y}}|^2 + \mathbb{E}|\ddot{\boldsymbol{X}} - \ddot{\boldsymbol{Y}}|^2} \\ &\le 2\sqrt{E(t)}\sqrt{E(t) + \mathbb{E}|\ddot{\boldsymbol{X}} - \ddot{\boldsymbol{Y}}|^2} \le 2E(t) + \mathbb{E}|\ddot{\boldsymbol{X}} - \ddot{\boldsymbol{Y}}|^2. \end{aligned} \tag{3.54}$$

From the fact that $\mathcal{K}_\theta[\partial_\theta T_\theta\dot{\theta}](z) = \partial_\theta T_\theta(z)\dot{\theta}$, we can check

$$\begin{aligned} &\mathbb{E}|\Gamma^\theta(T_\theta^{-1}(\boldsymbol{Y}), t) - \mathcal{K}_\theta[\Gamma^\theta(\cdot, t)](T_\theta^{-1}(\boldsymbol{Y}))|^2 \\ &= \int \left|\sum_{k=1}^m \dot{\theta}_k\partial_{\theta_k}\partial_\theta T_\theta(z)\dot{\theta} + \partial_\theta T_\theta(z)\ddot{\theta} - \mathcal{K}_\theta\left[\sum_{k=1}^m \dot{\theta}_k\partial_{\theta_k}\partial_\theta T_\theta(z)\dot{\theta} + \partial_\theta T_\theta(z)\ddot{\theta}\right](z)\right|^2 d\lambda(z) \\ &= \int \left|\sum_{k=1}^m \dot{\theta}_k\partial_{\theta_k}\partial_\theta T_\theta(z)\dot{\theta} - \mathcal{K}_\theta\left[\sum_{k=1}^m \dot{\theta}_k\partial_{\theta_k}\partial_\theta T_\theta(z)\dot{\theta}\right](z)\right|^2 d\lambda(z) \\ &\le |\dot{\theta}|^4 \sum_{1\le i,j\le m} \int |\partial_{\theta_i}\partial_{\theta_j} T_\theta(z) - \mathcal{K}_\theta[\partial_i\partial_{\theta_j} T_\theta](z)|^2 \, d\lambda(z) = \delta_1|\dot{\theta}|^4. \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

By the definition of δ_0, we have

$$\mathbb{E}\left|\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}),\boldsymbol{Y})-\mathcal{K}_\theta\left[\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta\right](T_\theta^{-1}(\boldsymbol{Y}))\right|^2$$
$$=\int\left|\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta(z)-\mathcal{K}_\theta\left[\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta\right](z)\right|^2 d\lambda(z)\leq\ \delta_0.$$

By Assumption 2 on the potential $\mathcal{F}(\rho)$, and noticing that $\boldsymbol{X}$ is a push-forward of $\boldsymbol{X}(0)$ through the dynamics (3.51), we have

$$\mathbb{E}\left|\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}),\boldsymbol{Y})-\nabla\frac{\delta}{\delta\rho}\mathcal{F}(\rho(\boldsymbol{X}),\boldsymbol{X})\right|^2\leq C_{\mathcal{F}}\mathbb{E}|\boldsymbol{Y}-\boldsymbol{X}|^2. \tag{3.55}$$

Combining the last three inequalities and applying the decomposition in Lemma 3.22, we obtain

$$\begin{aligned}\mathbb{E}|\ddot{\boldsymbol{X}}-\ddot{\boldsymbol{Y}}|^2=\mathbb{E}\Bigg|&\left(\Gamma^\theta(T_\theta^{-1}(\boldsymbol{Y}),t)-\mathcal{K}_\theta[\Gamma^\theta(\cdot,t)](T_\theta^{-1}(\boldsymbol{Y}))\right)\\&+\left(\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}),\boldsymbol{Y})-\mathcal{K}_\theta\left[\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\cdot),\cdot)\circ T_\theta\right](T_\theta^{-1}(\boldsymbol{Y}))\right)\\&-\left(\nabla\frac{\delta}{\delta\rho}\mathcal{F}(T_{\theta\sharp}\varrho(\boldsymbol{Y}),\boldsymbol{Y})-\nabla\frac{\delta}{\delta\rho}\mathcal{F}(\rho(\boldsymbol{X}),\boldsymbol{X})\right)\Bigg|^2\\&\leq 3\left(\delta_0+\delta_1|\dot\theta|^4+C_{\mathcal{F}}\mathbb{E}|\boldsymbol{Y}-\boldsymbol{X}|^2\right)\\&\leq 3\left(\delta_0+4\delta_1\frac{(H_0-F_{\min})^2}{\lambda_{\min,\Theta}^2}+C_{\mathcal{F}}\mathbb{E}|\boldsymbol{Y}-\boldsymbol{X}|^2\right).\end{aligned}$$

Continuing from (3.54), we compute

$$\frac{d}{dt}E(t)\leq 2E(t)+3\left(\delta_0+4\delta_1\frac{(H_0-F_{\min})^2}{\lambda_{\min,\Theta}^2}+C_{\mathcal{F}}\mathbb{E}|\boldsymbol{Y}-\boldsymbol{X}|^2\right) \tag{3.56}$$

$$\leq 3\delta_0+12\delta_1\frac{(H_0-F_{\min})^2}{\lambda_{\min,\Theta}^2}+(2+3C_{\mathcal{F}})E(t). \tag{3.57}$$

Recalling $C=2+3C_{\mathcal{F}}$ and applying Gronwall's inequality, we arrive at

$$E(t)\leq e^{Ct}E(0)+\left(\frac{3\delta_0}{C}+\frac{12\delta_1(H_0-F_{\min})^2}{C\lambda_{\min,\Theta}^2}\right)(e^{Ct}-1). \tag{3.58}$$

Now we estimate the initial error

(3.59)

$$E(0)=\mathbb{E}|\boldsymbol{X}(0)-\boldsymbol{Y}(0)|^2+\mathbb{E}|\dot{\boldsymbol{X}}(0)-\dot{\boldsymbol{Y}}(0)|^2$$

(3.60)

$$=\mathbb{E}_{z_0\sim\lambda}|\omega(T_{\theta(0)}(z_0))-T_{\theta(0)}(z_0)|^2+\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,\omega(T_{\theta(0)}(z_0)))-\partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Since ω is the Monge map from $\rho_{\theta(0)}$ to ρ_0, the term $\mathbb{E}|\omega(T_{\theta(0)}(z_0)) - T_{\theta(0)}(z_0)|^2 = W_2^2(\rho_{\theta(0)}, \rho_0) = \epsilon_\rho$. For the second term above, we have

$$\begin{aligned}
&\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,\omega(T_{\theta(0)}(z_0))) - \partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2\\
&= \mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,\omega(T_{\theta(0)}(z_0))) - \nabla\Phi(0,T_{\theta(0)}(z_0)) + \nabla\Phi(0,T_{\theta(0)}(z_0)) - \partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2\\
&\le 2\,\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,\omega(T_{\theta(0)}(z_0))) - \nabla\Phi(0,T_{\theta(0)}(z_0))\right|^2\\
&\quad + 2\,\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,T_{\theta(0)}(z_0)) - \partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2\\
&\le 2C^2_{\nabla\Phi_0} W_2^2(\rho_{\theta(0)},\rho_0) + 2\,\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,T_{\theta(0)}(z_0)) - \partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2
\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}_{z_0\sim\lambda}\left|\nabla\Phi(0,T_{\theta(0)}(z_0)) - \partial_\theta T_{\theta(0)}(z_0)\dot\theta(0)\right|^2\\
&= \mathbb{E}\left|\nabla\Phi(0,T_{\theta(0)}(z_0)) - \partial_\theta T_{\theta(0)}(z_0)\widehat{G}(\theta(0))^\dagger \int \partial_\theta T^\top_{\theta(0)}(z)\nabla\Phi(0,T_{\theta(0)}(z))d\lambda(z)\right|^2\\
&= \mathbb{E}\left|\nabla\Phi(0,T_{\theta(0)}(z_0)) - \mathcal{K}_{\theta(0)}[\nabla\Phi(0,T_{\theta(0)}(\cdot))](z_0)\right|^2 = \delta_2.
\end{aligned}$$

Thus the initial error

$$E(0) \le (1 + 2C^2_{\nabla\Phi_0})\epsilon_\rho + 2\delta_2. \tag{3.61}$$

Combining (3.58) and (3.61), we get

$$E(t) \le e^{Ct}\left((1 + 2C^2_{\nabla\Phi_0})\epsilon_\rho + 2\delta_2\right) + \left(\frac{3\delta_0}{C} + \frac{12\delta_1(H_0 - F_{\min})^2}{C\lambda^2_{\min,\Theta}}\right)(e^{Ct} - 1). \tag{3.62}$$

Since

$$W_2^2(\mathrm{Law}(\boldsymbol{Y}), \mathrm{Law}(\boldsymbol{X})) \le \mathbb{E}|\boldsymbol{Y} - \boldsymbol{X}|^2 \le E(t),$$

and $\mathrm{Law}(\boldsymbol{Y}(t)) = \rho_{\theta(t)} = T_{\theta(t)\sharp}\lambda$ and $\mathrm{Law}(\boldsymbol{X}(t)) = \rho(t,\cdot)$, we obtain (3.50). ∎

Theorem 3.23 provides an upper bound of the approximation error of $\rho_{\theta(t)}$ to the true solution ρ_t using the simplified metric $\widehat{G}$. This bound, as shown on the right-hand side of (3.50), appears to be linearly dependent on the initial error ϵ_ρ, the quantities δ_0, δ_1, and δ_2 for any t. However, the coefficient is an exponential factor e^{Ct}, which is typical for such dynamics.

Remark 3.24. From the proof of Theorem 3.23, we can see that there is a trade-off between $\{\delta_i : i = 0,1,2\}$ and $\lambda_{\min,\Theta}$. In fact, if we choose $\widetilde{G}$ as the inner product matrix of a subspace $\widetilde{\mathcal{Q}} \subset \mathcal{Q}^\theta$, the arguments in this section still hold true. In this case, the smallest positive eigenvalue $\lambda_{\min}(\widetilde{G})$ may be larger than $\lambda_{\min}(\widehat{G})$, while the corresponding approximation errors $\{\tilde\delta_i : i = 0,1,2\}$ may be larger than the original ones. A more detailed analysis of the relationship among these quantities may serve as a future research direction.

THEOREM 3.25 (error estimation on Φ). *Denote* $\vec{u}_\Theta(t,\cdot) := \partial_\theta T_{\theta(t)} \circ T^{-1}_{\theta(t)}(\cdot)\dot\theta(t) : \mathbb{R}^d \to \mathbb{R}^d$. *If the Hamilton–Jacobi equation*

$$\frac{\partial\Phi(t,x)}{\partial t} + \frac{1}{2}|\nabla\Phi(t,x)|^2 = -\nabla\frac{\delta}{\delta\rho}\mathcal{F}(\rho(x),x), \quad \Phi(0,\cdot) = \Phi_0(\cdot), \tag{3.63}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

admits a $C^1([0,t_1)\times\mathbb{R}^d)$ *solution on a time interval* $[0,t_1)$*, then there is*

$$\int_{\mathbb{R}^d} |\vec{u}_\Theta(t,x)-\nabla\Phi(t,x)|^2\rho_{\theta(t)}(x)\,dx \le 2(1+\mathrm{Lip}(\nabla\Phi(t,\cdot)))\,\mathcal{C}(\mathcal{F},\rho_0,\Phi_0,\theta_0,p_0,T_\theta) \tag{3.64}$$

for all $0\le t<\min\{t_0,t_1\}$*, where* $\rho_{\theta(t)}=T_{\theta(t)\sharp}\lambda$*,* $\mathcal{C}(\mathcal{F},\rho_0,\Phi_0,\theta_0,\lambda,T_\theta,t)$ *is the bound on the right-hand side of* (3.50) *and depends on potential* F*, the initial values* $\rho_0,\Phi_0,\theta(0)$*,* $p(0)$*, the push-forward map* T_θ*, and time* t*.*

The proof of Theorem 3.25 is a continuation of inequality (3.50) and can be found in supplementary material section SM1.7.

Remark 3.26. The time intervals $[0,t_0)$ and $[0,t_1)$ used in Theorems 3.23 and 3.25 are determined by the singularity development of ρ and Φ in the WHF (1.1), respectively. However, we would like to highlight that the solutions of PWHF (3.23) and (3.39) may exist beyond these singularities. The same is true for the solution of the particle WHF (3.7) due to the solution existence and uniqueness of ODEs. When this happens, we may use $\boldsymbol{X}$ and v to define ρ and Φ beyond the singularity of (1.1). The error estimates obtained in both theorems still hold as long as both the solutions of (3.7) and (3.39) exist. The examples on Wasserstein geodesic and harmonic oscillators given in section 5 can illustrate this situation. In both examples, the solutions of PWHF exist on $[0,\infty)$ while finite time singularities are developed in ρ and Φ in (1.1).

3.5. Harmonic oscillator with affine transform as push-forward map. To better convey our idea on how PWHF is proposed and formulated, we present an illustrative example that have exact solutions for (3.39). Additional examples can be found in section SM3.2.1.

Let us use affine transform $T_\theta(z)=\Gamma z+b$, $\theta=(\Gamma,b)$, $z\in\mathbb{R}^d$ as the parameterized push-forward map. Here Γ is a $d\times d$ invertible matrix and b is a d dimensional vector. We consider a Hamiltonian system with Hamiltonian $H(x,v)=\frac{1}{2}|v|^2+\frac{1}{2}x^\top Ux$, where U is $d\times d$ self-adjoint and positive definite matrix. The corresponding WHF is

$$\begin{aligned}\frac{\partial\rho(t,x)}{\partial t}&=-\nabla\cdot(\rho(t,x)\nabla\Phi(t,x)),\\ \frac{\partial\Phi(t,x)}{\partial t}&=-\frac{1}{2}|\nabla\Phi(t,x)|^2-\frac{1}{2}x^\top Ux,\end{aligned}$$

with $\mathcal{F}=\int\frac{1}{2}x^\top Ux\rho(x)dx$.

We take the initial values as Gaussian distribution and quadratic function, respectively, i.e., $\rho_0=\mathcal{N}(\mu,\Sigma)$, and $\Phi(0,x)=\frac{1}{2}x^\top Mx$, where Σ is the covariance and M is a symmetric positive definite matrix. We choose the reference distribution λ to be the standard normal distribution $\mathcal{N}(0,I)$. The proposed PWHF is formulated as

$$\dot\theta=\widehat{G}(\theta)^{-1}p=p, \tag{3.65a}$$

$$\dot p=\frac{1}{2}\dot\theta^\top\nabla_\theta\widehat{G}(\theta)\dot\theta-\nabla_\theta F(\theta)=-\nabla_\theta F(\theta), \tag{3.65b}$$

$$\theta(0)=(\sqrt{\Sigma},\mu),\quad p(0)=(M\sqrt{\Sigma},M\mu). \tag{3.65c}$$

As shown in section SM3.2, the solution to the parameterized system is

$$\begin{aligned}\Gamma(t)&=(\cos(t\sqrt{U})+\sin(t\sqrt{U})\sqrt{U}^{-1}M)\sqrt{\Sigma},\\ b(t)&=(\cos(t\sqrt{U})+\sin(t\sqrt{U})\sqrt{U}^{-1}M)\mu.\end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

One can tell that the parameterized Hamiltonian flow $\{(\Gamma(t), b(t))\}$ yields the exact flow $\{\rho(t,\cdot), \Phi(t,\cdot)\}$. Figure SM1 also shows that our numerical solution matches the true solution.

4. Numerical scheme. In this section, we develop a numerical scheme to solve the PWHF (3.39) in general cases. Since (3.39) is a Hamiltonian system, it is desirable that the scheme has a symplectic structure. For clarity, we denote the iteration number of variables as superscripts in this section.

4.1. Symplectic scheme. We solve the general PWHF (3.39) following the sympletic Euler scheme [27]:

$$\frac{\theta^{l+1}-\theta^l}{h} = \nabla_p H(\theta^{l+1}, p^l) = \widehat{G}(\theta^{l+1})^\dagger p^l, \tag{4.1a}$$

$$\begin{aligned}\frac{p^{l+1}-p^l}{h} &= -\nabla_\theta H(\theta^{l+1}, p^l)\\ &= \frac{1}{2}[(\widehat{G}(\theta^{l+1})^\dagger p^l)^\top(\partial_{\theta_k}\widehat{G}(\theta^{l+1}))\widehat{G}(\theta^{l+1})^\dagger p^l]_{k=1}^m - \nabla_\theta F(\theta^{l+1}),\end{aligned} \tag{4.1b}$$

where $h>0$ is the time step size and $l\in\mathbb{N}$ is the (outer) iteration number. Note that (4.1a) is implicit in θ^{l+1}, which needs to be solved from this equation for fixed (θ^l, p^l). To this end, we employ a fixed point iteration method given by

$$\xi^{l,j+1} = \operatorname*{argmin}_\xi \left\{\frac{1}{2}\xi^\top \widehat{G}(\alpha^{l,j})\xi - \xi^\top p^l\right\}, \tag{4.2a}$$

$$\alpha^{l,j+1} = \theta^l + h\xi^{l,j+1}, \tag{4.2b}$$

for $j=0,1,2,\ldots$, where j is the (inner) iteration number of the fixed point iteration (4.2) for solving (4.1a) with fixed l. Then upon convergence as $j\to\infty$, there are

$$\alpha^{l,j}\to\theta^{l+1} \qquad \text{and} \qquad \xi^{l,j}\to\widehat{G}(\theta^{l+1})^\dagger p^l. \tag{4.3}$$

We have two choices to initialize $(\xi^{l,0}, \alpha^{l,0})$ in the above fixed point iterations (4.2). Note that we need not only $\alpha^{l,0}$ but also $\xi^{l,0}$ since (4.2a) is itself a linear least squares problem that requires an initial $\xi^{l,0}$ when being solved. The first one is to set $\alpha^{l,0} = \theta^l$ and $\xi^{l,0} = (\theta^l - \theta^{l-1})/h$, and the second choice is to set $\alpha^{l,0} = \theta^l$ and $\xi^{l,0} = \widehat{G}(\theta^l)^\dagger p^l$. The first choice utilizes information from a previous time step and is more computationally efficient, while the second choice requires solving a linear system, but generally performs better than the first one empiricially. We adopt the second choice to initialize α, ξ in all of our experiments. If $(\alpha^{l,j}, \xi^{l,j}) \to (\alpha^{l,*}, \xi^{l,*})$ as $j\to\infty$, then we set $\theta^{l+1} = \alpha^{l,*}$ and $p^{l+1} = p^l + h\frac{1}{2}(\xi^{l,*})^\top\nabla_\theta\widehat{G}(\theta^{l+1})\xi^{l,*} - h\nabla_\theta F(\theta^{l+1})$ because of (4.3).

It can be shown that the inner iteration converges if $\lambda_{\min}(\widehat{G}(\theta))$ is bounded away from 0. In our experiments, the minimization of ξ in (4.2a) is approximated by a one-step gradient descent with step size γ,

$$\xi^{l,j+1} = \xi^{l,j} - \gamma(\widehat{G}(\alpha^{l,j})\xi^{l,j} - p^l),$$

and the inner iteration (4.2) reduces to

$$\xi^{l,j+1} = \xi^{l,j} - \gamma(\widehat{G}(\theta^l + h\xi^{l,j})\xi^{l,j} - p^l), \tag{4.4}$$

which is the fixed point iteration applied to solving ξ from $J(\xi) := (\widehat{G}(\theta^l + h\xi)\xi - p^l = 0$. Since $\nabla_\xi J(\xi) = h\nabla\widehat{G}(\theta^l + h\xi)\xi + \widehat{G}(\theta^l + h\xi)$, for certain choices of T_θ (e.g., $T_\theta(z) = \Gamma z + b$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

and $z \sim \lambda = \mathcal{N}(0, I)$, or $T_\theta(z) = z + \sigma(\Gamma z + b)$, or deeper ResNet, or normalizing flows with bounded $\sigma, \sigma', \sigma''$ where σ is the activation function), we can show that there exists $M > 0$ such that $\|\nabla_\xi J(\xi)\| \leq M$ for all ξ. Then $\tau \in (0, 1/M)$ guarantees that the fixed point iteration is linearly convergent since $I - \tau J$ is a contraction.

4.2. Evaluate $\widehat{G}^\dagger p$ by solving linear system. We have determined on solving (3.39) numerically using (4.1), where the implicit update step (4.1a) is done by applying the fixed point iteration (4.4). The only question is on obtaining the initial $\alpha^{l,0} = \theta^l$ and $\xi^{l,0} = \widehat{G}(\theta^l)^\dagger p^l$ of (4.4) for each l. Thus the key computation is on finding $\widehat{G}(\theta^l)^\dagger p^l$, for which we employ the algorithm MINRES [45] as the solver.

While the simplified metric tensor $\widehat{G}$ is defined through the push-forward map via (3.16), which can be evaluated through samples, the computational cost grows fast when the number of parameters increases. Therefore, we should avoid directly computing and storing $\widehat{G}$, but instead treat it as a linear operator in the matrix-vector product (mvp). Detailed discussion on the numerical implementation of the mvp operator is included in section SM4. In this case, we essentially apply MINRES as a linear system solver which only requires mvp in each iteration. The accuracy of MINRES is determined by a user-chosen parameter called tolerence, which yields a unique solution to the linear system $\widehat{G}(\theta^l)\xi = p^l$ as the orthogonal projection of p^l onto the subspace spanned by the eigenvectors of $\widehat{G}(\theta^l)$ whose corresponding eigenvalues are greater than or equal to the tolerance parameter. This is the only place we apply MINRES in each iteration l.

4.3. Initialization. In numerical implementations of PWHF (4.1), it is important to initialize θ and p properly. The initializations of these two variables are done differently, and we provide the details as follows.

4.3.1. Initialization of $\boldsymbol{\theta}$. It is natural to initialize θ^0 such that ρ_{θ^0} is equal or close to the given initial value ρ_0. In practice, we need a number of N_θ or N_p (see Algorithm 4.1) samples of ρ_{θ^0} to estimate all relevant terms and implement the iterative scheme of PWHF in (4.1). There are different ways to achieve this. In what follows, we describe two common cases where this is done. The first method results in an exact matching $\rho_{\theta^0} = \rho_0$ for any given initial ρ_0, whereas the second one can generate θ^0 such that ρ_{θ^0} approximates ρ_0 in the sense of the Kullback–Leibler (KL) divergence, which is not exact. More methods to finding θ^0 minimizing the difference between ρ_{θ^0} and ρ_0 in the KL divergence and Stein's discrepancy can be found in [43].

In the first case, which is the one we consider in this work, we set our reference density ϱ to be the given initial ρ_0 and set the initial θ^0 such that T_{θ^0} is the identity map (this can be done, for example, by setting $T_\theta = \mathrm{Id} + f_\theta$, where Id is the identity map and f_θ is some neural network satisfying $f_{\theta^0} \equiv 0$ when the initial parameter $\theta^0 = 0$). In this case, $\rho_{\theta^0} = T_{\theta^0 \sharp}\varrho = \varrho = \rho_0$, which is exact. In many real-world applications, ρ_0 is directly given as samples, in which case we can simply use these samples because they are drawn from $\rho_{\theta^0} = \rho_0$. If the given initial ρ_0 is provided in an (possibly unnormalized) analytic form, then one can apply any standard Monte Carlo sampling approach, such as those based on the well-known Metropolis–Hasting algorithm, to draw samples from ρ_0.

In the second case, a practitioner may prefer to specify his/her own naive reference density ϱ on $\mathbb{R}^d$, such as the standard Gaussian, as long as ϱ is easy to sample from and has an analytic form. We can perform a standard maximum likelihood formulation to obtain a desired approximation to the initial as $\theta^0 = \arg\min_\theta \mathcal{D}_{\mathrm{KL}}(\rho_0 \| \rho_\theta) = \arg\max_\theta \mathbb{E}_{x \sim \rho_0}[\log \rho_\theta(x)]$. Similarly as before, we either have the samples of ρ_0 di-

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

rectly or can generate samples of ρ_0 using the Monte Carlo method when ρ_0 is given in its (possibly unnormalized) analytic form. Then we can replace $\mathbb{E}_{x\sim\rho_0}[\log\rho_\theta(x)]$ with the empirical expectation and ρ_θ with (3.28). This results in an optimization problem for network parameter θ that has been addressed by many machine learning techniques, such as [10]. In this case not only does $T_{\theta\sharp}\varrho$ have an analytic form but its samples can be trivially obtained by pushing forward the samples of ϱ using T_θ. We note that other probability distances and generative models can be used too. For example, the Wasserstein generative adversarial network [4] and many followup works were developed to learn a generative network T_θ such that $T_{\theta\sharp}\varrho$ is close to ρ_0 in the sense of 1-Wasserstein distance. In practical implementations, it is possible that the learned θ^0 has undesired properties, for example, its values have very different scales, which may cause the evolution of PWHF (4.1) to be unstable. A remedy to this issue is that we build a separate preconditioning network $\widehat{T}$ and train it to satisfy $\hat{\rho}_0 := \widehat{T}_\sharp\varrho = \rho_0$ (or approximately equal), and set a network $\widetilde{T}_\theta$ to be the same as the one in the first case with initial $\theta^0 = 0$. Then the overall push-forward map can be set to $T_\theta := \widetilde{T}_\theta \circ \widehat{T}$. Such T_θ has a trivial initial value to start and may stabilize the entire computation of PWHF (4.1). We omit further discussions as they are out of the scope of this paper.

4.3.2. Initialization of p. From the particle interpretation of Φ, we can derive

$$p^0 = \widehat{G}(\theta^0)\dot{\theta}^0 = \int_{\mathbb{R}^d} \partial_\theta T_\theta^\top(z)\nabla\Phi(0, T_\theta(z))d\lambda(z) = \mathbb{E}_{z\sim\varrho}[\partial_\theta T_\theta^\top(z)\nabla_\theta\Phi(0, T_\theta(z))]. \tag{4.5}$$

The above identity suggests to approximate p^0 using the empirical expectation on the given initial value $\Phi(0,\cdot)$. More specifically, we can draw N_p samples $\{z_i\}_{i=1}^{N_p}$ from the reference distribution λ with the associated density ϱ, and set p^0 to be $(1/N_p)\cdot\sum_{i=1}^{N_p}\nabla_\theta\Phi(0, T_\theta(z_i))$.

4.4. Overall algorithm. In summary, we suggest the numerical algorithm, Algorithm 4.1, to solve the PWHF. In Algorithm 4.1, n_{in} is the user-chosen number of inner iterations. In each iteration indexed by l, we generate the particles $\{X_1, \ddots, X_{N_\theta}\}$ (line 4) by pushing forward the initial particles $x_1, \ldots, x_{N_\theta} \sim \rho_0$ using the current T_θ. Note that this step is simple as we are considering the first case as given in section 4.3.1 where $\varrho = \rho_0$. If we consider the second case in section 4.3.1, then this step should be modified accordingly. Lines 5–10 solve the implicit update equation of θ^{l+1} in (4.1a) using the variant fixed point iteration (4.4). Specifically, line 5 computes the initial $\xi^{l,0}$ and $\alpha^{l,0}$, and lines 6–9 implement the fixed point iteration (4.4). Lines 11–12 update p^{l+1} using the explicit rule in (4.1b). Details are given in section 4.1.

5. Numerical results.

In this section, we demonstrate the performance of Algorithm 4.1 by solving several examples of WHFs with different potential energy $\mathcal{F}$. We first present some details of our experiment settings, and then show the numerical results.

5.1. Experiment settings. We first provide details of the structure of the neural network we used to represent the push-forward mapping T_θ. Then we provide some other important parameters in our implementation and test. We also discuss the strategy we used to measure the quality of the solution T_θ. Finally, we provide an empirical estimate of the error δ_0.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 4.1. Parameterized WHF solver.

Set neural network structure T_θ. Set terminal time t_1 and number of steps K. Set step size $h = t_1/K$.
Initialize θ^0 and p^0 as in sections 4.3.1 and 4.3.2, respectively.
for $l = 0, \dots, K-1$ **do**
  Set $X_i = T_{\theta^l}(x_i)$ and $x_i \sim \rho_0$ for $i = 1, \dots, N_\theta$
  Apply MINRES to solve $\xi^{l,0}$ from $\widehat{G}(\theta^l)\xi^{l,0} = p^l$. Set $\alpha^{l,0} = \theta^l$
  for $j = 1, \dots, n_{in}$ **do**
    Update $\alpha^{l,j} = \theta^l + h\xi^{l,j}$
    Update $\xi^{l,j+1} = \xi^{l,j} - \gamma(\widehat{G}(\alpha^{l,j})\xi^{l,j} - p^l)$
  end for
  Set $\theta^{l+1} = \alpha^{l,n_{in}}, \eta^{l+1} = \xi^{l,n_{in}}$
  Set $X_i = T_{\theta^{l+1}}(z_i)$ and $z_i \sim \rho_0$ for $i = 1, \dots, N_p$ and evaluate $\nabla_\theta F(\theta^{l+1})$
  Set $p^{l+1} = p^l + \frac{h}{2}[(\eta^{l+1})^\top \partial_{\theta_k} \widehat{G}(\theta^{l+1})\eta^{l+1}]_{k=1}^m - h\nabla_\theta F(\theta^{l+1})$
end for
Output: Solution $(\theta(t), p(t))$ to PWHF (3.14) and $T_{\theta(t)}$ at $t = lh$ for $l = 0, \dots, K$.

5.1.1. Structure of push-forward map. In our experiments, we use neural networks as the push-forward map T_θ unless stated otherwise. There are multiple choices for the neural networks structure to represent T_θ. We can use the invertible neural networks (e.g., normalizing flow [43], Real NVP [16], and neural ODE [10]) or noninvertible neural networks (e.g., the multilayer perceptron or ResNet [28]), each having its own advantages. Normalizing flow simplifies the computation of the log determinant of the Jacobian matrix of the map, so we can easily compute the density function. In some cases, the potential function explicitly depends on the value of density ρ_θ, and invertible neural networks provide an effective way to evaluate it. In this work, we use the residual neural network as a push-forward map if not specified:

$$T_\theta = \mathrm{Id} + f_\theta, \tag{5.1}$$

where Id is the identity map and $f_\theta : \mathbb{R}^d \to \mathbb{R}^d$ is a standard multilayer perceptron with two hidden layers, and each hidden layer contains 50 and 80 neurons for the two-dimensional (2D) and 10D examples, respectively. We take the hyperbolic tangent function as the activation since we require the second-order derivative in the computation. The bias for the output layer in f_θ is set to be none.

5.1.2. Parameter settings. In Algorithm 4.1, MINRES is applied to solve the linear system $\widehat{G}(\theta)\dot{\theta} = p$ with tolerance 3×10^{-4}. We chose this number manually, which appears to yield satisfactory accuracy in all of our experiments. In fact, the tolerence value for MINRES has a subtle effect on the current setting of Algorithm 4.1 (PWHF): a smaller tolerence may yield a more accurate solution to $\widehat{G}(\theta)\dot{\theta} = p$, but the resulting ODE $\dot{\theta} = \widehat{G}(\theta)^\dagger p$ may suffer stiffness due to the smaller truncation threshold of eigenvalues of $\widehat{G}(\theta)$ in MINRES and consequently large condition number (ratio of largest and smallest positive eigenvalues) of $\widehat{G}(\theta)^\dagger$. This may cause instability of the numerical ODE solver even with smaller step size h. Therefore, it requires calibrations between accuracy of a linear system solution and stability of the numerical ODE solver when choosing the tolerence. This is an interesting property and worth further investigation but out of the scope of the present paper. Another parameter we set

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

in Algorithm 4.1 is the iteration number n_{in} of the fixed point iteration (4.3). In all our experiments, we found that setting n_{in} to 2 or 3, sometimes even 1, is enough to generate promising results.

For all the experiments, we choose the reference distribution λ with density ϱ to be the standard Gaussian $\mathcal{N}(0, I_d)$. We take θ^0 such that $\rho_0 = T_{\theta^0\sharp}\varrho$. We set the initial θ randomly with zero mean and tiny variation, which results in T_θ being close to the identity map and ρ_0 close to the standard Gaussian. Note that we did not really choose $\theta^0 = 9$ because in that case the Jacobian is $\nabla T_{\theta^0} = \nabla f_{\theta^0} = 0$ with our choice of f_θ (a standard mulitlayer perceptron without bias). For the computation, 50,000 samples are generated from the reference distribution to evaluate the mvp. For those cases with potential energy in the form $\mathcal{F}(\rho) = \mathbb{E}_\rho[V(x)]$, we use the same sample size to evaluate $F(\theta)$. For other cases, we only use 12,000 samples for the computation of $F(\theta)$ due to our machine memory limitation.

It is possible that Hamiltonian-type PDEs develop singularity in finite time. This means that, at some time t_0, the density may become a delta function and/or Φ is no longer well-defined. Our method can handle this singularity since the push-forward is still well-defined and smooth at time t_0. See the geodesic equation (section 5.2) and harmonic oscillator (section 5.3.1) as examples.

5.1.3. Evaluation strategy. To validate the solution obtained by Algorithm 4.1 (PWHF), we measure the difference between its generated density $\rho_{\theta(t)} (= T_{\theta(t)\sharp}\varrho)$ and the true density ρ_t when the ground truth push-forward T_t is available, i.e., $\rho_t = T_{t\sharp}\rho_0$. While there are different metrics to quantify the distance/divergence between two densities, we choose to measure the difference between $\rho_{\theta(t)}$ and ρ_t by the maximal difference between the particles pushed forward by $T_{\theta(t)}$ and T_t, respectively. More precisely, we use the error defined by

$$\widehat{\epsilon}(T_\theta, T) := \max\left\{ \frac{1}{N_\theta} \sum_{i=1}^{N_\theta} |T_{\theta(lh)}(x_i) - T_{lh}(x_i)| \ : \ l = 0, 1, \ldots, K \right\}, \tag{5.2}$$

where the initials of the particles $\{x_i : i = 1, \ldots, N_\theta\}$ are samples of ρ_0.

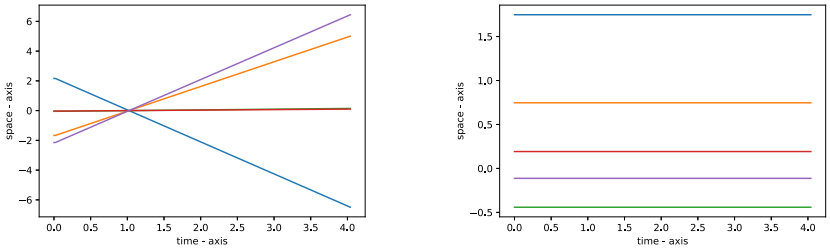
When the analytic form of T_t is not available, we can consider using a numerical ODE solver to directly push forward a number of initial particles (we use 10,000 particles in our experiments) by following the particle dynamics (3.7). Whether this is feasible or not relies on the feasibility of approximating $\nabla \frac{\delta}{\delta \rho} \mathcal{F}(\rho_t(X), X)$ using samples only, since we do not have the analytic form of ρ_t. In the case of interactive potential $\mathcal{F}$ as given in section 5.4, the variation of $\mathcal{F}$ has an integral linear in ρ_t and thus can be well approximated by the particles. We conduct our evaluation in this case. However, for more general cases, such approximations may not be feasible and alternative ways must be sought for evaluation. We will not consider those cases in the present work. Nevertheless, it does not affect the proposed Algorithm 4.1.

5.2. Geodesic equation as Wasserstein Hamiltonian flow. We first consider the geodesic equation on $\mathbb{R}^2$ as a WHF, which corresponds to the zero potential case, i.e., $\mathcal{F}(\rho) \equiv 0$. The WHF is given by

$$\begin{aligned} \frac{\partial}{\partial t}\rho + \nabla \cdot (\rho \nabla \Phi) &= 0, \\ \frac{\partial}{\partial t}\Phi + \frac{1}{2}|\nabla \Phi|^2 &= 0. \end{aligned}$$

The Wasserstein geodesic equation plays an important role in the optimal transport theory. The trajectory of the corresponding particle level dynamics is a straight line,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.



(a) Projection onto the x_1 axis versus time.

(b) Projection onto the x_2 axis versus time.

FIG. 1. *Numerical results of the geodesic equation example in section* 5.2. *Plots on the left and right show the projections of the trajectories* $T_{\theta(t)}(x)$ *onto the* x_1 *and* x_2 *axes, respectively, for five different initial positions* $x = (x_1, x_2) \in \mathbb{R}^2$ *(shown by different colors) sampled from* ρ_0. *The results match exactly to the true push-forward map* $T_t(x) = ((1-t)x_1, x_2)$. *Note that this approach does not suffer the singularity of* ρ_t *at* $t = 1$.

i.e., the true push-forward map is $T_t(x) = x + \nabla\Phi(0,x)t$ where x is an arbitrary initial position following the reference density ϱ. Note that $T_t(x)$ is only determined by the initial velocity $\nabla\Phi(0,x)$ in this case. For some choices of Φ, the system may develop a singularity (e.g., the density function ρ_t reduces to one on a low-dimesional subset of $\mathbb{R}^d$) in finite time. The numerical experiments, which are presented next, show that our method can approximate the solution very well, even beyond the singularity. The PWHF (3.39) corresponding to the Wasserstein geodesic equation reduces to

$$\begin{aligned} \dot{\theta} &= \widehat{G}(\theta)^{\dagger} p, \\ \dot{p} &= \frac{1}{2}(\widehat{G}(\theta)^{\dagger} p)^{\top} (\nabla_{\theta} \widehat{G}(\theta)) \widehat{G}(\theta)^{\dagger} p. \end{aligned} \tag{5.3}$$

To demonstrate that our method can overcome singularity, we set $\Phi(0,x) = -\frac{1}{2}x_1^2$ with $x = (x_1, x_2) \in \mathbb{R}^2$, in which case $\nabla\Phi(0,x) = (-x_1, 0)$. Therefore the true push-forward is $T_t(x) = ((1-t)x_1, x_2)$. As we can see, under such a push-forward, all particles will arrive at the x_2-axis, and the density ρ_t degenerates to a Gaussian density supported on the x_2-axis only, at time $t = 1$. Then we solve the parameter dynamics (5.3) using Algorithm 4.1 (PWHF) and compare the result $T_{\theta(t)}$ with the true solution T_t using their particle level dynamics.

To illustrate the solution $T_{\theta(t)}$ of Algorithm 4.1, we choose five points (shown by different colors) sampled randomly from ρ_0 and plot the projections of their trajectories onto x_1 and x_2 axes in the left and right panels of Figure 1. As we can see, their trajectories obtained by push-forward $T_{\theta(t)}$ match the true solution T_t exactly. To qualitatively evaluate the approximation accuracy of $\rho_{\theta(t)}$ to the true ρ_θ, we draw 2,000 samples from reference distribution λ and then push-forward them using $T_{\theta(t)}$ obtained by Algorithm 4.1. The densities $\rho_{\theta(t)}$ at different times are shown in section SM3.3. Note that $T_1(x) = (0, x_2)$ for all $x = (x_1, x_2) \in \mathbb{R}^2$, which results in the singularity of $\rho_{\theta(t)}$, matching the actual behavior of ρ_t, at time $t = 1$.

5.3. Wasserstein Hamiltonian flow with quadratic potential. In the second experiment, we consider a set of three examples where the potential $\mathcal{F}$ is set to be the quadratic function of the position variable, i.e., $\mathcal{F}(\rho) = \int_{\mathbb{R}^d} V(x)\rho(x)dx$, where V is a quadratic function of x. The problem can be solved explicitly in this case as shown in section 3.5, and the approximation errors δ_0 and δ_1 vanish if we set the push-forward map T_θ as an affine transform.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

In our experiments, the potential function V and initial condition of Φ are set to have the following form:

$$V(x) = \sum_{i=1}^{d} \frac{1}{2} a_i x_i^2 \qquad \text{and} \qquad \Phi(0,x) = \sum_{i=1}^{d} \frac{1}{2} b_i x_i^2, \tag{5.4}$$

where $a_i > 0$ and $b_i \in \mathbb{R}$ for all i. We can write the WHF as

$$\frac{\partial}{\partial t}\rho + \nabla \cdot (\rho \nabla \Phi) = 0, \tag{5.5a}$$

$$\frac{\partial}{\partial t}\Phi + \frac{1}{2}|\nabla \Phi|^2 = -\sum_i \frac{1}{2} a_i x_i^2 \tag{5.5b}$$

with initial $\rho_0 = \mathcal{N}(0, I)$ and $\Phi(0,x) = \sum_{i=1}^{d} \frac{1}{2} b_i x_i^2$. From the particle version of equations, we know that both the position and the velocity can be expressed explicitly: the ith component of the solution $X(t,x)$ (location of the particle at time t started from x at time 0) given by

$$X_i(t,x) = \sqrt{1+b_i^2} \cdot x_i \cdot \cos(\sqrt{a_i} \cdot t - \arctan(b_i)), \tag{5.6}$$

where $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ and $X_i(t,x)$. The true push-forward map can be shown to satisfy

$$T_t(x) = A(t)x, \qquad \text{where} \qquad A(t) = \operatorname{diag}\left(\left[\sqrt{1+b_i^2} \cos(\sqrt{a_i} \cdot t - \arctan(b_i)) \right]_{i=1}^{d} \right). \tag{5.7}$$

We can see that a_i determines the frequency, while b_i determines the amplitude and initial phase. In what follows, we will consider a set of three examples with different dimension d and values of a_i's and b_i's.

5.3.1. 2D harmonic oscillator. We consdier a 2D harmonic oscillator with potential energy $\mathcal{F}(\rho) = \int_{\mathbb{R}^2} \frac{1}{2}(2.25x_1^2 + 0.36x_2^2)\rho(x)dx$ and initial dual function $\Phi(0,x) = -\frac{1}{2}x_1^2$ as in section 3.5. We choose affine transform as the push-forward map, i.e.,

$$T_\theta(x) = \Gamma z + b, \qquad \theta := (\Gamma, b), \qquad \text{where} \qquad \Gamma \in \mathbb{R}^{2\times 2}, b \in \mathbb{R}^2. \tag{5.8}$$

In this case, the true push-forward map T_t is the same as in (5.7) and $T_{\theta(t)} \equiv T_t$ for all t. We also observe a linear dependence between the error $\widehat{\epsilon}$ defined in (5.2) and the time step size h empirically, as shown in the left panel of Figure 2.

We also confirm that the symplectic Euler (or alike) is necessary to preserve the Hamiltonian $H(\theta, p)$ in our numerical scheme. In the right panel of Figure 2, we plot the change of Hamiltionian $H(\theta^{l+1}, p^{l+1}) - H(\theta^l, p^l)$ for $l = 0, \ldots, K-1$ obtained by Algorithm 4.1 (PWHF) in the orange curve. In comparison, if we replace the symplectic Euler with a standard forward Euler step, the change of the Hamiltonian is significantly deviated from zero, as shown by the blue curve, suggesting the Hamiltonian is not preserved.

5.3.2. 2D Lissajous curves. For a single point $x = (x_1, x_2) \in \mathbb{R}^2$, its motion under the Hamiltonian flow (5.5) is a 2D harmonic oscillator. We again consider quadratic potential and initial dual in the form given in (5.4) for $d = 2$. Denoting $\Delta\beta_0 = \arctan(b_1) - \arctan(b_2)$ as the initial phase difference and $r = \sqrt{a_1/a_2}$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

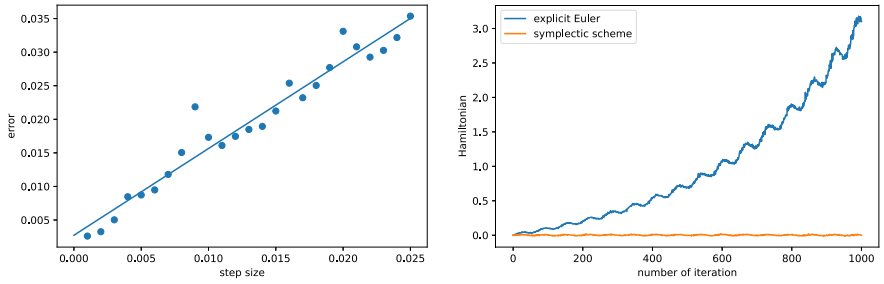
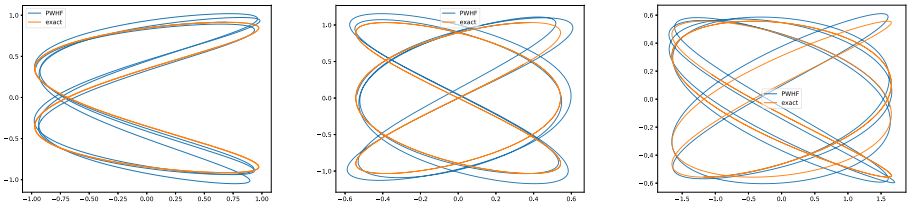


FIG. 2. *Numerical results of 2D harmonic oscillator example in section* 5.3.1. *(Left) Error of push-forward mapping* $\widehat{\epsilon}(T_\theta, T)$ *versus different step sizes* h. *Blue line is a linear regression line fitting the errors. (Right) Change of Hamiltonian* $H(\theta^{l+1}, p^{l+1}) - H(\theta^l, p^l)$ *obtained by Algorithm* 4.1 *(orange curve) by a standard Euler scheme to* (3.14) *(blue curve) versus iteration number* l.



(a) $\Delta\beta_0 = \frac{\pi}{2}, r = \frac{1}{2}$. (b) $\Delta\beta_0 = \frac{\pi}{2}, r = \frac{2}{3}$. (c) $\Delta\beta_0 = \frac{\pi}{2}, r = \frac{3}{4}$.

FIG. 3. *Numerical results of 2D Lissajous curves in section* 5.3.2. *The three plots show the ground truth Lissajous curves (orange curves) and the push-forward trajectories obtained by Algorithm* 4.1 *(blue curves) corresponding to different combinations of* $(\Delta\beta_0, r)$.

as the ratio of frequency for two components, it is well-known that we may have interesting periodic patterns known as the Lissajous curves for some combination of $(\Delta\beta_0, r)$.

We run Algorithm 4.1 (PWHF) where T_θ is the same as in section 5.2 for several different combinations of $(\Delta\beta_0, r)$ with suitable initial positions, and plot their trajectories under the map $T_{\theta(t)}$ obtained by the algorithm in Figure 3. To show the full structure of the Lissajous curve, we run 20,000 steps, which corresponds to a physical time $t_1 = 40$ and step size $h = 0.002$. From Figure 3, we can see that the trajectory obtained by Algorithm 4.1 (blue curve) follows the true Lissajous curve (orange curve) closely, demonstrating promising accuracy of the proposed approach.

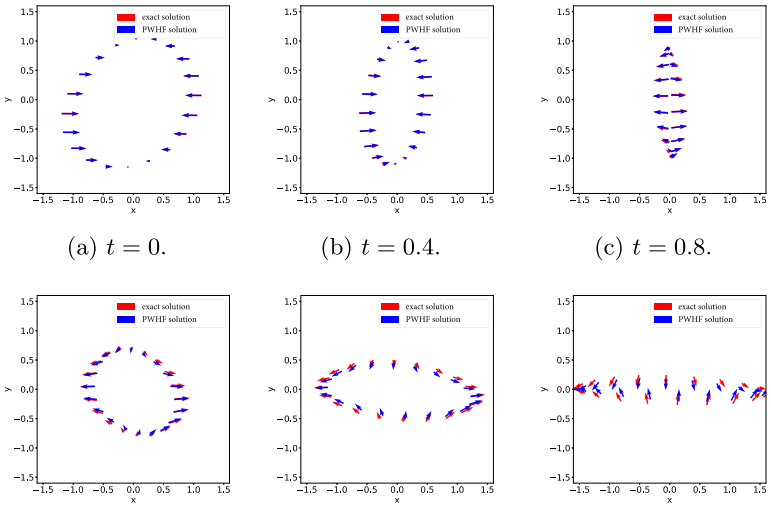
In addition to the verification of accurate position tracking above, we can also show the accuracy of velocity of our method. Specifically, we consider a particular choice of quadratic potential function and initial value of Ψ as follows:

$$\mathcal{F}(\rho) = \int_{\mathbb{R}^2} \left(\frac{1}{2} x_1^2 + \frac{1}{3} x_2^2 \right) \rho(x)\, dx, \qquad \Phi(0, x) = -\frac{1}{2} x_1^2. \tag{5.9}$$

For a fixed initial point $x = T_{\theta_0}(z)$ where z is from the reference density ϱ, its true velocity at time t is $v(t, x) = \frac{d}{dt} X(t, x)$, where $X(t, x)$ is given in (5.6), and the velocity from our model can be evaluated through $\tilde{v}(t, x) := \frac{\partial}{\partial \theta} T_{\theta(t)}(z) \cdot \dot{\theta}(t)$. We choose 20 initial points and plot their positions as well as the velocities on the (x_1, x_2) plane as in Figure 4 at different times t, where v are shown in red arrows and $\tilde{v}$ in blue arrows. The starting point of each arrow represents the position, and the arrow (length and

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy



(a) $t = 0$. (b) $t = 0.4$. (c) $t = 0.8$.

(d) $t = 1.2$. (e) $t = 1.6$. (f) $t = 2$.

FIG. 4. *Evolution of position and velocity in a 2D Lissajous curve example given in section* 5.3.2. *There are* 20 *position-velocity pairs* $(X(t), v(t, X(t))$ *shown as arrows at* 6 *different times* t. *Tails of arrows show the positions and the direction and length of arrows show the corresponding velocity. Red arrows are ground truth position-velocity pairs, and blue arrows are position-velocity pairs obtained by Algorithm* 4.1 *(PWHF).*

direction) itself represents the velocity. We can see close matching of the computed velocity $\tilde{v}$ and the true one v. Note that the true trajectories for different initial positions may intersect, which leads to the singularity of density function ρ_t. The velocity $\tilde{v}$ is still close to the true velocity v despite the singularity.

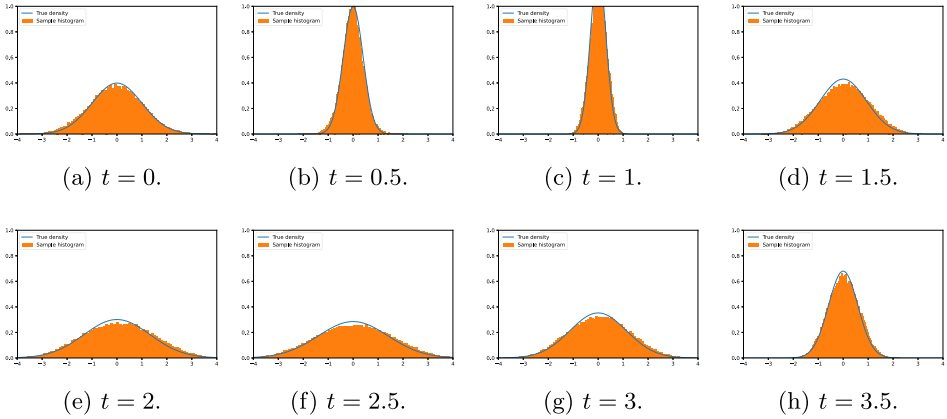
5.3.3. 10D harmonic oscillator. One of the major advantages of our approach is that it can be applied to high-dimensional PDEs. Note that classical numerical schemes suffer the so-called curse of dimensionality as their computation complexity grows exponentially fast in problem dimension d and thus often become infeasible for $d > 3$. In contrast, our method is sample-based and can handle the issue effectively. In this example, we test Algorithm 4.1 (PWHF) on a 10D harmonic oscillator problem with potential function and initial dual variable given by

$$\mathcal{F}(\rho) = \int_{\mathbb{R}^{10}} \left(\frac{3}{8} x_1^2 + \frac{1}{2} \sum_{i=2}^{10} x_i^2 \right) \rho(x)\, dx, \qquad \Phi(0, x) = \frac{1}{2} \sum_{i=2}^{10} x_i^2.$$

We set T_θ to the residual neural network as in (5.1) with 80 neurons in each hidden layer. We solve the system up to $t_1 = 2\pi$ with step size $h = 0.001$. We generate 10,000 samples from $\rho_{\theta(t)}$ and plot the sample histograms (orange bars) projected to the x_2-axis and compare them with the true density ρ_t (blue curve) in Figure 5. The projection of the true density ρ_t onto the x_2-axis is a Gaussian distribution with mean zero and variance a cosine function of time t, as shown by the blue curve in this figure. Similar to the geodesic equation example, ρ_t develops a singularity in finite time in this case, while numerical results show that our approach handles the problem well despite the singularity and high dimension. We plot the empirical Hamiltonian from these samples in the left panel of Figure 6. We emphasize that the value of the Hamiltonian is concentrated around 9.35 with rather small variations. This indicates

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy



(a) $t = 0$. (b) $t = 0.5$. (c) $t = 1$. (d) $t = 1.5$.

(e) $t = 2$. (f) $t = 2.5$. (g) $t = 3$. (h) $t = 3.5$.

FIG. 5. *Numerical result of* 10*D harmonic oscillator in section* 5.3.3. *The plots show the evolution of the histogram of the push-forward density* $\rho_{\theta(t)}$ *(orange bars) and the true density* ρ_t *(blue curves), both projected onto the* x_2*-axis, at different times* t.

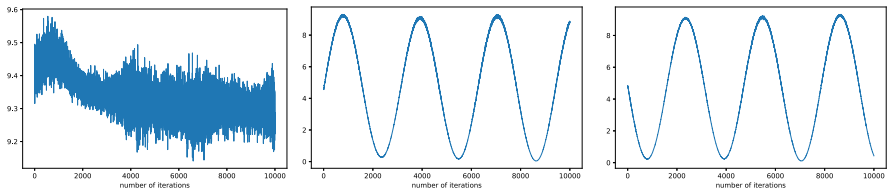


FIG. 6. *Numerical result of* 10*D harmonic oscillator in section* 5.3.3. *The plots show the evolutions of Hamiltonian (left), kinetic energy (middle), and potential energy (right) versus iteration number. The range in the vertical axis on the left plot is small, indicating the conservation of Hamiltonian.*

that the Hamiltonian is essentially preserved. The kinetic (middle) and potential (right) energy oscillate in the opposite phase as shown in Figure 6, respectively.

5.4. Wasserstein Hamiltonian flow with quadratic nonlinear interaction potential. We show that our method can also be applied to more challenging cases with nonlinear potential fucntions such as the interaction potential or entropy in the remainder of this section. The nonlinear terms are generally hard to evaluate directly through the traditional methods, although some algorithms such as kernel density estimation [7, 47] can be used for it. On the contrary, push-forward maps may provide a good alternative for this purpose. Interaction potential is described through the following potential energy $\mathcal{F}$:

$$\mathcal{F}(\rho) = \iint_{\mathbb{R}\times\mathbb{R}} C(x,y)\rho(x)\rho(y)dxdy. \tag{5.10}$$

As we can see, $\mathcal{F}$ involves a double integral of ρ and introduces nonlinearity, which makes the problem much more difficult to solve in general. In physics, $C(x,y) = \frac{a}{|x-y|^2}$ is often used to model gravity or Coulomb force. To avoid numerical instability we will use a modified $C_b(x,y) = \frac{1}{b+|x-y|^2}$ with $b = 0.1$ in the following experiment.

We choose the push-forward map T_θ to be the neural network described in section 5.2. For this problem we do not have an analytic form of the true push-forward T_t, so we compare the result with particle level numerical simulations. More precisely, we generate 10,000 random samples and run the particle level dynamics (3.7)

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

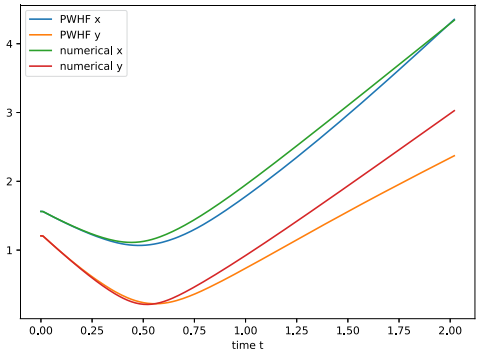


FIG. 7. *Numerical result on nonlinear interactive potential in section* 5.4. *The plot shows the projections onto the* x_1- *and* x_2-axes *(denoted by* x *and* y *in the plots, respectively) of the trajectories* $T_{\theta(t)}(x)$ *obtained by Algorithm* 4.1 *and* $T_t(x)$ *obtained by solving the particle level dynamics* (3.7). *Specifically, the blue and orange curves are the projections of* $T_{\theta(t)}(x)$ *onto the* x_1- *and* x_2-*axes, respectively. The green and red curves are the projections of* $T_t(x)$ *onto the* x_1- *and* x_2-*axes, respectively.*

to obtain the numerical approximation to $T_t(x)$ for all x from those samples and then we compare $T_{\theta(t)}(x)$ obtained by Algorithm 4.1 (PWHF) with $T_t(x)$. We randomly select an initial point x and plot the projections onto the x_1- and x_2-axes of these two trajectories, which are shown in Figure 7. This plot shows a close matching between the two trajectories.

5.5. Empirical bound on δ_0. In section 3.4, we showed the convergence result of $\theta(t)$. There are two crucial terms δ_0 and δ_1, which show up in the error estimation (3.50) as well as (3.25). It is generally difficult to give a sharp bound on these values, although one may resort to the well-known universal approximation theorem stating that they are small if proper neural networks are used. Here we provide an empirical estimation of δ_0 in some of the experiments we discussed above. To this end, we denote the approximation error at θ as

$$\begin{aligned}\delta(\theta) &= \int \left| \nabla \frac{\delta}{\delta \rho} \mathcal{F} \circ T_\theta(z) - \mathcal{K}_\theta \left[\nabla \frac{\delta}{\delta \rho} \mathcal{F} \circ T_\theta \right](z) \right|^2 d\lambda(z) \\ &= \eta^\top \widehat{G}(\theta) \eta - 2 \nabla_\theta F(\theta)^\top \eta + \mathbb{E}_{\rho_\theta} \left[\left| \nabla \frac{\delta}{\delta \rho} \mathcal{F} \circ T_\theta(z) \right|^2 \right],\end{aligned}$$

where $\eta = \widehat{G}(\theta)^\dagger \nabla_\theta F(\theta)$. It is easy to see that δ_0 can be replaced by $\widehat{\delta}_0 = \sup_{t \in [0,T]} \delta(\theta(t))$ in our error estimation, since only the information of $\delta(\theta)$ along the solution curve $\{\theta(t) : t \in [0,T]\}$ is used in the proof.

Assume $\{\theta^l : 0 \le l \le K\}$ is the numerical solution generated by Algorithm 4.1, where $K = t_1/h$ is the number of iterations and t_1 is the terminal physical time. Denote the empirical bound on $\widehat{\delta}_0$ as

$$\tilde{\delta}_0 = \max_{0 \le k \le K} \delta(\theta_k). \tag{5.11}$$

Then we report $\tilde{\delta}_0$, which is an empirical estimation of δ_0 for several experiments with quadratic potential $V(x)$ in Table 1. It can be seen that $\tilde{\delta}_0$ is generally small in these experiments.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

TABLE 1
Empirical estimation $\tilde{\delta}_0$ (defined in (5.11)) of δ_0 in the cases of 2D geodesic equation (section 5.2), 2D harmonic oscillator (section 5.3.1), and 10D harmonic oscillator (section 5.3.3).

Experiment	Terminal time t_1	# iterations K	$\tilde{\delta}_0$
2D geodesic equation	4	2,000	0
2D harmonic oscillator	40	20,000	0.0035
10D harmonic oscillator	10	10,000	0.0908

6. Discussion. The proposed method is also potentially applicable to many other important topics, for example, the nonlinear Schrödinger equation, the Schrödinger bridge problem, and the geodesic between two points on the Wasserstein manifold. However, it needs further investigation to extend our algorithm to them, which is beyond the scope of this paper. We only provide a brief discussion here.

Geodesic between two points on Wasserstein manifold. In the numerical experiments, we consider the initial value problem for the Wasserstein geodesic equation with given ρ_0, Φ_0. The commonly encountered geodesic problem is a two-point boundary value problem, i.e., with given ρ_0, ρ_T but no prior knowledge about the initial Φ_0. In this case, there is not enough information for us to initialize the p variable. A possible attempt is to choose an initial guess $\Phi(0, \cdot)$ and then apply the shooting method on the ODE dynamics. Collocation or finite difference can be other options too. They deserve careful study in the future.

Schrödinger equation. With the Madelung transform, the Schrödinger equation can be reformulated as a WHF with Hamiltonian [12]:

(6.1)
$$\mathcal{H}(\rho, \Phi) = \int_{\mathbb{R}^d} \frac{1}{2} |\nabla \Phi(x)|^2 \rho(x) dx + \int_{\mathbb{R}^d} V(x)\rho(x)dx + \mathcal{F}_1(\rho) + \frac{1}{8} \int_{\mathbb{R}^d} |\nabla \log\ \rho(x)|^2 \rho(x) dx.$$

Possible challenges may arise due to the nonlinear potential $\int_{\mathbb{R}^d} |\nabla \log\ \rho(x)|^2 \rho(x) dx$. This term involves the computation of a space derivative of the log density function, which is generally unavailable for a multilayer perceptron or the normalizing flow. One strategy is to use the neural ODE as the push-forward map [10, 24], which supports such a computation. Another possible solution is to reformulate this functional as a minimization problem and estimate it via optimization techniques.

Schrödinger bridge problem. Similar to the Schrödinger equation, the Schrödinger bridge problem can also be reformulated as WHF through the Hopf–Cole transform. The challenge in this problem is the combination of the difficulties in the two aforementioned examples. On the one side, we need to handle the computational challenge of $\nabla \log\ \rho$ term as mentioned in the Schrödinger equation. On the other side, the Schrödinger bridge problem is a two-point boundary value problem with given ρ_0, ρ_T but no prior knowledge about Φ_0.

7. Conclusion. We close the discussion by summarizing that we developed a sampling-based approach called PWHF for solving WHF in this work. PWHF is derived by applying Hamiltonian mechanics in the parameter space equipped with the pullback Wasserstein metric. Error estimates show that PWHF can approximate WHF with provable accuracy provided the push-forward map is efficient in approximation. Numerical examples demonstrate that our method is robust to the singularity of the equation and can scale up to high-dimensional problems. There is still much work

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

to be done about the WHF, which includes the application of PWHF to other Hamiltonian systems such as the Schrödinger equation or the Schrödinger bridge system; theoretical analysis on the quantities $\delta_0, \delta_1, \delta_2, \lambda_{\min,\Theta}(\widehat{G})$; and extension of PWHF to general Hamiltonian flow with on-quadratic kinetic energy. We hope the current study may serve as a starting point for furthering those investigations.

REFERENCES

[1] Y. Achdou and I. Capuzzo-Dolcetta, *Mean field games: Numerical methods*, SIAM J. Numer. Anal., 48 (2010), pp. 1136–1162, https://doi.org/10.1137/090758477.

[2] Y. Achdou and Z. Kobeissi, *Mean field games of controls: Finite difference approximations*, Math. Eng., 3 (2021), pp. 1–35, https://doi.org/10.3934/mine.2021024.

[3] S.-I. Amari, *Natural gradient works efficiently in learning*, Neural Comput., 10 (1998), pp. 251–276, https://doi.org/10.1162/089976698300017746.

[4] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein generative adversarial networks*, in Proceedings of the International Conference on Machine Learning, 2017, pp. 214–223.

[5] G. Bao, X. Ye, Y. Zang, and H. Zhou, *Numerical solution of inverse problems by weak adversarial networks*, Inverse Problems, 36 (2020), 115003, https://doi.org/10.1088/1361-6420/abb447.

[6] M. Betancourt, *A Conceptual Introduction to Hamiltonian Monte Carlo*, preprint, arXiv:1701.02434, 2017.

[7] Z. I. Botev, J. F. Grotowski, and D. P. Kroese, *Kernel density estimation via diffusion*, Ann. Statist., 38 (2010), pp. 2916–2957, https://doi.org/10.1214/10-AOS799.

[8] F. Camilli and F. Silva, *A semi-discrete approximation for a first order mean field game problem*, Netw. Heterog. Media, 7 (2012), pp. 263–277, https://doi.org/10.3934/nhm.2012.7.263.

[9] R. Chen and M. Tao, *Data-driven prediction of general Hamiltonian dynamics via learning exactly-symplectic maps*, in Proceedings of the International Conference on Machine Learning, PMLR, 2021, pp. 1717–1727.

[10] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, *Neural ordinary differential equations*, in Advances in Neural Information Processing Systems 31, 2018.

[11] Y. Chen and W. Li, *Optimal transport natural gradient for statistical manifolds with continuous sample space*, Inf. Geom., 3 (2020), pp. 1–32, https://doi.org/10.1007/s41884-020-00028-0.

[12] S.-N. Chow, W. Li, and H. Zhou, *Wasserstein Hamiltonian flows*, J. Differential Equations, 268 (2020), pp. 1205–1219, https://doi.org/10.1016/j.jde.2019.08.046.

[13] J. Cui, L. Dieci, and H. Zhou, *A continuation multiple shooting method for Wasserstein geodesic equation*, SIAM J. Sci. Comput., 44 (2022), pp. A2918–A2943, https://doi.org/10.1137/21M142160X.

[14] J. Cui, L. Dieci, and H. Zhou, *Time discretizations of Wasserstein–Hamiltonian flows*, Math. Comp., 91 (2022), pp. 1019–1075, https://doi.org/10.1090/mcom/3726.

[15] K. Cui and H. Koeppl, *Approximately solving mean field games via entropy-regularized deep reinforcement learning*, in Proceedings of the 24th International Conference on Artificial Intelligence and Statistics, Proc. Mach. Learn. Res. 130, A. Banerjee and K. Fukumizu, eds., PMLR, 2021, pp. 1909–1917, https://proceedings.mlr.press/v130/cui21a.html.

[16] L. Dinh, J. Sohl-Dickstein, and S. Bengio, *Density Estimation Using Real NVP*, preprint, arXiv:1605.08803, 2016.

[17] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, *Hybrid Monte Carlo*, Phys. Lett. B, 195 (1987), pp. 216–222, https://doi.org/10.1016/0370-2693(87)91197-X.

[18] W. E, J. Han, and A. Jentzen, *Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations*, Commun. Math. Stat., 5 (2017), pp. 349–380, https://doi.org/10.1007/s40304-017-0117-6.

[19] W. E and B. Yu, *The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems*, Commun. Math. Stat., 6 (2018), pp. 1–12, https://doi.org/10.1007/s40304-018-0127-z.

[20] N. Gaby, X. Ye, and H. Zhou, *Neural Control of Parametric Solutions for High-Dimensional Evolution PDEs*, preprint, arXiv:2302.00045, 2023.

[21] M. Girolami and B. Calderhead, *Riemann manifold Langevin and Hamiltonian Monte Carlo methods*, J. R. Stat. Soc. Ser. B. Stat. Methodol., 73 (2011), pp. 123–214, https://doi.org/10.1111/j.1467-9868.2010.00765.x.

[22] H. Goldstein, *Classical Mechanics*, Pearson Education India, 2011.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[23] G. H. Golub and V. Pereyra, *The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate*, SIAM J. Numer. Anal., 10 (1973), pp. 413–432, https://doi.org/10.1137/0710036.

[24] W. Grathwohl, R. T. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud, *Ffjord: Free-form continuous dynamics for scalable reversible generative models*, in Proceedings of the International Conference on Learning Representations, 2019.

[25] S. Greydanus, M. Dzamba, and J. Yosinski, *Hamiltonian neural networks*, in Advances in Neural Information Processing Systems 32, 2019.

[26] E. Haghighat, M. Raissi, A. Moure, H. Gomez, and R. Juanes, *A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics*, Comput. Methods Appl. Mech. Engrg., 379 (2021), 113741, https://doi.org/10.1016/j.cma.2021.113741.

[27] E. Hairer, M. Hochbruck, A. Iserles, and C. Lubich, *Geometric numerical integration*, Oberwolfach Rep., 3 (2006), pp. 805–882, https://doi.org/10.4171/owr/2006/14.

[28] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[29] K. Huang, Y. Wang, M. Tao, and T. Zhao, *Why do deep residual networks generalize better than deep feedforward networks?—A neural tangent kernel perspective*, in Proceedings of the 34th International Conference on Neural Information Processing Systems, Curran Associates, Red Hook, NY, 2020.

[30] R. Jordan, D. Kinderlehrer, and F. Otto, *The variational formulation of the Fokker–Planck equation*, SIAM J. Math. Anal., 29 (1998), pp. 1–17, https://doi.org/10.1137/S0036141096303359.

[31] J.-M. Lasry and P.-L. Lions, *Jeux à champ moyen.* I–*le cas stationnaire*, C. R. Math., 343 (2006), pp. 619–625, https://doi.org/10.1016/j.crma.2006.09.019.

[32] J.-M. Lasry and P.-L. Lions, *Mean field games*, Jpn. J. Math., 2 (2007), pp. 229–260, https://doi.org/10.1007/s11537-007-0657-8.

[33] M. Lauriere, S. Perrin, S. Girgin, P. Muller, A. Jain, T. Cabannes, G. Piliouras, J. Perolat, R. Elie, O. Pietquin, and M. Geist, *Scalable deep reinforcement learning algorithms for mean field games*, in Proceedings of the 39th International Conference on Machine Learning, Proc. Mach. Learn. Res. 162, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, eds., PMLR, 2022, pp. 12078–12095, https://proceedings.mlr.press/v162/lauriere22a.html.

[34] W. Li, A. T. Lin, and G. Montúfar, *Affine natural proximal learning*, in Proceedings of the Geometric Science of Information: 4th International Conference, Springer, Berlin, 2019, pp. 705–714.

[35] W. Li, S. Liu, H. Zha, and H. Zhou, *Parametric Fokker-Planck equation*, in Geometric Science of Information, F. Nielsen and F. Barbaresco, eds., Springer, Cham, 2019, pp. 715–724.

[36] W. Li and G. Montúfar, *Natural gradient via optimal transport*, Inf. Geom., 1 (2018), pp. 181–214, https://doi.org/10.1007/s41884-018-0015-3.

[37] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, *Fourier Neural Operator for Parametric Partial Differential Equations*, preprint, arXiv:2010.08895, 2020.

[38] A. T. Lin, S. W. Fung, W. Li, L. Nurbekyan, and S. J. Osher, *Alternating the population and control neural networks to solve high-dimensional stochastic mean-field games*, Proc. Natl. Acad. Sci. USA, 118 (2021), e2024713118, https://doi.org/10.1073/pnas.2024713118.

[39] A. T. Lin, W. Li, S. Osher, and G. Montúfar, *Wasserstein proximal of GANs*, in Geometric Science of Information: 5th International Conference, Springer, Berlin, 2021, pp. 524–533.

[40] S. Liu, W. Li, H. Zha, and H. Zhou, *Neural parametric Fokker–Planck equation*, SIAM J. Numer. Anal., 60 (2022), pp. 1385–1449, https://doi.org/10.1137/20M1344986.

[41] F. Otto, *The geometry of dissipative evolution equations: The porous medium equation*, Comm. Partial Differential Equations, 26 (2001), pp. 101–174, https://doi.org/10.1081/PDE-100002243.

[42] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, J. Comput. Phys., 378 (2019), pp. 686–707, https://doi.org/10.1016/j.jcp.2018.10.045.

[43] D. Rezende and S. Mohamed, *Variational inference with normalizing flows*, in Proceedings of the International Conference on Machine Learning, PMLR, 2015, pp. 1530–1538.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[44] L. Ruthotto, S. J. Osher, W. Li, L. Nurbekyan, and S. W. Fung, *A machine learning framework for solving high-dimensional mean field game and mean field control problems*, Proc. Natl. Acad. Sci. USA, 117 (2020), pp. 9183–9193, https://doi.org/10.1073/pnas.1922204117.
[45] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, 2003.
[46] O. So, G. Li, E. A. Theodorou, and M. Tao, *Data-driven discovery of non-Newtonian astronomy via learning non-Euclidean Hamiltonian*, in Machine Learning and the Physical Sciences Workshop, NeurIPS, 2022.
[47] G. R. Terrell and D. W. Scott, *Variable kernel density estimation*, Ann. Statist., 20 (1992), pp. 1236–1265, https://doi.org/10.1214/aos/1176348768.
[48] P. Toth, D. J. Rezende, A. Jaegle, S. Racanière, A. Botev, and I. Higgins, *Hamiltonian Generative Networks*, preprint, arXiv:1909.13789, 2019.
[49] C. Villani, *Optimal Transport: Old and New*, Grundlehren Math. Wiss. 338, Springer, Berlin, 2009.
[50] S. Wang, H. Wang, and P. Perdikaris, *Learning the solution operator of parametric partial differential equations with physics-informed DeepONets*, Sci. Adv., 7 (2021), eabi8605, https://doi.org/10.1126/sciadv.abi8605.
[51] Y. Zang, G. Bao, X. Ye, and H. Zhou, *Weak adversarial networks for high-dimensional partial differential equations*, J. Comput. Phys., 411 (2020), 109409, https://doi.org/10.1016/j.jcp.2020.109409.

Downloaded 04/10/25 to 108.69.91.181 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.