COMPUTATIONAL NONLOCAL MEAN FIELD GAMES WITH APPLICATIONS

SITING LIU, MATTHEW JACOBS, WUCHEN LI, LEVON NURBEKYAN,
AND STANLEY J. OSHER

ABSTRACT. We introduce a novel framework to model and solve mean-field game systems with nonlocal interactions extending the results in [42]. Our approach relies on kernel-based representations of mean-field interactions and feature-space expansions in the spirit of kernel methods in machine learning. We demonstrate the flexibility of our approach by modeling various interaction scenarios between agents. Additionally, our method yields an efficient saddle-point reformulation of the original problem that is amenable to state-of-the-art convex optimization methods such as the primal-dual hybrid gradient method (PDHG). We also discuss potential applications of our methods to multi-agent trajectory planning problems.

1. INTRODUCTION

In this paper, we study computational and modeling aspects of nonlocal mean-field game systems. Specifically, we are interested in the system

$$
\begin{cases}
-\phi_t + H(x, \nabla\phi) = f\left(x, \int_\Omega K(x,y)\rho(y,t)dy\right) \\
\rho_t - \nabla \cdot (\rho\nabla_p H(x, \nabla\phi)) = 0 \\
\rho(x,0) = \rho_0(x), \ \phi(x,1) = g\left(x, \int_\Omega S(x,y)\rho(y,1)dy\right)
\end{cases}
\tag{1.1}
$$

Above, $\Omega \subset \mathbb{R}^d$ is a domain with smooth boundary, $H : \Omega \times \mathbb{R}^d \to \mathbb{R}$ is the Hamiltonian, $K, S : \Omega \times \Omega \to \mathbb{R}$ are interaction kernels, and $f, g : \Omega \times \mathbb{R} \to \mathbb{R}$ are interaction functions.

System (1.1) describes Nash equilibria in an infinite-dimensional differential game where a continuum of agents interact through their distribution in the state-space. These games are called mean-field games (MFG) and were introduced by Huang, Malhamé, and Caines [30, 29], and Lasry and Lions [36, 37, 38]. Currently, MFG is a thriving research direction with applications in economics [28, 2, 25, 6], finance [34, 17, 18, 24], crowd motion [35, 14, 8, 7], industrial engineering [33, 22, 27], data science [23], material dynamics [43], and more. For a detailed introduction and review of MFG theory we refer to [38, 28, 16, 26].

In (1.1), $\rho(\cdot, t)$ represents the population density at time t, and ρ_0 is an initial distribution. Furthermore, $(x, t) \mapsto \phi(x, t)$ is a value function that measures the optimal value of an agent at position x and time t. Functions f, g and kernels K, S model the interaction between a single agent and the population.

Throughout the paper, we assume that f, g, K, S are C^2 functions, and a pair (ϕ, ρ) is a weak solution of (1.1) if ϕ is a viscosity and ρ is a distribution solution

for HJB and continuity equations, respectively. The PDE theory of (1.1) is well understood in this setting, and we refer to [38, 16] for a detailed exposition of the subject.

Our goal is to develop computational and modeling methods for (1.1). There are several general purpose numerical methods for MFG that apply to (1.1). In [5, 3, 1, 4], the authors develop and analyze finite-difference methods for (1.1) and related models. Their approach is based on a solution of the discrete problem applying Newton's method.

For so-called *potential* MFG systems, there are several primal-dual convex optimization methods such as alternating direction method of multipliers (ADMM) [9, 10] and primal-dual hybrid gradient (PDHG) [13, 12]. These methods are extensions of the celebrated Brenier-Benamou method for computation of optimal transport maps [11] to the MFG setting.

In this paper, our goal is to develop computational and modeling methods specifically for nonlocal MFG systems. The term nonlocal refers to expressions

$$f\left(x, \int_\Omega K(x,y)\rho(y,t)dy\right), \quad g\left(x, \int_\Omega S(x,y)\rho(y,1)dy\right).$$

Indeed, the calculation of these terms at x requires the knowledge of $\rho(y,t)$ at all y. For general MFG systems, these terms are replaced by $f(x,\rho)$, $g(x,\rho)$ where one allows local interactions as well; that is, $f(x,\rho) = \tilde{f}(x,\rho(x,t))$, and $g(x,\rho) = \tilde{g}(x,\rho(x,1))$ where, $\tilde{f}, \tilde{g} : \Omega \times \mathbb{R}_+ \to \mathbb{R}$ are some functions, and $\rho(x,t)$ is the density of ρ. Note that for local interactions, the calculation of f, g terms at x requires information only at x.

The majority of numerical methods above apply to general models, including nonlocal interactions. However, they suit best the local ones, where a discretization of ρ on a grid yields a straightforward discretization for interactions. For the nonlocal case though, the calculation of interaction terms requires matrix multiplication on a full grid to evaluate the expressions $\int_\Omega K(x,y)\rho(y,t)dy$, $\int_\Omega S(x,y)\rho(y,1)dy$. Our approach solves this problem by encoding the interactions in a small number of *expansion coefficients*.

Furthermore, a critical feature of primal-dual methods mentioned above is that one of the proximal steps results in a decoupled system of one-dimensional convex optimization problems at the grid-points. Therefore, this step is parallelizable and yields a linear computational cost in the number of grid-points. However, direct applications of these methods to nonlocal problems yield fully coupled systems that are not parallelizable and yield a superlinear computational cost. Our method solves this problem as well. The expansion coefficients, that encode the interactions, decouple the aforementioned systems. Furthermore, we update these coefficients by an explicit proximal step that yields a linear computational cost.

Our method is also well-suited for the Lagrangian framework. Indeed, in [42], where this approach was introduced, the authors solved (1.1) in Lagrangian coordinates. Thus, this approach paves a way to efficient computational methods for high-dimensional MFG problems.

Another appealing feature of our method is the flexibility of modeling interactions. We expand K, S in a basis that can be interpreted as features from kernel methods in ML [41, Chapter 6]. This allows us to design various interactions by only manipulating the basis. In particular, we can easily model heterogeneous regimes where agents interact only within specific subdomains and other interesting scenarios. Additionally, we can handle nonlocal interactions that are given by differential operators [5, Tests 5, 6].

Finally, we would like to point out potential applications of our methods to multi-agent trajectory planning. In general, even single-agent trajectory-panning problems are highly complex. With the number of agents increasing in a system, problems quickly become computationally overwhelming.

A critical difficulty comes from modeling and computing the interactions. MFG theory provides a flexible framework to solve this problem. Theoretically, MFG solutions are optimal only when the number of agents is infinite. Nevertheless, one can generate sub-optimal trajectories that have appealing properties such as no-collision. Since our method provides a way of encoding mean-field interactions in a few coefficients independent of the number of agents, it is potentially applicable to large multi-agent trajectory planning problems.

The paper has the following organization. In Section 2, we provide a detailed description of our method. In Section 3, we devise a PDHG algorithm to solve (1.1) based on our method and discuss the computational efficiency of our approach. In Section 4, we show how to model and approximate interactions using kernel methods from ML. Next, in Section 5, we discuss potential applications of our methods to multi-agent trajectory planning problems. Finally, in Section 6 we provide several numerical experiments.

2. The method of coefficients

For the sake of simplicity, we assume that $f(x, \zeta) = \zeta$, and $g(x, \zeta) = g(x)$; that is, we consider the system

$$\begin{cases} -\phi_t + H(x, \nabla \phi) = \int_\Omega K(x, y)\rho(y, t)dy \\ \rho_t - \nabla \cdot (\rho \nabla_p H(x, \nabla \phi)) = 0 \\ \rho(x, 0) = \rho_0(x), \ \phi(x, 1) = g(x) \end{cases} \tag{2.1}$$

The essence of the method is in an expansion of $K(x, y)$ in a family of functions. More precisely, assume that $\{f_i\}_{i=1}^r \subset C^2(\Omega)$ is an arbitrary family of functions. Furthermore, suppose that

$$K(x, y) = \sum_{i,j=1}^r k_{ij} f_i(x) f_j(y) \tag{2.2}$$

Remark 2.1. In general, K may not have the form (2.2). In such cases, we approximate K with kernels of such form.

We denote by $\mathbf{K} = (k_{ij}) \in M_{r \times r}(\mathbb{R})$, and, without loss of generality, we assume that $\mathbf{K}$ is invertible. A straightforward calculation yields that

$$\int_\Omega K(x, y)\rho(y, t)dt = \sum_{i=1}^r a_i(t) f_i(x),$$

where

$$a_i(t) = \sum_{j=1}^r k_{ij} \int_\Omega f_j(y)\rho(y, t)dy.$$

This means that no matter what ρ is, the expression $\int_\Omega K(x, y)\rho(y, t)dy$ is a combination of $\{f_i\}_{i=1}^r$ with some *unknown coefficients* $a_1, a_2, \cdots, a_r$. These coefficients encode all necessary information about the interactions. Thus, (a_i) will be our new unknowns. Note that once we find (a_i) we can solve (2.1) by solving decoupled HJB and continuity equations.

It turns out that (a_i) are zeroes of a specific operator that is *monotone* if $\rho \mapsto \int_\Omega K(x,y)\rho(y)dy$ is monotone, and a *gradient* when K is symmetric. To state the results precisely, denote by ϕ_a the viscosity solution of the HJB

$$\begin{cases} -\phi_t + H(x, \nabla\phi) = \sum_{i=1}^r a_i(t)f_i(x) \\ \phi(x,1) = g(x) \end{cases} \tag{2.3}$$

Furthermore, denote by ρ_a the distributional solution of the continuity equation

$$\begin{cases} \rho_t - \nabla \cdot (\rho\nabla_p H(x, \nabla\phi_a)) = 0 \\ \rho(x,0) = \rho_0(x) \end{cases} \tag{2.4}$$

The following two theorems are the basis of our approach.

Theorem 2.2. [42, Theorem 2.3] *The functional $a \mapsto \int_\Omega \phi_a(x,0)\rho_0(x)dx$ is concave and everywhere Fréchet differentiable. Moreover,*

$$\frac{\delta}{\delta a_i} \int_\Omega \phi_a(x,0)\rho_0(x)dx = \int_\Omega f_i(x)\rho_a(x,\cdot)dx, \ 1 \le i \le r. \tag{2.5}$$

Theorem 2.3. [42, Theorem 3.1]

i. *A pair (ϕ, ρ) is a solution of (2.1) if and only if $(\phi, \rho) = (\phi_a, \rho_a)$ for some $a \in C([0,1];\mathbb{R}^r)$ such that*

$$a = \mathbf{K}\frac{\delta}{\delta a}\int_\Omega \phi_a(x,0)\rho_0(x)dx \tag{2.6}$$

ii. *If $\mathbf{K}$ is monotone, then (2.6) is equivalent to finding a zero of a monotone operator $a \mapsto \mathbf{K}^{-1}a - \frac{\delta}{\delta a}\int_\Omega \phi_a(x,0)\rho_0(x)dx$, $a \in C([0,1];\mathbb{R}^r)$.*

iii. *Additionally, if $\mathbf{K}$ is symmetric, (2.6) is equivalent to the convex optimization problem*

$$\inf_{a\in C([0,1];\mathbb{R}^r)} \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi_a(x,0)\rho_0(x)dx, \tag{2.7}$$

where $\langle a, b \rangle = \sum_{i=1}^r \int_0^1 a_i(t)b_i(t)dt$ for $a, b \in C([0,1];\mathbb{R}^r)$.

Remark 2.4. Several remarks are in order.

1. Theorem 2.3 asserts that instead of finding (ϕ, ρ) in (2.1) we just need to find the right coefficients (a_i) and then solve the decoupled equations (2.3), (2.4).

2. Here, we do not concentrate on technical aspects of Theorems 2.2, 2.3 that are available for a periodic case, $\Omega = \mathbb{T}^d$, in [42]. For the non-periodic case, (2.1) and (2.3) must be complemented with the no-flux Neumann boundary condition $\nabla_p H(x, \nabla\phi) \cdot \nu = 0$ on $\partial\Omega$.

3. $\mathbf{K}$ is monotone if and only if $\rho \mapsto \int_\Omega K(x,y)\rho(y)dy$ is monotone; that is, $\int_{\Omega^2} K(x,y)(\rho_2(x) - \rho_1(x))(\rho_2(y) - \rho_1(y))dxdy \ge 0$ for all $\rho_1, \rho_2 \in \mathcal{P}(\Omega)$. This condition is essential for the uniqueness of solutions of (2.1) [38].

4. $\mathbf{K}$ is symmetric if and only if K is symmetric; that is, $K(x,y) = K(y,x)$ for all $x, y \in \Omega$.

5. For monotone interactions, (a_i) are solutions of a monotone variational inequality or a convex optimization problem. Therefore, one can apply powerful convex optimization techniques to find (a_i).

6. Equation (2.5) is critical for update rules of (a_i). Indeed, it gives the ascent direction $\int_\Omega \phi_a(x,0)\rho_0(x)dx$ with respect to (a_i). Moreover, this direction depends on ρ_a, which is available once ϕ_a is (approximately) computed at current (a_i). As we shall see below, this property yields extremely simple update rules for (a_i).

7. These previous points are also very appealing for potential applications of the methods here to multi-agent trajectory planning problems.

8. Optimization problem (2.7) is equivalent to the infinite-dimensional optimal control problem (58)-(59) in [38].

9. There is nothing special about the choice $f(x, \zeta) = \zeta$, $g(x, \zeta) = g(x)$ except simplicity. Analogous results can be proven for general f, g as well. This topic is a subject of our future research.

10. Precisely the same theorems are valid for stochastic systems as well. Again, we will address this topic in our future work.

3. A PRIMAL-DUAL HYBRID GRADIENT ALGORITHM

Here, we formulate (2.7) as a convex-concave saddle point problem and devise a primal-dual hybrid gradient (PDHG) algorithm of Chambolle and Pock [19, 20] to solve it. Introducing a Lagrange multiplier for the HJB equation in (2.3) we obtain

$$
\begin{aligned}
&\inf_{a \in C([0,1];\mathbb{R}^r)} \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi_a(x,0)\rho_0(x)dx \\
&= \inf_{\phi,a} \left\{ \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi(x,0)\rho_0(x)dx \text{ s.t. (2.3) holds} \right\} \\
&= \inf_{\substack{\phi(x,1)=g \\ a}} \sup_\rho \left\{ \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi(x,0)\rho_0(x)dx \right. \\
&\qquad \left. + \int_\Omega \int_0^1 \rho \left(-\phi_t + H(x, \nabla\phi) - \sum_{i=1}^r a_i(t)f_i(x) \right) dtdx \right\} \\
&= \inf_{\substack{\phi(x,1)=g \\ a}} \sup_{\rho,v} \left\{ \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi(x,0)\rho_0(x)dx - \int_\Omega \int_0^1 (\rho\phi_t + \rho v \cdot \nabla\phi) \, dtdx \right. \\
&\qquad \left. - \int_\Omega \int_0^1 \rho \left(L(x,v) + \sum_{i=1}^r a_i(t)f_i(x) \right) dtdx \right\},
\end{aligned}
$$

(3.1)

where we used the convex duality

$$
H(x, \nabla\phi) = \sup_v -v \cdot \nabla\phi - L(x,v).
$$

The saddle point problem in (3.1) is convex in (ϕ, a) and almost concave in (ρ, v). Non-concavity comes from the terms $\rho v \cdot \phi$ and $\rho L(x,v)$. Following [11], we remedy this problem by replacing v with a flux variable $m = \rho v$. Thus, we obtain an equivalent saddle point problem

$$
\begin{aligned}
&\inf_{\substack{\phi(x,1)=g \\ a}} \sup_{\rho,m} \left\{ \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi(x,0)\rho_0(x)dx - \int_\Omega \int_0^1 (\rho\phi_t + m \cdot \nabla\phi) \, dxdt \right. \\
&\qquad \left. - \int_\Omega \int_0^1 \rho \left(L \left(x, \frac{m}{\rho} \right) + \sum_{i=1}^r a_i(t)f_i(x) \right) dxdt \right\} \\
&= \inf_{\substack{\phi(x,1)=g \\ a}} \sup_{\rho,m} \mathcal{L}(\phi, a, \rho, m)
\end{aligned}
$$

(3.2)

Note that $(\phi, a) \mapsto \mathcal{L}(\phi, a, \rho, m)$ is convex, $(\rho, m) \mapsto \mathcal{L}(\phi, a, \rho, m)$ is concave, and the coupling between (ϕ, a) and (ρ, m) is bilinear. Thus, we can apply PDHG [19, 20]

to solve (3.2). Also, note that the first-order optimality conditions for ϕ are

$$\begin{cases} \rho_t + \nabla \cdot m = 0 & \text{in} \quad \Omega \times (0,1) \\ m(x,t) \cdot \nu = 0 & \text{in} \quad \partial\Omega \times (0,1) \\ \rho(x,0) = \rho_0(x) & \text{in} \quad \Omega \end{cases}$$

Therefore, the no-flux boundary condition for m and the initial condition for ρ are incorporated in (3.2), and no extra considerations are necessary.

Furthermore, we must add a constraint $\rho \geq 0$ to account for ρ being a probability distribution. This adjustment is coherent with the derivation (3.1) because the viscosity-solution constraint (2.3) should be replaced by pointwise constraints $-\partial_t + H(x, \nabla\phi) \leq \sum_{i=1}^{r} a_i(t) f_i(x)$, $\phi(x,1) \leq g(x)$. We refer to [10] for details.

As illustrated in [32, 31], the choices of spaces for variables are crucial when applying PDHG. Correct choices render algorithms with grid-size-independent convergence rates. For a, ρ, m we choose L^2 spaces, whereas for ϕ we choose H^1. The motivation for this choice comes from convergence analysis of PDHG. Indeed, upper bounds on the step-sizes depend on the inverse of the norm of the bilinear coupling

$$\left| \int_\Omega \int_0^1 (\rho\phi_t + m \cdot \nabla\phi) \, dx dt \right| \leq \|(\rho, m)\|_{L^2} \cdot \|\phi\|_{H^1}$$

This norm is finite if we choose L^2 norm for (ρ, m) and H^1 norm for ϕ. If we chose L^2 norm for ϕ, the bilinear coupling would have infinite norm. Therefore the corresponding norm of the finite-dimensional coupling on a grid would depend on the grid-size and converge to infinity as grid gets finer. Consequently, the step-sizes that guarantee the convergence of the algorithm would shrink to 0 and yield an impractically slow algorithm. The H^1 norm, on the other hand, yields convergence guarantees and rates that are grid-independent.

For step-sizes $\tau_{\nabla\phi}, \tau_{\phi_t}, \tau_{\phi(0)}, \tau_\rho, \tau_m$, and current iterates $(a^k, \phi^k, \rho^k, m^k, \bar{a}^k, \bar{\phi}^k)$ the update rules for PDHG are

$$\begin{cases} (\rho^{k+1}, m^{k+1}) & = \underset{\rho,m}{\text{argmax}} \ \mathcal{L}(\bar{\phi}^k, \bar{a}^k, \rho, m) - \frac{1}{2\tau_\rho}\|\rho - \rho^k\|_{L^2_{x,t}}^2 - \frac{1}{2\tau_m}\|m - m^k\|_{L^2_{x,t}}^2 \\ (a^{k+1}, \phi^{k+1}) & = \underset{a,\phi}{\text{argmin}} \ \mathcal{L}(\phi, a, \rho^{k+1}, m^{k+1}) + \frac{1}{2\tau_{\phi(0)}}\|\phi(\cdot, 0) - \phi^k(\cdot, 0)\|_{L^2_x}^2 \\ & \quad + \frac{1}{2\tau_{\nabla\phi}}\|\nabla\phi - \nabla\phi^k\|_{L^2_{x,t}}^2 + \frac{1}{2\tau_{\phi_t}}\|\phi_t - \phi_t^k\|_{L^2_{x,t}}^2 + \frac{1}{2\tau_a}\|a - a^k\|_{L^2_t}^2 \\ (\bar{a}^{k+1}, \bar{\phi}^{k+1}) & = 2(a^{k+1}, \phi^{k+1}) - (a^k, \phi^k) \end{cases}$$
(3.3)

The critical observation is that the variational problems above are well-posed and easy to solve. In what follows we discuss in details each of the updates.

The updates for (ρ, m). We have that

$$\frac{\delta\mathcal{L}}{\delta\rho} = -\phi_t - L\left(x, \frac{m}{\rho}\right) + \nabla_v L\left(x, \frac{m}{\rho}\right) \cdot \frac{m}{\rho} - \sum_{i=1}^{r} a_i(t) f_i(x)$$

$$\frac{\delta\mathcal{L}}{\delta m} = -\nabla\phi - \nabla_v L\left(x, \frac{m}{\rho}\right)$$

Therefore, for updating (ρ, m) we must solve the following system

$$\begin{cases} \nabla_v L\left(x, \frac{m}{\rho}\right) \cdot \frac{m}{\rho} - L\left(x, \frac{m}{\rho}\right) - \frac{\rho - \rho^k}{\tau_\rho} = \bar{\phi}_t^k + \sum_{i=1}^{r} \bar{a}_i^k(t) f_i(x) \\ \nabla_v L\left(x, \frac{m}{\rho}\right) + \frac{m - m^k}{\tau_m} = \nabla\bar{\phi}^k \end{cases}$$
(3.4)

Remark 3.1. System (3.4) yields decoupled one-dimensional convex optimization problems at the grid-points. Therefore the proximal update for (ρ, m) can de performed efficiently in parallel. This feature is one of the most appealing properties of ADMM [11, 9, 10] and PDHG [13, 12, 32, 39] algorithms for MFG systems and related problems for local couplings. However, direct extensions of aforementioned methods to nonlocal MFG systems do not preserve this property. One of the critical features of our approach is that we preserve this property. We refer to Section 3.1 for details.

For some Lagrangians, (3.4) simplifies greatly. For instance, for $L(x,v) = \frac{|v|^2}{2} + Q(x,t)$ we have that $\nabla_v L(x,v) = v$, and (3.4) becomes

$$\begin{cases} \frac{|m|^2}{2\rho^2} - \frac{\rho - \rho^k}{\tau_\rho} = Q(x,t) + \bar{\phi}_t^k + \sum_{i=1}^r \bar{a}_i^k(t) f_i(x) \\ \frac{m}{\rho} + \frac{m - m^k}{\tau_m} = \nabla \bar{\phi}^k \end{cases}$$

Furthermore, eliminating m from the second equation, we obtain

$$\begin{cases} \frac{|m^k + \tau_m \nabla \bar{\phi}^k|^2}{2(\tau_m + \rho)^2} - \frac{\rho - \rho^k}{\tau_\rho} = Q(x,t) + \bar{\phi}_t^k + \sum_{i=1}^r \bar{a}_i^k(t) f_i(x) \\ m = \rho \frac{m^k + \tau_m \nabla \bar{\phi}^k}{\tau_m + \rho} \end{cases} \tag{3.5}$$

Therefore, we just need to solve a cubic equation for ρ and update m by an explicit formula.

Remark 3.2. As mentioned before, we must add a constraint $\rho \geq 0$ in (3.2). Therefore, equations above must be complemented by the condition $\rho \geq 0$. Furthermore, the expression $L\left(x, \frac{m}{\rho}\right)$ must be understood in the following sense

$$L\left(x, \frac{m}{\rho}\right) = \begin{cases} L\left(x, \frac{m}{\rho}\right), & \text{when } \rho > 0 \\ 0, & \text{when } m = 0, \ \rho = 0 \\ +\infty, & \text{when } m \neq 0, \ \rho = 0 \end{cases}$$

The function $\gamma(\rho) = \frac{|m^k + \tau_m \nabla \bar{\phi}^k|^2}{2(\tau_m + \rho)^2} - \frac{\rho - \rho^k}{\tau_\rho}$, $\rho \geq 0$ is strictly decreasing. Therefore, either $\gamma(0) \geq Q(x,t) + \bar{\phi}_t^k + \sum_{i=1}^r \bar{a}_i^k(t) f_i(x)$ or $\gamma(0) < Q(x,t) + \bar{\phi}_t^k + \sum_{i=1}^r \bar{a}_i^k(t) f_i(x)$. In the former case, there exists a unique $\rho^{k+1}(x,t) \geq 0$ such that $\gamma(\rho^{k+1}(x,t)) = Q(x,t) + \bar{\phi}_t^k(x,t) + \sum_{i=1}^r \bar{a}_i^k(t) f_i(x)$. In the latter case, we set $\rho^{k+1}(x,t) = 0$. In both cases, we update m accordingly.

The updates for (a, ϕ). We have that

$$\frac{\partial \mathcal{L}}{\partial a} = \mathbf{K}^{-1} a - \left(\int_\Omega f_i(x) \rho(x,t) dx \right)_{i=1}^r$$

$$\frac{\partial \mathcal{L}}{\partial \phi} = \rho_t + \nabla \cdot m + (\rho - \rho_0(x)) (dx \, \llcorner \, \Omega) \times \delta_{t=0} - m \cdot \nu \, d_{\partial \Omega} x \times dt$$

where ν is the outward normal of Ω, and $d_{\partial \Omega}$ is the surface measure of $\partial \Omega$. Note that we only consider variations that preserve the boundary condition $\phi(x,1) = g(x)$.

Therefore, to update (a, ϕ) we must solve the system

$$
\begin{cases}
\mathbf{K}^{-1}a - \left(\int_\Omega f_i(x)\rho^{k+1}(x,t)dx\right)_{i=1}^r + \frac{a-a^k}{\tau_a} = 0 \\
\rho_t^{k+1} + \nabla \cdot m^{k+1} - \frac{\phi_{tt}-\phi_{tt}^k}{\tau_{\phi_t}} - \frac{\Delta\phi-\Delta\phi^k}{\tau_{\nabla\phi}} = 0 \\
\rho^{k+1}(x,0) - \rho_0(x) - \frac{\phi_t(x,0)-\phi_t^k(x,0)}{\tau_{\phi_t}} + \frac{\phi(x,0)-\phi^k(x,0)}{\tau_{\phi(0)}} = 0 \\
\phi(x,1) = g(x) \\
\left(-m^{k+1} + \frac{\nabla\phi-\nabla\phi^k}{\tau_{\nabla\phi}}\right) \cdot \nu = 0
\end{cases}
\tag{3.6}
$$

Note that the equations for a and ϕ are decoupled. Furthermore, we obtain explicit updates for a:

$$
a^{k+1} = (\tau_a \mathbf{K}^{-1} + \mathbf{I})^{-1}\left(a^k + \tau_a\left(\int_\Omega f_i(x)\rho^{k+1}(x,t)dx\right)_{i=1}^r\right)
\tag{3.7}
$$

Finally, to update ϕ we must solve the space-time elliptic equation

$$
\begin{cases}
\frac{\phi_{tt}}{\tau_{\phi_t}} + \frac{\Delta\phi}{\tau_{\nabla\phi}} = \rho_t^{k+1} + \nabla \cdot m^{k+1} + \frac{\phi_{tt}^k}{\tau_{\phi_t}} + \frac{\Delta\phi^k}{\tau_{\nabla\phi}} & \text{in} \quad \Omega \times (0,1) \\
\frac{\phi_t(x,0)}{\tau_{\phi_t}} - \frac{\phi(x,0)}{\tau_{\phi(0)}} = \rho^{k+1}(x,0) - \rho_0(x) + \frac{\phi_t^k(x,0)}{\tau_{\phi_t}} - \frac{\phi^k(x,0)}{\tau_{\phi(0)}} & \text{in} \quad \Omega \\
\phi(x,1) = g(x) \quad \text{in} \quad \Omega \\
\frac{\partial\phi(x,t)}{\partial\nu} = \frac{\partial\phi^k(x,t)}{\partial\nu} + \tau_{\nabla\phi}m^{k+1} \cdot \nu \quad \text{in} \quad \partial\Omega \times (0,1)
\end{cases}
\tag{3.8}
$$

This step can be efficiently performed by the Fast Fourier Transform (FFT).

3.1. **Dimension reduction.** Here, we illustrate the dimension reduction and computational efficiency obtained with our method. Assume that Ω is bounded, and K_r, S_r are approximations of K, S in the C^2 norm. Then we have that

$$
\left\|\int_\Omega K(\cdot,y)\rho(y)dy - \int_\Omega K_r(\cdot,y)\rho(y)dy\right\|_{C^2} \leq \|K - K_r\|_{C^2},
$$

$$
\left\|\int_\Omega S(\cdot,y)\rho(y)dy - \int_\Omega S_r(\cdot,y)\rho(y)dy\right\|_{C^2} \leq \|S - S_r\|_{C^2},
$$

for all $\rho \in \mathcal{P}(\Omega)$. Therefore, if we approximate f, g by f_r, g_r in the C^2 norm, we obtain C^2 approximations of the terms

$$
f\left(x, \int_\Omega K(x,y)\rho(y,t)dy\right), \quad g\left(x, \int_\Omega S(x,y)\rho(y,1)dy\right)
$$

that are uniform in ρ. Therefore, f_r, g_r, K_r, S_r produce an approximation of (1.1) that is independent of the grid-size or the number of agents.

Furthermore, from the stability theory of (1.1) [38, 16], we have that solutions of (1.1) corresponding to f_r, g_r, K_r, S_r are precompact in $C(\Omega\times[0,1])\times C([0,1],\mathcal{P}(\Omega))$, and all accumulation points are solutions corresponding to f, g, K, S. Additionally, if the operators

$$
\rho \mapsto f\left(x, \int_\Omega K(x,y)\rho(y)dy\right), \quad \rho \mapsto g\left(x, \int_\Omega S(x,y)\rho(y)dy\right)
$$

are monotone, (1.1) admits a unique solution, (ϕ, ρ), and

$$
\lim_{r\to\infty}\|\phi_r - \phi\|_{L^\infty} = 0, \quad \lim_{r\to\infty}\sup_{t\in[0,1]}W_1(\rho_r(\cdot,t),\rho(\cdot,t)) = 0,
$$

if $\lim_{r\to\infty}\|\xi - \xi_r\|_{C^2} = 0$, $\xi \in \{f, g, K, S\}$, where W_1 is the 1-Wasserstein or Monge-Kantorovich distance in $\mathcal{P}(\Omega)$.

Thus, once we produce approximations of f, g, K, S, we obtain an approximation of (1.1) that works equally fine across all discretizations. Therefore, once we fix

r, we can solve the r-problem as accurately as we wish without extra cost for fine meshes. Additionally, as we show below, for fixed r, the computational cost is on par with those of existing algorithms for local couplings. Of course, the smaller r the better, and the size of r depends on how well $\{f_i\}$ approximate f, g, K, S.

We now compare the computational complexity of our method versus direct applications of primal-dual optimization algorithms to solve (1.1). The starting point for these methods is to write (2.1) as a convex optimization problem introduced in [38]. More precisely, when K is symmetric, one has that (2.1) is equivalent to

$$
\begin{aligned}
\inf_{\phi(x,1)\leq g} \sup_{\rho\geq 0,m} \Bigg\{ &- \int_\Omega \phi(x,0)\rho_0(x)dx - \int_\Omega \int_0^1 (\rho\phi_t + m\cdot\nabla\phi)\,dxdt \\
&- \int_0^1 \left\{ \int_\Omega \rho L\left(x,\frac{m}{\rho}\right)dx + \mathcal{F}(\rho(\cdot,t)) \right\}dt \Bigg\} \\
&= \inf_{\substack{\phi(x,1)\leq g \\ \alpha}} \sup_{\rho\geq 0,m} \ \mathcal{L}_1(\phi,\rho,m)
\end{aligned}
\tag{3.9}
$$

where

$$
\mathcal{F}(\rho) = \frac{1}{2}\int_{\Omega^2} K(x,y)\rho(x)\rho(y)dxdy, \quad \rho\in\mathcal{P}(\Omega).
\tag{3.10}
$$

One can also work with the convex dual $\mathcal{F}^*$ of $\mathcal{F}$ by introducing a dual variable α:

$$
\begin{aligned}
&\inf_{\substack{\phi(x,1)\leq g \\ \alpha}} \sup_{\rho\geq 0,m} \Bigg\{ \int_0^1 \mathcal{F}^*(\alpha(\cdot,t))dt - \int_\Omega \phi(x,0)\rho_0(x)dx - \int_\Omega \int_0^1 (\rho\phi_t + m\cdot\nabla\phi)\,dxdt \\
&- \int_\Omega \int_0^1 \rho\left(L\left(x,\frac{m}{\rho}\right) + \alpha\right)dxdt \Bigg\} \\
&= \inf_{\substack{\phi(x,1)\leq g \\ \alpha}} \sup_{\rho\geq 0,m} \ \mathcal{L}_2(\phi,\alpha,\rho,m),
\end{aligned}
$$

$$
\tag{3.11}
$$

where $\mathcal{F}^*(\alpha) = \sup_\rho \int_\Omega \alpha(x)\rho(x)dx - \mathcal{F}(\rho)$. Therefore, there are two options for solving (2.1): (i) work directly with $\mathcal{F}$ and solve (3.9) or its variants [13, 12], (ii) work with the dual $\mathcal{F}^*$ and solve (3.11) or its variants [9, 10]. We illustrate that direct applications of both approaches to nonlocal problems lead to computationally expensive updates.

For concreteness, we estimate the computational complexity of the PDHG algorithm proposed here, with and without applying the coefficients method. The analysis of other primal-dual algorithms is analogous. First, we discuss the option of working directly with $\mathcal{F}$ and solving (3.9). In this case, the proximal update for (ρ,m) would be

$$
\begin{aligned}
\sup_{\rho\geq 0,m} \Bigg\{ &- \int_\Omega \bar{\phi}^k(x,0)\rho_0(x)dx - \int_\Omega \int_0^1 \left(\rho\bar{\phi}_t^k + m\cdot\nabla\bar{\phi}^k\right)dxdt \\
&- \int_0^1 \left\{ \int_\Omega \rho L\left(x,\frac{m}{\rho}\right)dx + \mathcal{F}(\rho(\cdot,t)) \right\}dt \Bigg\} \\
&- \frac{1}{2\tau_\rho}\|\rho - \rho^k\|^2_{L^2_{x,t}} - \frac{1}{2\tau_m}\|m - m^k\|^2_{L^2_{x,t}}
\end{aligned}
$$

Therefore, we must solve the following system

$$\begin{cases} L\left(x, \frac{m}{\rho}\right) - \nabla_v L\left(x, \frac{m}{\rho}\right) \cdot \frac{m}{\rho} + \frac{\rho - \rho^k}{\tau_\rho} + \delta_\rho \mathcal{F}(\rho) = 0 \\ \nabla_v L\left(x, \frac{m}{\rho}\right) + \frac{m - m^k}{\tau_m} = \nabla \bar{\phi}^k \end{cases} \tag{3.12}$$

For local interactions, one has that $\mathcal{F}(\rho) = \int_\Omega F(\rho(x))dx$ for some F, and (3.12) becomes

$$\begin{cases} L\left(x, \frac{m}{\rho}\right) - \nabla_v L\left(x, \frac{m}{\rho}\right) \cdot \frac{m}{\rho} + \frac{\rho - \rho^k}{\tau_\rho} + F'(\rho) = 0 \\ \nabla_v L\left(x, \frac{m}{\rho}\right) + \frac{m - m^k}{\tau_m} = \nabla \bar{\phi}^k, \end{cases}$$

which is a decoupled system of one-dimensional equations that can be solved efficiently at each node. Therefore, the computational complexity of solving this system for local problems is linear in the number of grid-points. However, for nonlocal interactions such as in (3.10) we have that $\delta_\rho \mathcal{F} = \int_\Omega K(x,y)\rho(y)dy$, and so (3.12) is now a fully coupled (in space) system of nonlinear equations. Additionally, the complexity of the systems grows with the mesh-size. One could approximate the term by $\delta_\rho \mathcal{F}(\rho)$ by $\delta_\rho \mathcal{F}(\rho^k)$ and decouple the system. Nevertheless, this would require a matrix multiplication that yields a superlinear computational cost. Moreover, the proximal step could not be parallelized.

With our method, on the other hand, we obtain fully parallel proximal updates for (ρ, m) (3.4) at the expense of solving an $r \times r$ system of equations to update the coefficients (a_i). To assemble the system for (a_i), we need to evaluate terms $\int_\Omega f_i(x)\rho^{k+1}(x,t)dx$ that yields a linear cost in the number of grid-points. Therefore, once r is fixed, we obtain an overall linear cost for updating (ρ, m) and (a_i) which is the case for local interactions.

Next, we discuss the option of working with $\mathcal{F}^*$ and solving (3.11). In this case, the proximal update for α would be

$$\inf_\alpha \int_0^1 \mathcal{F}^*(\alpha(\cdot, t))dt - \int_\Omega \int_0^1 \rho^{k+1}\alpha \, dx dt + \frac{1}{2\tau_\alpha} \|\alpha - \alpha^k\|_{L^2_{x,t}}^2$$

Therefore, we must solve the following system

$$\delta_\alpha \mathcal{F}^*(\alpha)(x,t) - \rho^{k+1}(x,t) + \frac{\alpha(x,t) - \alpha^k(x,t)}{\tau_\alpha} = 0 \tag{3.13}$$

For local interactions, $\mathcal{F}(\rho) = \int_\Omega F(\rho(x))dx$, we can calculate $\mathcal{F}^*(\alpha)$ on the continuum level by an explicit formula

$$\mathcal{F}^*(\alpha) = \int_\Omega F^*(\alpha(x))dx,$$

where F^* is the convex dual of F. Therefore, $\delta_\alpha \mathcal{F}(\alpha)(x) = (F^*)'(\alpha(x))$, and (3.13) becomes

$$(F^*)'(\alpha(x,t)) - \rho^{k+1}(x,t) + \frac{\alpha(x,t) - \alpha^k(x,t)}{\tau_\alpha} = 0$$

As before, we obtain one-dimensional decoupled equations that can be solved in parallel at grid-points.

In the nonlocal case though, the first issue is that we cannot calculate $\mathcal{F}^*$ analytically on the continuum level unless K is special. However, one can calculate $\mathcal{F}^*$ on the discrete level. Assume that $\{x_i\}_{i=1}^N$ is some space-discretization. Then we have that

$$\mathcal{F}(\rho) = \frac{1}{2}\sum_{i,j} K(x_i, x_j)\rho_i\rho_j, \quad \mathcal{F}^*(\alpha) = \frac{1}{2}\sum_{i,j} Q_{ij}\alpha_i\alpha_j, \quad \delta_\alpha \mathcal{F}^*(\alpha)_i = \sum_j Q_{ij}\alpha_j$$

where $\rho_i = \rho(x_i)$, $\alpha_i = \alpha(x_i)$, and $Q = (Q_{ij}) = (K(x_i, x_j))^{-1}$. Therefore, (3.13) becomes an $N \times N$ system of linear equations

$$\sum_j Q_{ij}\alpha_j(t) - \rho_i^{k+1}(t) + \frac{\alpha_i(t) - \alpha_i^k(t)}{\tau_\alpha} = 0, \quad i \in \{1, 2, \cdots, N\}. \tag{3.14}$$

As before, we obtain a coupled system in the nonlocal case. The solution of this system yields a polynomial computational complexity unless $(K(x_i, x_j))$ is special. For instance, if $(K(x_i, x_j))$ is diagonalizable in a Fourier basis the computational cost is of order $N \log N$ via FFT.

In our method, on the other hand, we replace dual variables $(\alpha_i(t))_{i=1}^N$ by coefficients $(a_i(t))_{i=1}^r$, and (3.14) is replaced by an $r \times r$ system (3.7). Therefore, we have to store much less variables and, as mentioned before, obtain a linear computational cost. Additionally, we can calculate the $r \times r$ matrix $(\tau_a \mathbf{K}^{-1} + \mathbf{I})^{-1}$ prior to optimization and use it afterward. Moreover, since the size of this matrix is independent of the mesh-size, we do not have to deal with conditioning issues for every mesh-size separately: we can do it once and for all before optimization.

Finally, as we will see in Section 4, a smart choice of basis functions $\{f_i\}$ yields $\mathbf{K} = \mathbf{I}$. Therefore, the updates for $(a_i(t))_{i=1}^r$ are trivial and there is no need to solve linear systems at all!

4. Modeling interactions with kernels

Here, we discuss modeling aspects of nonlocal MFG systems. In particular, we show how to build kernels to enforce suitable behavior of agents. For that, we draw inspiration from kernel methods in machine learning [41, Chapter 6].

As mentioned before, (2.1) is well posed when $\rho \mapsto \int_\Omega K(x, y)\rho(y)dy$ is monotone. This condition means that agents repel one another and try to minimize the cost

$$\phi(x, t) = \inf_u \int_t^1 \left\{ L(z(s), u(s)) + \int_\Omega K(z(s), y)\rho(y, t)dy \right\}ds + g(z(T))$$
$$\text{s.t. } \dot{z}(s) = c(z(s), u(s)), \quad z(t) = x$$

Therefore, $K(x, y)$ is a *similarity measure* between positions x and y that agents try to minimize. Kernel methods in ML study exactly this type of K for data separation. Different choices of K lead to different separations.

The simplest example of K is the inner product, $K(x, y) = x \cdot y$, which is amenable to rigorous mathematical analysis. Natural extensions of the inner product are *positive definite symmetric (PDS)* kernels.

Definition 4.1. $K : (x, y) \mapsto K(x, y)$ *is a PDS kernel if* $(K(x^i, x^j))_{i,j=1}^m$ *is symmetric positive semidefinite matrix for all* $\{x^i\}_{i=1}^m \subset \mathbb{R}^d$.

Assume K is a continuous PDS. Thus, its symmetric, and for arbitrary $\rho_k = \frac{1}{N}\sum_i w_k^i \delta_{x^i}$, $k = 1, 2$ we have that

$$\int_{\Omega^2} K(x, y)(\rho_2(x) - \rho_1(x))(\rho_2(y) - \rho_1(y))dxdy$$
$$= \sum_{i,j} K(x^i, x^j)(w_2^i - w_1^i)(w_2^j - w_1^j) \geq 0,$$

and hence $\rho \mapsto \int_\Omega K(x, y)\rho(y)dy$ is monotone.

The discussion above shows that *PDS kernels suit MFG models extremely well*. Thus, we will build various MFG models by choosing suitable PDS kernels. In this context, as we shall see below, the basis $\{f_i\}$ corresponds to feature vectors.

The remarkable fact about PDS kernels is that all of them are inner products. More precisely, K is PDS iff there exists a Hilbert space $\mathcal{H}$ and a mapping $x \mapsto f(x) \in \mathcal{H}$ such that

$$K(x,y) = \langle f(x), f(y) \rangle_{\mathcal{H}}, \quad \forall x, y \tag{4.1}$$

In other words, one can associate points $\{x\}$ in the input space with vectors $\{f(x)\}$ in a Hilbert space so that $K(x,y)$ is precisely the inner product of $f(x)$ and $f(y)$ in $\mathcal{H}$ [41, Theorem 6.8]. The vector $f(x)$ is called the *feature vector* of x. If $\mathcal{H}$ is separable we can write $f(x) = (f_1(x), f_2(x), \cdots, f_n(x), \cdots)$ in some basis of $\mathcal{H}$ and $f_1(x), f_2(x), \cdots, f_n(x), \cdots$ will be the *features* of x. $\mathcal{H}$ is called a *reproducing kernel Hilbert space (RKHS)*. For $K(x,y) = x \cdot y = \sum x_i y_i$ the features are simply the coordinates.

The RKHS theory blends very well with our coefficients method by providing the basis we need in the form of feature vectors. Indeed, assume that K is a PDS kernel and $\mathcal{H}$ is its RKHS with a basis $\{e_i\}$. Then, we obtain that

$$
\begin{aligned}
K(x,y) =& \langle f(x), f(y) \rangle_{\mathcal{H}} = \langle \sum_i f_i(x) e_i, \sum_i f_i(y) e_i \rangle_{\mathcal{H}} \\
=& \sum_{i,j} \langle e_i, e_j \rangle_{\mathcal{H}} f_i(x) f_j(y) = \sum_{i,j} k_{ij} f_i(x) f_j(y),
\end{aligned}
$$

which is the representation we need. In this case, $\mathbf{K} = (\langle e_i, e_j \rangle_{\mathcal{H}})$ is the Gram matrix associated to the basis $\{e_i\}$ in $\mathcal{H}$. Below, we present several common choices for K and provide the basis $\{f_i\}$ and the matrix $\mathbf{K}$.

4.1. **Maximal spread.** Assume that we want to enforce a maximal spread of the population by penalizing individual agents for being close to the average position of the population. This means that an individual agent faces an optimal control problem

$$
\begin{aligned}
&\inf_{\dot{z}=c(z,u)} \int_0^1 \ell(z(s), u(s)) - \sum_{i=1}^d \lambda_i \left| z_i(s) - \int_\Omega y_i \rho(y,s) dy \right|^2 ds + g(z(1)) \\
=& \inf_{\dot{z}=c(z,u)} \int_0^1 \left[\ell(z(s), u(s)) - \sum_{i=1}^d \lambda_i |z_i(s)|^2 + 2 \sum_{i=1}^d \lambda_i \left\{ \int_\Omega z_i(s) \cdot y_i \rho(y,s) dy \right\} \right. \\
&\left. - \sum_{i=1}^d \lambda_i \left| \int_\Omega y_i \rho(y,s) dy \right|^2 \right] ds + g(z(1))
\end{aligned}
$$

Above, $\lambda_1, \lambda_2, \cdots, \lambda_d \geq 0$ signify how much we enforce spreading in each coordinate direction. Additionally, $\ell(x, u)$ is some intrinsic running cost.

Since the term $-\sum_{i=1}^d \lambda_i \left| \int_{\mathbb{R}^d} y_i \rho(y,s) dy \right|^2$ does not depend on the trajectory z, the problem above is equivalent to

$$
\begin{aligned}
\inf_{\dot{z}=c(z,u)} \int_0^1 \ell(z(s), u(s)) - \sum_{i=1}^d \lambda_i |z_i(s)|^2 + \int_\Omega \left\{ 2 \sum_{i=1}^d \lambda_i z_i(s) \cdot y_i \right\} \rho(y,s) dy dt \\
+ g(z(1))
\end{aligned}
$$

Therefore, we obtain an MFG system (2.1) where

$$L(x, u) = \ell(x, u) - \sum_{i=1}^{d} \lambda_i |x_i|^2$$

$$K(x, y) = 2 \sum_{i=1}^{d} \lambda_i x_i y_i$$

$$H(x, p) = \sup_u \{-p \cdot c(x, u) - L(x, u)\}$$

The key point is that $\mathbb{R}^d$ is an RHKS for K, and

$$K(x, y) = \sum_{i=1}^{d} f_i(x) f_i(y)$$

where $f_i(x) = \sqrt{2\lambda_i} x_i$, $1 \le i \le d$. Thus, we use these $\{f_i\}$ as the basis in our method. An excellent feature of this choice is that we obtain $\mathbf{K} = \mathbf{K}^{-1} = \mathbf{I}$ which yields a trivial update rule (3.7) for a that reads

$$a_i^{k+1}(t) = \frac{\tau_a \int_\Omega f_i(x) \rho^{k+1}(x, t) dx + a_i^k(t)}{\tau_a + 1}$$

4.2. **Gaussian repulsion.** Another common choice for PDS kernels are Gaussians

$$K(x, y) = \mu \prod_{i=1}^{d} \exp\left(-\frac{|x_i - y_i|^2}{2\sigma_i^2}\right),$$

for some $\mu, \sigma_1, \sigma_2, \cdots, \sigma_d > 0$. The parameter σ_i signifies how repulsive are the agents in i-th coordinate direction.

As before, we will try to find a suitable expansion of K. Using the power series expansion of e^x one can show that

$$K(x, y) = \sum_{\alpha_1, \alpha_2, \cdots, \alpha_d \ge 0} \left\{ \sqrt{\mu} e^{-\sum_{i=1}^{d} \frac{|x_i|^2}{2\sigma_i^2}} \prod_{i=1}^{d} \frac{x_i^{\alpha_i}}{\sigma_i^{\alpha_i} \alpha_i!} \right\} \cdot \left\{ \sqrt{\mu} e^{-\sum_{i=1}^{d} \frac{|y_i|^2}{2\sigma_i^2}} \prod_{i=1}^{d} \frac{y_i^{\alpha_i}}{\sigma_i^{\alpha_i} \alpha_i!} \right\}$$

$$= \sum_{\alpha_1, \alpha_2, \cdots, \alpha_d \ge 0} f_{\alpha_1, \alpha_2, \cdots, \alpha_d}(x) f_{\alpha_1, \alpha_2, \cdots, \alpha_d}(y),$$

where

$$f_{\alpha_1, \alpha_2, \cdots, \alpha_d}(x) = \sqrt{\mu} e^{-\sum_{i=1}^{d} \frac{|x_i|^2}{2\sigma_i^2}} \prod_{i=1}^{d} \frac{x_i^{\alpha_i}}{\sigma_i^{\alpha_i} \alpha_i!}, \quad \alpha_1, \alpha_2, \cdots, \alpha_d \ge 0.$$

Hence, we choose $\{f_{\alpha_1, \alpha_2, \cdots, \alpha_d}\}$ as the basis for our coefficients method and choose n to approximate K with functions of order $\sum_{i=1}^{d} \alpha_i \le n$. As before, an excellent feature of this basis is that $\mathbf{K} = \mathbf{K}^{-1} = \mathbf{I}$.

Interactions in sub-regions. Methods described above also provide flexible framework to model interactions within sub-regions. Assume that Ω_1, Ω_2 are complementary in Ω. Furthermore, assume that agents in Ω_i interact only with those in Ω_i for $i = 1, 2$. There is a straightforward way of extending the framework above to this setting.

Suppose that kernels modeling the interaction in Ω_1, Ω_2 are K_1, K_2, respectively. Additionally, assume that the basis for K_1 is $\{f_i^1\}$, and the one for K_2 is $\{f_i^2\}$ that

can be of a different size. Thus, we want to construct K such that

$$K(x,y) = \begin{cases} K_i(x,y), & (x,y) \in \Omega_i \times \Omega_i \\ 0, & \text{otherwise} \end{cases}$$

Furthermore, we want to construct a basis for K out of $\{f_i^1\}$ and $\{f_i^2\}$. These can be done as follows:

$$\begin{aligned} K(x,y) =& K_1(x,y) \cdot \chi_{\Omega_1}(x)\chi_{\Omega_1}(y) + K_2(x,y) \cdot \chi_{\Omega_2}(x)\chi_{\Omega_2}(y) = \\ =& \sum_{ij} k_{ij}^1 f_i^1(x) f_j^1(y)\chi_{\Omega_1}(x)\chi_{\Omega_1}(y) + \sum_{ij} k_{ij}^2 f_i^2(x) f_j^1(y)\chi_{\Omega_2}(x)\chi_{\Omega_2}(y) \\ =& \sum_{ij} k_{ij}^1 f_i^1(x)\chi_{\Omega_1}(x) f_j^1(y)\chi_{\Omega_1}(y) + \sum_{ij} k_{ij}^2 f_i^2(x)\chi_{\Omega_2}(x) f_j^2(y)\chi_{\Omega_2}(y), \end{aligned}$$

where χ_A is the characteristic function of A. Therefore, the basis for K can be obtained from the ones of K_1, K_2 by simply restricting them into subdomains and combining:

$$\left\{ f_i^1(x)\chi_{\Omega_1}(x), \; f_i^2(x)\chi_{\Omega_2}(x) \right\}$$

Furthermore, we have that

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 \end{pmatrix}, \quad \mathbf{K}^{-1} = \begin{pmatrix} \mathbf{K}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2^{-1} \end{pmatrix},$$

where $\mathbf{K}_i = (k_{ij}^i)$, $i = 1, 2$. Therefore, low complexity matrices for K_1, K_2 yield a low complexity matrix for K.

In case we have multiple regions $\Omega_1, \Omega_2, \cdots, \Omega_N$ with kernels $K_1, K_2, \cdots, K_N$ and bases $\{f_i^1\}, \{f_i^2\}, \cdots \{f_i^N\}$ we obtain a basis

$$\left\{ f_i^1(x)\chi_{\Omega_1}(x), \; f_i^2(x)\chi_{\Omega_2}(x), \; \cdots, f_i^N(x)\chi_{\Omega_N}(x) \right\},$$

and

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \cdots & \mathbf{K}_{N-1} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{K}_N \end{pmatrix}, \quad \mathbf{K}^{-1} = \begin{pmatrix} \mathbf{K}_1^{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2^{-1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \cdots & \mathbf{K}_{N-1}^{-1} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{K}_N^{-1} \end{pmatrix},$$

where $\mathbf{K}_i$ is the matrix corresponding to K_i.

4.3. **Interactions given by differential operators.** Finally, we demonstrate how our methods work for interactions given by differential operators. In a seminal paper on numerical methods for MFG [5], the authors consider an interaction term

$$V[\rho] = \mu(I - \Delta)^{-2}\rho, \tag{4.2}$$

where $\mu > 0$, Δ is the Laplacian operator, and the problem is set on a flat torus $\mathbb{T}^d$. We have that

$$V[\rho] = \int_{\mathbb{T}^d} \Gamma(x - y)\rho(y)dy,$$

where Γ is the fundamental solution; that is,

$$(I - \Delta)^2\Gamma = \mu\delta_0. \tag{4.3}$$

Thus, we have that $K(x,y) = \Gamma(x - y)$. As pointed out in [42], for convolutions on a torus, the appropriate basis is the trigonometric one. Thus, we need to expand Γ into Fourier series with respect to functions $\{\cos(2\pi\alpha \cdot x), \sin(2\pi\alpha \cdot x)\}$ where

$\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_d)$. Furthermore, by the symmetry we have that $\Gamma(x) = \Gamma(-x)$. Therefore, the expansion of Γ contains only even functions; that is,

$$\Gamma(x) = \sum_{\alpha \geq 0} \gamma_\alpha \cos(2\pi\alpha \cdot x)$$

Next, solving (4.3) in a Fourier space yields

$$\gamma_0 = \mu, \quad \gamma_\alpha = \frac{2\mu}{1 + 8\pi^2|\alpha|^2 + 16\pi^4|\alpha|^4}, \quad \alpha > 0,$$

where $|\alpha|^2 = \sum_{i=1}^d \alpha_i^2$. Furthermore, we have that

$$\begin{aligned}
K(x, y) =& \Gamma(x - y) = \sum_{\alpha \geq 0} \gamma_\alpha \cos(2\pi\alpha \cdot (x - y)) \\
=& \sum_{\alpha \geq 0} \left(\gamma_\alpha \cos(2\pi\alpha \cdot x) \cos(2\pi\alpha \cdot y) + \gamma_\alpha \sin(2\pi\alpha \cdot x) \sin(2\pi\alpha \cdot y) \right) \\
=& \sum_{\alpha \geq 0} f_\alpha^{\cos}(x) f_\alpha^{\cos}(y) + \sum_{\alpha > 0} f_\alpha^{\sin}(x) f_\alpha^{\sin}(y),
\end{aligned}$$

where

$$f_\alpha^{\cos}(x) = \sqrt{\gamma_\alpha} \cos(2\pi\alpha \cdot x), \quad f_\alpha^{\sin}(x) = \sqrt{\gamma_\alpha} \sin(2\pi\alpha \cdot x), \quad \alpha \geq 0.$$

Therefore, we choose $\{f_\alpha^{\cos}, f_\alpha^{\sin}\}$ as the basis for our coefficients method and choose n to approximate K with functions of order $\sum_{i=1}^d \alpha_i \leq n$. Again, this choice renders $\mathbf{K} = \mathbf{K}^{-1} = \mathbf{I}$.

5. Potential applications to multi-agent trajectory planning problems

Here, we discuss potential applications of our methods to multi-agent trajectory planning problems. We start by a brief derivation of (1.1) and (2.1). Assume that we have a swarm of agents where agent $i \in \{1, 2, \cdots, N\}$ aims at minimizing a cost

$$\begin{aligned}
\inf_{u_i} \int_t^1 & L(z_i(s), u_i(s), s) + f_i(z_i(s), z_{-i}(s))ds + g_i(z_i(1), z_{-i}(1)) \\
\text{s.t. } & \dot{z}_i(s) = c(z_i(s), u_i(s)), \quad z_i(t) = x_i
\end{aligned} \tag{5.1}$$

Above, $z_{-i} = (z_j)_{j \neq i}$, and f, g model interactions between the agents. This problem leads to a system of N coupled HJBs that is extremely challenging to solve, especially in high-dimensions and for many agents.

The MFG framework provides a solution to this problem by assuming symmetric interactions and considering the continuum limit $N = \infty$. More precisely, if we suppose that

$$f_i(z_i, z_{-i}) = f\left(z_i, \frac{1}{N-1} \sum_{j \neq i} \delta_{z_j} \right), \quad g_i(z_i, z_{-i}) = g\left(z_i, \frac{1}{N-1} \sum_{j \neq i} \delta_{z_j} \right),$$

and formally pass to the limit when $N \to \infty$ we obtain a system where a generic agent solves an optimal control problem

$$\begin{aligned}
\phi(x, t) =& \inf_u \int_t^1 L(z(s), u(s), s) + f(z(s), \rho(\cdot, s))ds + g(z(1), \rho(\cdot, 1)) \\
& \text{s.t. } \dot{z}(s) = c(z(s), u(s)), \quad z(t) = x,
\end{aligned} \tag{5.2}$$

where $\rho(\cdot, s)$ is the distribution of population at time s. We have that ϕ solves the HJB equation

$$\begin{cases} -\phi_t + \sup_u \{-\nabla\phi(x,t) \cdot c(x,u) - L(x,u,t)\} = f(x, \rho(x,t)) \\ \phi(x,1) = g(x, \rho(x,1)), \end{cases}$$

and the optimal control, u^*, is given by the Pontryagin Maximum Principle:

$$u^*(x,t) \in \operatorname{argmax}_u \{-\nabla\phi(x,t) \cdot c(x,u) - L(x,u,t)\}$$

Furthermore, ρ satisfies the continuity equation

$$\begin{cases} \rho_t(x,t) + \nabla \cdot (\rho(x,t)c(x, u^*(x,t))) = 0 \\ \rho(x,0) = \rho_0(x), \end{cases}$$

where ρ_0 is the initial distribution of the agents. Collecting all equations together we obtain the MFG system

$$\begin{cases} -\phi_t + \sup_u \{-\nabla\phi \cdot c(x,u) - L(x,u,t)\} = f(x, \rho(x,t)) \\ \rho_t(x,t) + \nabla \cdot (\rho(x,t)c(x, u^*(x,t))) = 0 \\ u^*(x,t) \in \operatorname{argmax}_u \{-\nabla\phi(x,t) \cdot c(x,u) - L(x,u)\} \\ \rho(x,0) = \rho_0(x), \ \phi(x,1) = g(x, \rho(x,1)) \end{cases} \tag{5.3}$$

Remark 5.1. Equations (1.1), (2.1) correspond to the case $c(x,u) = u$, and $u \mapsto L(x,u)$ convex, for which we have a rigorous mathematical analysis [38].

The appealing feature of MFG systems is that instead of solving a highly coupled system of N HJBs in $d \times N$ dimensions we have to solve a single HJB coupled with a continuity equation in d dimensions. More importantly, the MFG optimal control, u^*, yields *near optimal* controls for the N agent problem (5.1).

Of course, the performance of the MFG control in (5.1) depends on N and gets better as N grows. Nevertheless, it still makes sense to apply MFG controls because they are faster to generate and provide appealing properties such as no-collision trajectories. For instance, if $c(x,u) = u$, and $L(x,u) = \frac{|u^2|}{2} + Q(x,t)$ for some smooth Q, one can show that trajectories corresponding to $u^*(x,t)$ do not intersect [16, Lemma 4.13].

In the context above, our method may provide a flexible way of augmenting existing solution methods for single-agent trajectory planning problems to generate MFG optimal controls for multi-agent problems.

Indeed, Theorem 2.3 asserts that, under the settings of this paper, (5.3) is equivalent to the optimization problem

$$\inf_{a \in C([0,1];\mathbb{R}^r)} \frac{\langle \mathbf{K}^{-1}a, a \rangle}{2} - \int_\Omega \phi_a(x,0)\rho_0(x)dx \tag{5.4}$$

where ϕ_a solves the HJB

$$\begin{cases} -\phi_t + \sup_u \{-\nabla\phi(x,t) \cdot c(x,u) - L(x,u,t)\} = \sum_{i=1}^r a_i(t)f_i(x) \\ \phi(x,1) = g(x), \end{cases} \tag{5.5}$$

In Section 3, we showed how to apply a PDHG algorithm to solve (5.4). Here, we argue that virtually any HJB solver (single-agent trajectory-planner) can be augmented to solve (5.4).

We propose solving (5.4) by some type of gradient descent on $a = (a_i)$. For that, we fix an iterate $a^{current}$ and run any single-agent trajectory planning algorithm to

solve (5.5) for $a = a^{current}$ and generate an optimal control $u^{current}$. Then we solve the forward continuity equation

$$\begin{cases} \rho_t(x,t) + \nabla \cdot \big(\rho(x,t)c(x, u^{current}(x,t))\big) = 0 \\ \rho(x,0) = \rho_0(x), \end{cases}$$

and generate $\rho^{current}$. Finally, we update $a = (a_i)$ by a gradient descent step using (2.5):

$$a_i^{new}(t) = a_i^{current}(t) - h \sum_{j=1}^{r} k_{ij} a_j^{current}(t) + h \int_{\Omega} f_i(x) \rho^{current}(x,t) dx,$$

where $h > 0$ is the descent step-sizes.

Remark 5.2. Several remarks are in order.

1. Explicit gradient descent steps and exact solutions $u^{current}$ can be replaced by implicit (proximal) steps and approximate solutions as in the PDHG here and [42].
2. The approach above works for both Eulerian and Lagrangian solvers. For latter, the terms $\int_{\Omega} f_i(x) \rho^{current}(x,t)$ are simply averages of f_i-s on the trajectories of particles [42].
3. The number of coefficients, r, does not depend on the number of agents, and we always need to (approximately) solve *one decoupled* HJB at each iteration.

Finally, we observe that the methods discussed here also work the other way around; that is, optimal control solvers can be easily adapted to solve MFG problems.

6. Numerical experiments

Here, we present several numerical experiments in a two-dimensional case for kernels and bases discussed in Section 4. We take $\Omega \times [0,T] = [0,1]^2 \times [0,1]$ and choose a uniform space-grid with $N_x = 64$ points per dimension and a uniform time-grid with $N_t = 32$ points in all our examples.

Given N_{x1}, N_{x2}, N_t, we have $\Delta x_1 = \frac{1}{N_{x1}}$, $\Delta x_2 = \frac{1}{N_{x2}}$, $\Delta t = \frac{1}{N_t}$. For $x_1 = i\Delta x_1, x_2 = j\Delta x_2, t_l = l\Delta t$, define

$$\rho_{i,j}^l = \rho(x_i, x_j, t_l) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}, 1 \le l \le N_t$$
$$m_{1,i+\frac{1}{2},j+\frac{1}{2}}^l = m_{x1}(x_{i+\frac{1}{2}}, x_{j+\frac{1}{2}}, t_l) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}, 1 \le l \le N_t$$
$$m_{2,i+\frac{1}{2},j+\frac{1}{2}}^l = m_{x2}(x_{i+\frac{1}{2}}, x_{j+\frac{1}{2}}, t_l) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}, 1 \le l \le N_t$$
$$\phi_{i,j}^l = \phi(x_i, x_j, t_l) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}, 1 \le l \le N_t$$
$$a_{k,l} = a_i(t_l) \quad 1 \le l \le N_t, 1 \le k \le r$$
$$f_{k,i,j} = f_k(x_i, x_j) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}, 1 \le k \le r$$
$$\rho_{i,j}^0 = \rho_0(x_i, x_j) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}$$
$$g_{i,j} = g(x_i, x_j) \quad 1 \le i \le N_{x1}, 1 \le j \le N_{x2}$$

To satisfy the Neumann boundary condition, we have

$$m_{1,i+\frac{1}{2},j+\frac{1}{2}}^l = 0 \quad i = 1, N_{x1}, 1 \le j \le N_{x2}, 1 \le l \le N_t$$
$$m_{2,i+\frac{1}{2},j+\frac{1}{2}}^l = 0 \quad 1 \le i \le N_{x1}, j = 1, N_{x2}, 1 \le l \le N_t$$

The Fokker-Planck equation discretized with forward difference in time as follows:

$$\frac{1}{\Delta t}\left(\rho_{i,j}^{l+1} - \rho_{i,j}^l\right) + \frac{1}{\Delta x_1}\left(m_{i+\frac{1}{2},j}^l - m_{i-\frac{1}{2},j}^l\right) + \frac{1}{\Delta x_2}\left(m_{i,j+\frac{1}{2}}^l - m_{i,j-\frac{1}{2}}^l\right) = 0,$$

The HJB equation is discretized with backward difference in time as follows:

$$-\frac{1}{\Delta t}\left(\phi_j^l - \phi_j^{l-1}\right) + H\left(x_j, \nabla_x \phi_{i,j}^l\right) = \sum_{k=1}^{r} a_{k,l} f_{k,i,j},$$

$$\text{with} \quad \nabla_x \phi_{i,j}^l = \left[\frac{\phi_{i+1,j}^l - \phi_{i,j}^l}{\Delta x_1}, \frac{\phi_{i,j+1}^l - \phi_{i,j}^l}{\Delta x_2}\right]^T.$$

With above finite difference scheme, we are ready to define the objective function of our min-max problem $\mathcal{L}(\phi, a, \rho, m)$ in discrete form:

$$\mathcal{L}(\phi, a, \rho, m) = \frac{\Delta t}{2}\sum_l \sum_{k_1,k_2=1}^{r} a_{k_1,l} a_{k_2,l} r_{k_1 k_2} - \Delta x_1 \Delta x_2 \sum_{i,j} \phi_{i,j,1} \rho_{i,j}^0$$

$$- \Delta x_1 \Delta x_2 \Delta t \sum_{i,j} \sum_l \left(\left(\rho_{i,j}^l \frac{\left(\phi_j^l - \phi_j^{l-1}\right)}{\Delta t} + m_{i+\frac{1}{2},j+\frac{1}{2}}^l \cdot \nabla_x \phi_{i,j}^l\right)\right.$$

$$\left. + \rho_{i,j}^l \left(L(x, \frac{m_{i+\frac{1}{2},j+\frac{1}{2}}^l}{\rho_{i,j}^l}) + \sum_{k=1}^{r} a_{k,l} f_{k,i,j}\right)\right),$$

$$\text{with} \quad m_{i,j}^l = \left[m_{1,i+\frac{1}{2},j+\frac{1}{2}}^l, m_{2,i+\frac{1}{2},j+\frac{1}{2}}^l\right]^T,$$

where $(r_{k_1 k_2}) = \mathbf{K}^{-1}$.

6.1. Maximal spread.

We consider a maximal spread model on from Section 4.1 in the domain $\Omega \times [0,T] = [0,1]^2 \times [0,1]$. We denote by $\rho_G(c_1, c_2, \sigma_G)$ the density of a homogeneous normal distribution centered at (c_1, c_2) with variance σ_G^2. We set the initial-terminal conditions for our MFG system to be

$$\rho_0(x_1, x_2) = \rho_G(0.5, 0.9, 0.2)$$

$$g(x_1, x_2) = 2\exp\left(-10\left(x_1 - 0.5\right)^2 - \left(x_2 - 0.1\right)^2\right)\left(\left(x_2 - 0.1\right)^2 - 1\right).$$

Furthermore, we set

$$L(x, v) = \frac{1}{2}\|v\|^2 + 10^3 \left(\max\left(|x_1 - 0.5|, |x_2 - 0.5|\right)\right)^8$$

We have computed the MFG solutions for four choices of parameters

$$(\lambda_1, \lambda_2) \in \{(0.1, 0.1), (0.1, 4), (4, 0.1), (4, 4)\}$$

The results are shown in Figure 1. We can see that, in accordance to theory, larger λ_i prompt larger spread in x_i directions. Additionally, we see the flexibility of our method for modeling interactions that are heterogeneous in different directions.

6.2. Gaussian repulsion with static obstacles.

We consider a MFG model with Gaussian repulsion from Section 4.2 on the domain $\Omega \times [0,T] = [-1,1]^2 \times [0,1]$. We set the initial-terminal conditions for our MFG system to be

$$\rho_0(x_1, x_2) = \rho_G(0, -0.9, 0.2)$$

$$g(x_1, x_2) = 2\exp\left(-5x_1^2 - 0.25\left(x_2 - 0.9\right)^2\right)\left(\left(x_2 - 0.9\right)^2 - 1\right) + x_1^2$$

We fix this g for all examples with Gaussian repulsion. Furthermore, we set

$$L(x, v) = \frac{1}{2}\|v\|^2 + 10^3 \left(\max\left(|x_1 - 0.5|, |x_2 - 0.5|\right)\right)^8 + Q(x),$$

(A) $\lambda_1 = 0.1, \lambda_2 = 0.1$

(B) $\lambda_1 = 0.1, \lambda_2 = 4$

(C) $\lambda_1 = 4, \lambda_2 = 0.1$
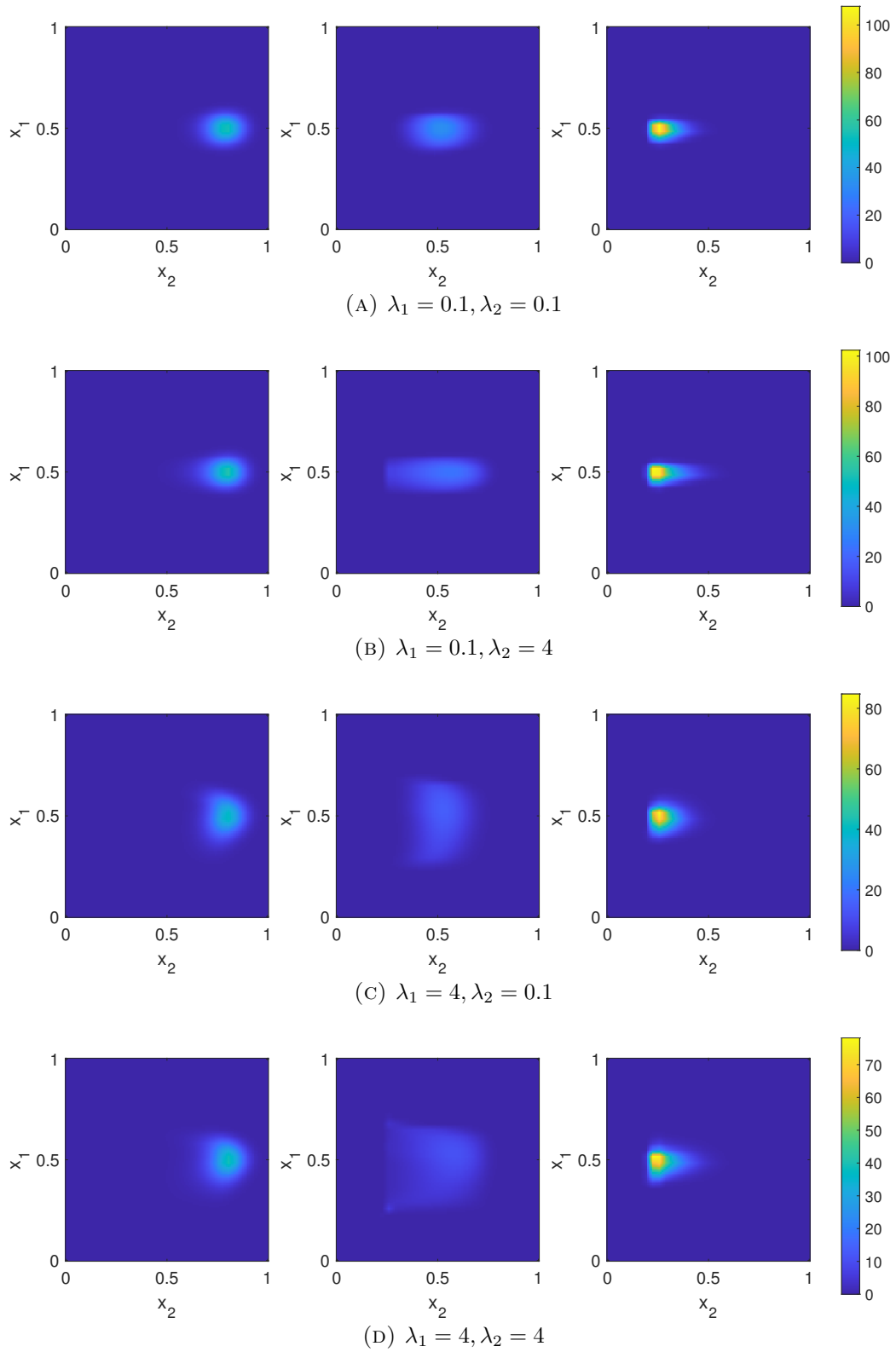
(D) $\lambda_1 = 4, \lambda_2 = 4$

FIG. 1. **Maximal spread.** MFG solution $\rho(x, t)$ at $t = 0.1, 0.5, 0.9$ with different choice of λ_i

where $Q(x)$ takes on extremely high values on the four rectangular regions in Figure 2 and 0 elsewhere. Thus, Q models four static rectangular obstacles. Finally, we choose $n = 3$ to approximate the kernel.

We have computed the MFG solutions for four choices of parameters

$$(\sigma_1, \sigma_2, \mu) \in \{(0.8, 0.8, 0.1), (0.2, 0.2, 5), (0.5, 0.2, 5), (0.2, 0.5, 5)\}$$

The results are shown in Figure 2. As we can see, agents travel through the channels created by $Q(x)$ to avoid high cost. Recall that small σ_i yields strong repulsion in x_i direction, which results in different behavior by agents. For instance, in Figure 2 (C) we impose a strong repulsion in x_2 direction and see horizontally elongated density evolution.

6.3. Gaussian repulsion with dynamic obstacles.
Next we consider a MFG model with Gaussian repulsion on $\Omega \times [0, T] = [-1, 1]^2 \times [0, 1]$ with dynamic obstacles. We set

$$\rho_0(x) = \frac{1}{5} \sum_{j=1}^{5} \rho_G(c_j, -0.9, \sigma_G), \quad \sigma_G = 0.2, \ c_j = -1.2 + 0.4j, \ 1 \leq j \leq 5$$

To model dynamic obstacles, we set

$$L(x, v) = \frac{1}{2} \|v\|^2 + 10^3 \left(\max \left(|x_1 - 0.5|, |x_2 - 0.5| \right) \right)^8 + Q(x, t)$$

where Q now represents time-dependent rectangular obstacles that move vertically. The rest of the parameters are the same as in the previous section. The results are shown in Figure 3. Again, values of σ_1, σ_2 control how spread is the solution in x_1, x_2 directions.

We also note that the computational cost for the static and dynamic models is the same.

6.4. Interactions in sub-regions.
Next, we consider a MFG model with a Gaussian repulsion on $\Omega \times [0, T] = [-1, 1]^2 \times [0, 1]$ where agents interact only within domains $\Omega_1 = \{(x_1, x_2) : x_1 \leq 0\}$ and $\Omega_2 = \{(x_1, x_2) : x_1 > 0\}$. This means that agents in Ω_i interact only with those in Ω_i. We set

$$g(x_1, x_2) = -4 \exp \left(-5 \left(x_1 - 0.0 \right)^2 - 2.5(x_2 - 0.5)^2 \right)$$

$$\rho_0(x) = \frac{1}{2} \rho_G(0.2, -0.9, \sigma_G) + \frac{1}{2} \rho_G(-0.2, -0.9, \sigma_G)$$

The rest of the parameters are the same as for previous examples with Gaussian repulsion. We apply the basis modification explained in Section 4.2 to compute the solution. The results are shown in Figure 4 where we have also included the solution with same data but full interaction. In both cases, densities spread before concentrating at the desired location. However, in the sub-region interaction case, Figure 4 (A), there is a concentration of agents along the common boundary of Ω_1, Ω_2. The reason is that agents on different sides of this boundary do not interact with each other, so they do not mind congestion.

6.5. Differential-operator interactions.
Throughout this section, we set $V = \mu(I - \Delta)^{-2}$, $\mu > 0$ and $\Omega = \mathbb{T}^d$. In [5], the authors solve a stationary MFG system

$$\begin{cases} H(x, \nabla\phi) = V[\rho] + \lambda \\ -\nabla \cdot (\rho \nabla_p H(x, \nabla\phi)) = 0 \\ \int \rho = 1, \ \rho \geq 0, \ \lambda \in \mathbb{R} \end{cases} \tag{6.1}$$

(A) $\sigma_1 = 0.8, \sigma_2 = 0.8, \mu = 0.1$

(B) $\sigma_1 = 0.2, \sigma_2 = 0.2, \mu = 5$

(C) $\sigma_1 = 0.5, \sigma_2 = 0.2, \mu = 5$
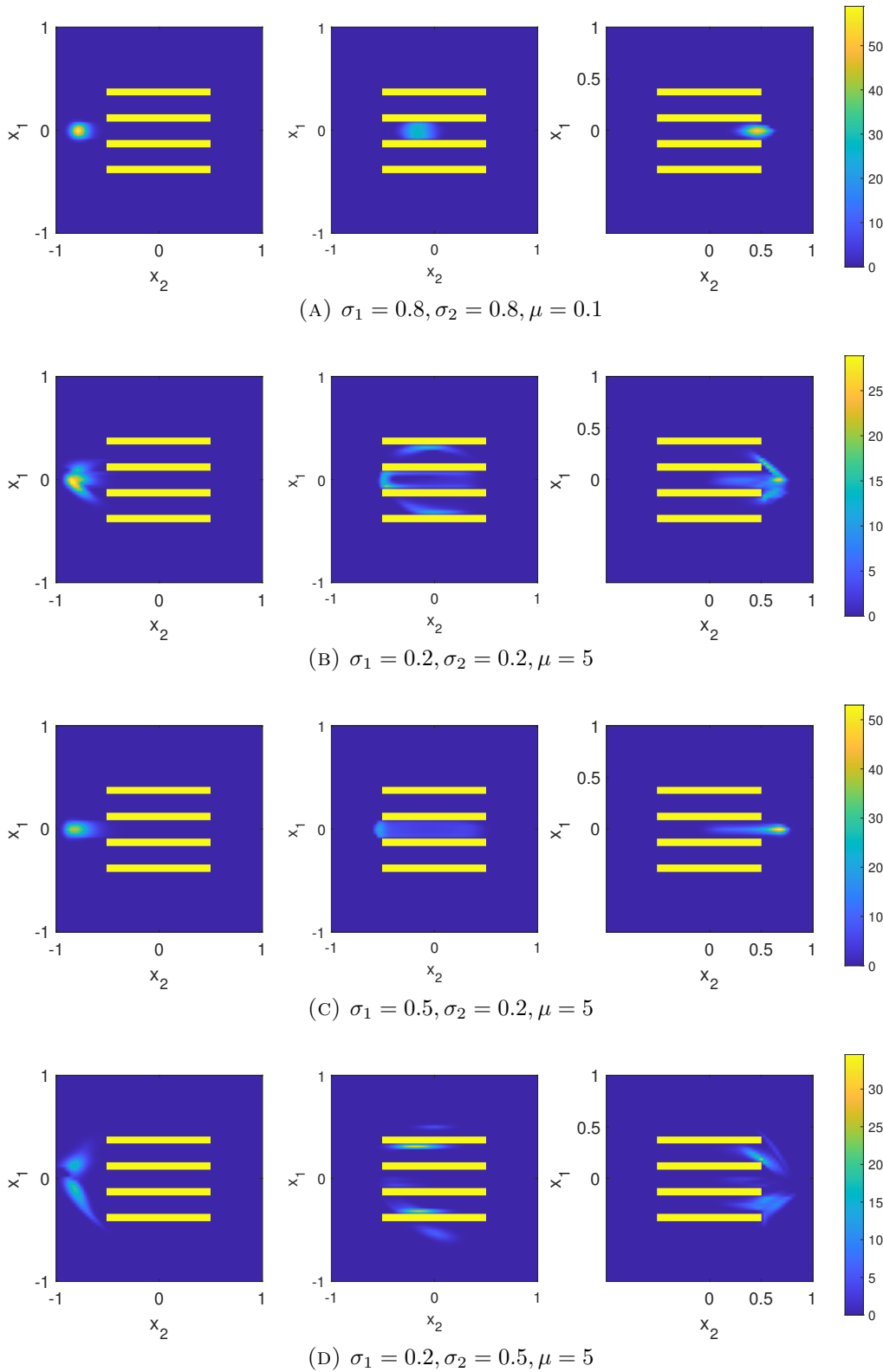
(D) $\sigma_1 = 0.2, \sigma_2 = 0.5, \mu = 5$

FIG. 2. **Gaussian repulsion with static obstacles.** MFG solution $\rho(x,t)$ at $t = 0.1, 0.5, 0.9$ with different Gaussian parameters $(\sigma_1, \sigma_2, \mu)$, where bright yellow rectangles represents static obstacles.
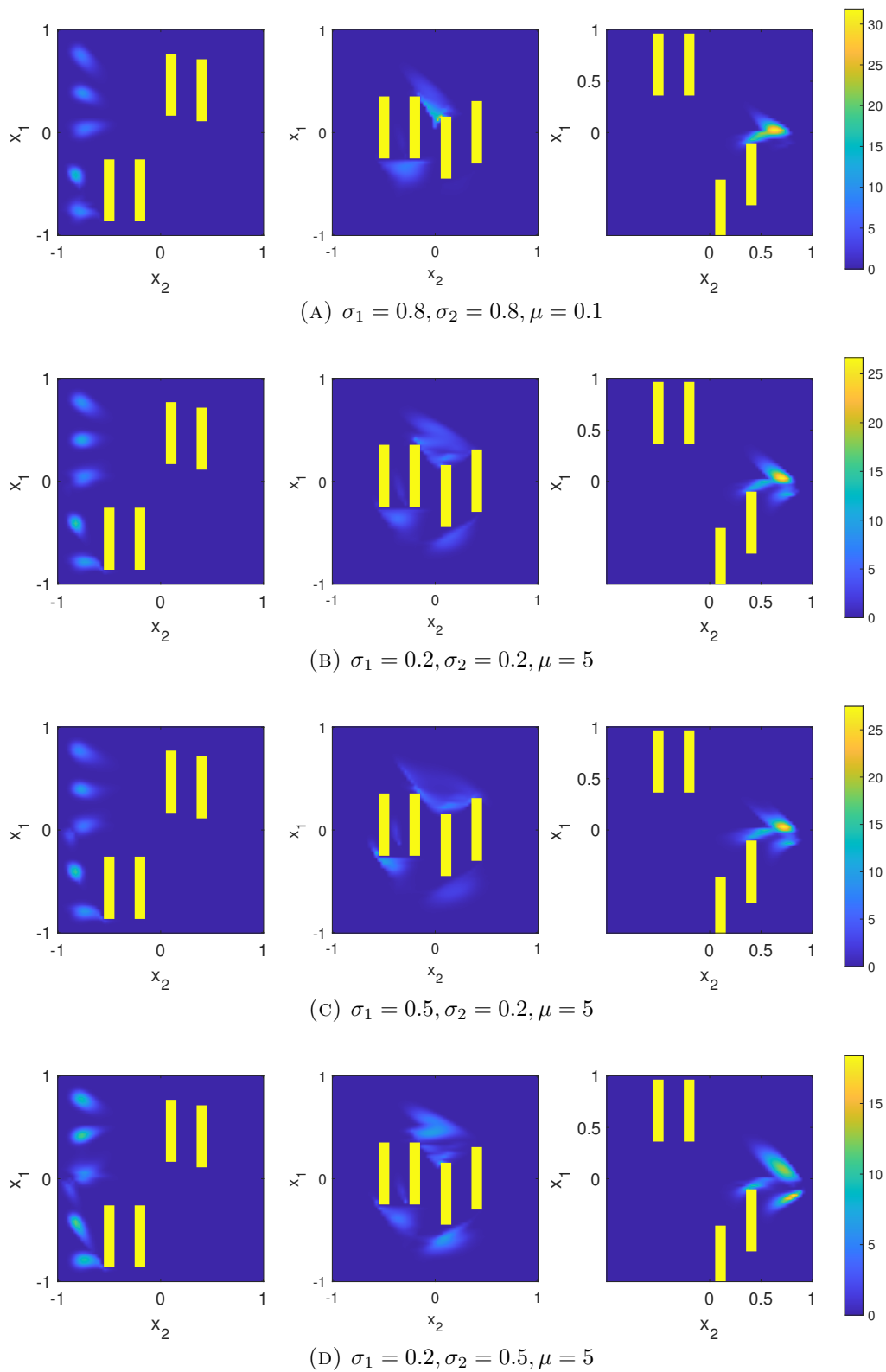
(A) $\sigma_1 = 0.8, \sigma_2 = 0.8, \mu = 0.1$

(B) $\sigma_1 = 0.2, \sigma_2 = 0.2, \mu = 5$

(C) $\sigma_1 = 0.5, \sigma_2 = 0.2, \mu = 5$

(D) $\sigma_1 = 0.2, \sigma_2 = 0.5, \mu = 5$

FIG. 3. **Gaussian repulsion with dynamic obstacles.** MFG solution $\rho(x,t)$ at $t = 0.1, 0.5, 0.9$ with different Gaussian parameters σ_1, σ_2, μ, where bright yellow rectangles represent obstacles moving along x_1 direction.
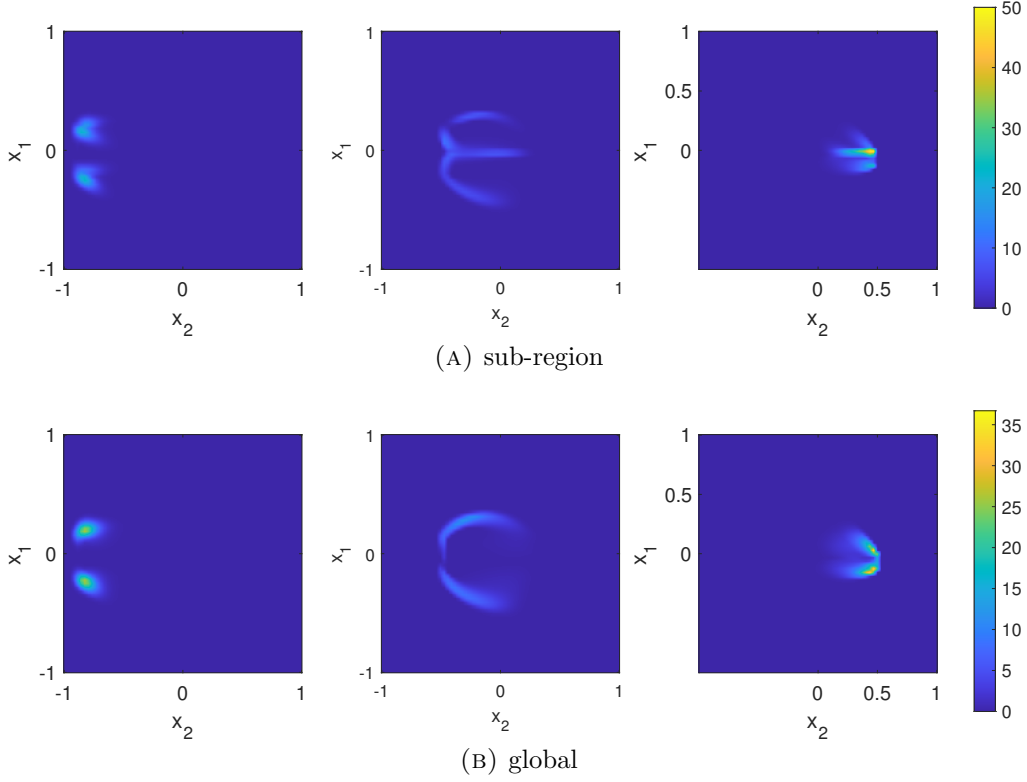
(A) sub-region

(B) global

FIG. 4. **Sub-region interactions.** MFG solution $\rho(x, 0.1), \rho(x, 0.5), \rho(x, 0.9)$ for sub-region and global interactions with parameters $\sigma_1 = \sigma_2 = 0.2, \ \mu = 5$

by approximating it with its second-order version

$$\begin{cases} -\sigma\Delta\phi + H(x, \nabla\phi) = V[\rho] + \lambda \\ -\sigma\Delta\rho - \nabla\cdot(\rho\nabla_p H(x, \nabla\phi)) = 0 \\ \int \rho = 1, \ \rho \geq 0, \ \lambda \in \mathbb{R} \end{cases} \tag{6.2}$$

for small $\sigma > 0$. Here, we recover the results in [5] using our method. Since we consider first-order time-dependent systems instead of second-order stationary ones, we apply a different approximation procedure for (6.1).

To approximate (6.1) we use the long-time convergence, or the turnpike property, of MFG systems discussed in [15]. More precisely, we approximate (6.1) by

$$\begin{cases} -\psi_t + H(x, \nabla\psi) = V[\nu] \\ \nu_t - \nabla\cdot(\nu\nabla_p H(x, \nabla\psi)) = 0 \\ \nu(x, 0) = \rho_0(x), \ \psi(x, T) = g(x) \end{cases} \tag{6.3}$$

where $\rho_0 \in L^\infty(\mathbb{T}^d), \ g \in C^2(\mathbb{T}^d)$ and $T > 0$ is large. To formulate the convergence results, we need to scale the time variable in (6.3) and obtain a problem on a time-interval $[0, 1]$. For that, we write $\psi(x, t) = \phi(x, \frac{t}{T}), \ \nu(x, t) = \rho(x, \frac{t}{T})$, and (6.3) becomes

$$\begin{cases} -\phi_t + T \cdot H(x, \nabla\phi) = TV[\rho] \\ \rho_t - \nabla\cdot\left(\rho\nabla_p(T \cdot H(x, \nabla\phi))\right) = 0 \\ \rho(x, 0) = \rho_0(x), \ \phi(x, 1) = g(x) \end{cases} \tag{6.4}$$

Furthermore, a triple $(\bar\phi, \bar\rho, \bar\lambda)$ is solution of (6.1) if $\bar\phi$ is a Lipschitz viscosity solution of the HJB in (6.1), $\nabla\bar\phi$ exists $\bar\rho$ a.e., and the continuity equation in (6.1) is satisfied in the sense of distributions. We summarize the results from [15] in the following theorem. We omit assumptions and technicalities and refer to the original paper for details.

Theorem 6.1 ([15]). *Under suitable assumptions,*
1. *system* (6.1) *has at least one solution. Moreover, if* $(\bar\phi_1, \bar\rho_1, \bar\lambda_1)$ *and* $(\bar\phi_2, \bar\rho_2, \bar\lambda_2)$ *are solutions, then* $\bar\lambda_1 = \bar\lambda_2$, *and* $V[\bar\rho_1] = V[\bar\rho_2]$.
2. *for a solution* $(\bar\phi, \bar\rho, \bar\lambda)$ *of* (6.1) *one has that*

$$\sup_{t\in[0,1]} \left\| \frac{\phi_T(\cdot, t)}{T} - \bar\lambda(1-t) \right\|_{L^\infty(\mathbb{T}^d)} \le \frac{C}{T^{\frac{1}{2}}},$$

and

$$\int_0^1 \|V[\rho_T(\cdot, t)] - V[\bar\rho]\|_{L^\infty(\mathbb{T}^d)} dt \le \frac{C}{T^{\frac{1}{2}}},$$

where (ϕ_T, ρ_T) *is the solution of* (6.4).

Therefore, to approximate solutions of (6.1) we need to solve (6.4). We take $H(x,p) = \frac{|p|^2}{2} - Q(x)$ where Q is a smooth periodic function. In this case, one can easily verify that assumptions in [15] are fulfilled.

As mentioned in the theorem above, a solution $\bar\phi$ in (6.1) is not necessarily unique even up to constants, whereas $\bar\lambda$ and $V[\bar\rho]$ are. However, for $V = \mu(I - \Delta)^{-2}$ the uniqueness of $V[\bar\rho]$ implies that of $\bar\rho$. Furthermore, $\lim_{T\to\infty} \|V[\rho_T(\cdot, t)] - V[\hat\rho]\|_{L^\infty(\mathbb{T}^d)} = 0$ implies a weak convergence $\rho_T(\cdot, t) \rightharpoonup \bar\rho$. Hence Theorem 6.1 guarantees that for a large set of times $t \in [0,1]$ the solution $\rho_T(\cdot, t)$ of (6.4) converges weakly to a well defined limit $\bar\rho$.

As in [5] we take $d = 2, \mu = 200$, and

$$Q(x_1, x_2) = -\sin(2\pi x_2) - \sin(2\pi x_1) - \cos(4\pi x_1)$$

We approximate V as in Section 4.3 using trigonometric polynomials up to order $n = 2$. Additionally, we set $T = 10$, $\rho_0(x) = 1$, $g(x) = 0$. The results are shown in Figure 5 where we plot $\rho_T(x, t)$. We also plot $\phi_T(x,t) - \int_{\mathbb{T}^d} \phi_T(y,t)dy$ to test whether it approximates a solution $\bar\phi$ of (6.1). Latter holds for second-order problems but not the first-order ones.

As we can see, we obtain accurate reconstructions of Tests 5, 6 in [5]. Our solutions are slightly less diffused because we consider first-order equations as opposed to second-order ones in [5]. Additionally, we use $H(x,p) = \frac{|p|^2}{2} - Q(x)$ whereas the examples in [5] are computed for $H(x,p) = |p|^{\frac{3}{2}} - Q(x)$. Nevertheless, we believe that qualitative properties and shapes of the solutions do not alter much due to this difference.

References

[1] Yves Achdou. Finite difference methods for mean field games. In *Hamilton-Jacobi equations: approximations, numerical analysis and applications*, volume 2074 of *Lecture Notes in Math.*, pages 1–47. Springer, Heidelberg, 2013.

[2] Yves Achdou, Francisco J. Buera, Jean-Michel Lasry, Pierre-Louis Lions, and Benjamin Moll. Partial differential equation models in macroeconomics. *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.*, 372(2028):20130397, 19, 2014.

[3] Yves Achdou, Fabio Camilli, and Italo Capuzzo-Dolcetta. Mean field games: numerical methods for the planning problem. *SIAM J. Control Optim.*, 50(1):77–109, 2012.

[4] Yves Achdou, Fabio Camilli, and Italo Capuzzo-Dolcetta. Mean field games: convergence of a finite difference method. *SIAM J. Numer. Anal.*, 51(5):2585–2612, 2013.
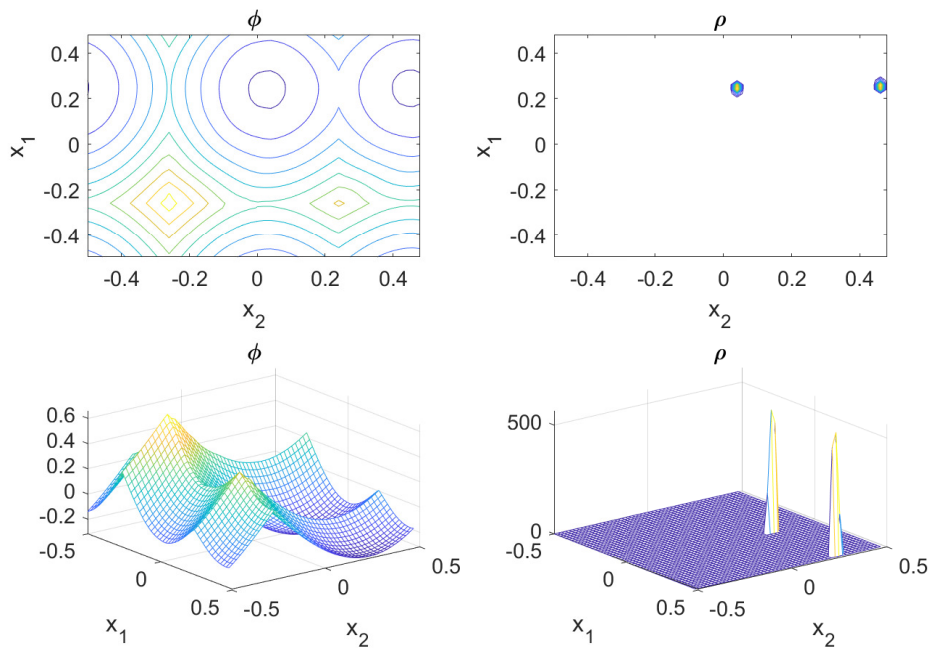
FIG. 5. The contours and graphs of $\phi_T(x,t) - \int_{\mathbb{T}^d} \phi_T(y,t)dy$ and $\rho_T(x,t)$ for $T = 10$ and $t = 0.4$.

[5] Yves Achdou and Italo Capuzzo-Dolcetta. Mean field games: numerical methods. *SIAM J. Numer. Anal.*, 48(3):1136–1162, 2010.

[6] Yves Achdou, Jiequn Han, Jean-Michel Lasry, Pierre-Louis Lions, and Benjamin Moll. Income and wealth distribution in macroeconomics: A continuous-time approach. Working Paper 23732, National Bureau of Economic Research, August 2017.

[7] Yves Achdou and Jean-Michel Lasry. Mean field games for modeling crowd motion. In *Contributions to partial differential equations and applications*, volume 47 of *Comput. Methods Appl. Sci.*, pages 17–42. Springer, Cham, 2019.

[8] Alexander Aurell and Boualem Djehiche. Mean-field type modeling of nonlocal crowd aversion in pedestrian crowd dynamics. *SIAM J. Control Optim.*, 56(1):434–455, 2018.

[9] J.-D. Benamou and G. Carlier. Augmented Lagrangian methods for transport optimization, mean field games and degenerate elliptic equations. *J. Optim. Theory Appl.*, 167(1):1–26, 2015.

[10] J.-D. Benamou, G. Carlier, and F. Santambrogio. Variational mean field games. In *Active particles. Vol. 1. Advances in theory, models, and applications*, Model. Simul. Sci. Eng. Technol., pages 141–171. Birkhäuser/Springer, Cham, 2017.

[11] Jean-David Benamou and Yann Brenier. A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. *Numer. Math.*, 84(3):375–393, 2000.

[12] L. Briceño Arias, D. Kalise, Z. Kobeissi, M. Laurière, Á. Mateos González, and F. J. Silva. On the implementation of a primal-dual algorithm for second order time-dependent mean field games with local couplings. In *CEMRACS 2017—numerical methods for stochastic models: control, uncertainty quantification, mean-field*, volume 65 of *ESAIM Proc. Surveys*, pages 330–348. EDP Sci., Les Ulis, 2019.

[13] L. M. Briceño Arias, D. Kalise, and F. J. Silva. Proximal methods for stationary mean field games with local couplings. *SIAM J. Control Optim.*, 56(2):801–836, 2018.

[14] Martin Burger, Marco Di Francesco, Peter A. Markowich, and Marie-Therese Wolfram. Mean field games with nonlinear mobilities in pedestrian dynamics. *Discrete Contin. Dyn. Syst. Ser. B*, 19(5):1311–1333, 2014.

[15] P. Cardaliaguet. Long time average of first order mean field games and weak KAM theory. *Dyn. Games Appl.*, 3(4):473–488, 2013.

[16] P. Cardaliaguet. Notes on mean field games, 2013. https://www.ceremade.dauphine.fr/ cardaliaguet/.

[17] Pierre Cardaliaguet and Charles-Albert Lehalle. Mean field game of controls and an application to trade crowding. *Math. Financ. Econ.*, 12(3):335–363, 2018.

[18] Philippe Casgrain and Sebastian Jaimungal. Algorithmic trading in competitive markets with mean field games. *SIAM News*, 52(2), 2019.

[19] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision*, 40(1):120–145, 2011.

[20] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal-dual algorithm. *Math. Program.*, 159(1-2, Ser. A):253–287, 2016.

[21] Y. T. Chow, J. Darbon, S. Osher, and W. Yin. Algorithm for overcoming the curse of dimensionality for time-dependent non-convex Hamilton-Jacobi equations arising from optimal control and differential games problems. *J. Sci. Comput.*, 73(2-3):617–643, 2017.

[22] A. De Paola, V. Trovato, D. Angeli, and G. Strbac. A mean field game approach for distributed control of thermostatic loads acting in simultaneous energy-frequency response markets. *IEEE Transactions on Smart Grid*, 10(6):5987–5999, Nov 2019.

[23] Weinan E, Jiequn Han, and Qianxiao Li. A Mean-Field Optimal Control Formulation of Deep Learning. *arXiv:1807.01083 [cs, math]*, 2018.

[24] D. Firoozi and P. E. Caines. An optimal execution problem in finance targeting the market trading speed: An mfg formulation. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 7–14, Dec 2017.

[25] Diogo A Gomes, Levon Nurbekyan, and Edgard A Pimentel. *Economic models and mean-field games theory*. IMPA Mathematical Publications. Instituto Nacional de Matemática Pura e Aplicada (IMPA), Rio de Janeiro, 2015.

[26] Diogo A. Gomes and J. Saúde. Mean field games models—a brief survey. *Dyn. Games Appl.*, 4(2):110–154, 2014.

[27] Diogo A. Gomes and J. Saúde. A mean-field game approach to price formation in electricity markets. *Preprint*, 2018. arXiv:1807.07088 [math.AP].

[28] Olivier Guéant, Jean-Michel Lasry, and Pierre-Louis Lions. Mean field games and applications. In *Paris-Princeton Lectures on Mathematical Finance 2010*, volume 2003 of *Lecture Notes in Math.*, pages 205–266. Springer, Berlin, 2011.

[29] M. Huang, P. E. Caines, and R. P. Malhamé. Large-population cost-coupled LQG problems with nonuniform agents: individual-mass behavior and decentralized ϵ-Nash equilibria. *IEEE Trans. Automat. Control*, 52(9):1560–1571, 2007.

[30] M. Huang, R. P. Malhamé, and P. E. Caines. Large population stochastic dynamic games: closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle. *Commun. Inf. Syst.*, 6(3):221–251, 2006.

[31] M. Jacobs and F. Léger. A fast approach to optimal transport: the back-and-forth method. *Preprint*, 2019. arXiv:1905.12154 [math.OC].

[32] Matt Jacobs, Flavien Léger, Wuchen Li, and Stanley Osher. Solving large-scale optimization problems with a convergence rate independent of grid size. *SIAM J. Numer. Anal.*, 57(3):1100–1123, 2019.

[33] Arman C. Kizilkale, Rabih Salhab, and Roland P. Malhamé. An integral control formulation of mean field game based large scale coordination of loads in smart grids. *Automatica*, 100:312 – 322, 2019.

[34] Aimé Lachapelle, Jean-Michel Lasry, Charles-Albert Lehalle, and Pierre-Louis Lions. Efficiency of the price formation process in presence of high frequency participants: a mean field game analysis. *Math. Financ. Econ.*, 10(3):223–262, 2016.

[35] Aimé Lachapelle and Marie-Therese Wolfram. On a mean field game approach modeling congestion and aversion in pedestrian crowds. *Transp. Res. Part B: Methodol.*, 45(10):1572 – 1589, 2011.

[36] Jean-Michel Lasry and Pierre-Louis Lions. Jeux à champ moyen. I. Le cas stationnaire. *C. R. Math. Acad. Sci. Paris*, 343(9):619–625, 2006.

[37] Jean-Michel Lasry and Pierre-Louis Lions. Jeux à champ moyen. II. Horizon fini et contrôle optimal. *C. R. Math. Acad. Sci. Paris*, 343(10):679–684, 2006.

[38] Jean-Michel Lasry and Pierre-Louis Lions. Mean field games. *Jpn. J. Math.*, 2(1):229–260, 2007.

[39] Wuchen Li, Ernest K. Ryu, Stanley Osher, Wotao Yin, and Wilfrid Gangbo. A parallel method for earth mover's distance. *J. Sci. Comput.*, 75(1):182–197, 2018.

[40] A. T. Lin, Y. T. Chow, and S. Osher. A splitting method for overcoming the curse of dimensionality in Hamilton-Jacobi equations arising from nonlinear optimal control and differential games with applications to trajectory generation. *Preprint*, 2018. arXiv:1803.01215v1 [math.OC].

[41] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2018. Second edition of [MR3057769].

[42] Levon Nurbekyan and J. Saúde. Fourier approximation methods for first-order nonlocal mean-field games. *Port. Math.*, 75(3-4):367–396, 2018.

[43] P. M. Welch, K. Ø. Rasmussen, and C. F. Welch. Describing nonequilibrium soft matter with mean field game theory. *The Journal of Chemical Physics*, 150(17):174905, 2019.

(S. L.) Department of Mathematics, UCLA
Email address: siting6@math.ucla.edu

(M. J.) Department of Mathematics, UCLA
Email address: majaco@math.ucla.edu

(W. L.) Department of Mathematics, UCLA
Email address: wcli@math.ucla.edu

(L. N.) Department of Mathematics, UCLA
Email address: lnurbek@math.ucla.edu

(S. J. O.) Department of Mathematics, UCLA
Email address: sjo@math.ucla.edu