

## Article

# A Mean-Field Game Control for Large-Scale Swarm Formation Flight in Dense Environments

Guofang Wang <sup>1,2,3</sup> , Wang Yao <sup>2,4,\*</sup> , Xiao Zhang <sup>1,2,3</sup> and Ziming Li <sup>1,2</sup> 

<sup>1</sup> School of Mathematical Sciences, Beihang University, Beijing 100191, China; wangguofang@buaa.edu.cn (G.W.); xiao.zh@buaa.edu.cn (X.Z.); zimingli@buaa.edu.cn (Z.L.)

<sup>2</sup> Key Laboratory of Mathematics, Informatics and Behavioral Semantics, Ministry of Education, Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China

<sup>3</sup> Peng Cheng Laboratory, Shenzhen 518055, China

<sup>4</sup> Institute of Artificial Intelligence, Beihang University, Beijing 100191, China

\* Correspondence: yaowang@buaa.edu.cn

**Abstract:** As an important part of cyberphysical systems (CPSs), multiple aerial drone systems are widely used in various scenarios, and research scenarios are becoming increasingly complex. However, planning strategies for the formation flying of aerial swarms in dense environments typically lack the capability of large-scale breakthrough because the amount of communication and computation required for swarm control grows exponentially with scale. To address this deficiency, we present a mean-field game (MFG) control-based method that ensures collision-free trajectory generation for the formation flight of a large-scale swarm. In this paper, two types of differentiable mean-field terms are proposed to quantify the overall similarity distance between large-scale 3-D formations and the potential energy value of dense 3-D obstacles, respectively. We then formulate these two terms into a mean-field game control framework, which minimizes energy cost, formation similarity error, and collision penalty under the dynamical constraints, so as to achieve spatiotemporal planning for the desired trajectory. The classical task of a distributed large-scale aerial swarm system is simulated by numerical examples, and the feasibility and effectiveness of our method are verified and analyzed. The comparison with baseline methods shows the advanced nature of our method.



**Citation:** Wang, G.; Yao, W.; Zhang, X.; Li, Z. A Mean-Field Game Control for Large-Scale Swarm Formation Flight in Dense Environments.

*Sensors* **2022**, *22*, 5437. <https://doi.org/10.3390/s22145437>

Academic Editors: Rajan Shankaran, Wei Ni, Xiaojing Chen and Bochun Wu

Received: 19 June 2022

Accepted: 18 July 2022

Published: 21 July 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As an important part of cyberphysical systems (CPSs), multiple aerial drone systems have shown great advantages and value in performing tasks such as environmental reconnaissance, information transmission & reception, and target tracking in the fields of disaster rescue, geological exploration, and smart cities [1]. Compared with common multi unmanned aerial vehicle (UAV) systems, hundreds or thousands of larger-scale UAV swarms have better maneuverability performance and mission realization efficiency [2]. When the drone swarm performs actual missions, it will fly in formation as needed. In the process of approaching the target, it not only needs to avoid obstacles in time but also needs to consider the trajectory collision avoidance between the drones in the swarm. As a result, the drone swarm requires a large amount of communication and computing resources to guarantee real-time communication information interaction and perception information calculation, which will grow exponentially with the scale.

Research work on the flight control of aerial swarms is extensive. A large number of research studies focus on the formation flight, trajectory planning, and obstacle avoidance of the swarm, and their combination forms are more common. Specifically, Zhang et al. [3] proposed a formation control method based on the communication consensus mechanism

and leader–follower strategy, which realized the efficient flight of multiple UAVs in formation around a single static obstacle. In the same year, Zhang et al. [4] proposed an obstacle avoidance control algorithm based on virtual structure and 3-D spatial leader–follower strategy, which realized a triangular formation of multiple UAVs flying through multiple static obstacles with high efficiency. Shao et al. [5] proposed an improved particle swarm optimization algorithm for formation-obstacle avoidance-trajectory planning in a 3-D environment and completed an experiment in which multiple UAVs formed a formation after avoiding the static cone and ellipsoid obstacles. Combined with an improved artificial potential field method (APF), Wu et al. [6] performed a swarm control for multiple UAVs based on a virtual core structure, solved the local minima problem in APF, and successfully achieved the swarm control of multiple UAVs and the flexible obstacle avoidance of adaptive formation flight in an urban environment. In 2021, the Fei Gao team [7] proposed a distributed swarm trajectory optimization method for formation flight in a dense environment, which realizes multiple UAVs maintain formation flight through unknown obstacle-rich scenarios. Peng et al. [8] put forward a perception sharing and swarm trajectory global optimal algorithm, which experimentally showed that the success rate of obstacle avoidance in a dense obstacle environment could reach at least 80%. In the latest research so far, the formation-obstacle avoidance scenarios of multi-UAV systems have become more and more complex, and the continuously developed optimal control algorithms provide decision-making guarantees for mission flight in the corresponding scenarios. However, the current situation shows that the research schemes in the above literature are all limited by the large-scale of UAVs because the traditional optimization and control technologies deal with the dynamic interaction between individuals, respectively. With the increase of the number of UAVs, the difficulty of solving the optimal strategy will increase significantly.

For the large-scale UAVs flight control problem, some researchers have contributed interesting work. For example, [9] used the mean-field game (MFG) model to solve the real-time flight control problem of large-scale UAVs on the 2-D plane, and the proposed algorithm effectively realizes interaircraft collision avoidance and reduces the energy consumption of UAVs. Shiri et al. [10] combined a federal learning method that can share the parameters of neural network models on UAVs and completed a simulation experiment based on the MFG model for large-scale-UAV swarms flying to the destination under 2-D conditions. Chen et al. [11] proposed a mean-field trust-region policy optimization control method based on multi-agent reinforcement learning, which solved the problem of limited communication range for large-scale UAV flight control. Xu et al. [12] proposed a dual-fields approach, in which the improved APF method overcomes the failure of traditional APF in collision avoidance; the MFG method greatly reduces the communication interference of the UAV group, couples these two methods to adjust the flight trajectory and power consumption, and realizes the flight obstacle avoidance of large-scale UAVs in multiple static obstacles with height differences. Gao et al. [13] proposed an energy-efficient velocity control algorithm for a large number of UAVs based on MFG, which expressed the speed control of large-scale UAVs as a differential game and used the original double mixed gradient method to solve the problem. Their experiments completed that large-scale UAVs bypassed static obstacles in the case of 2-D while minimizing energy consumption. The above documents all use the mean-field method to convert the traditional control 1-vs.-N game into a 1-vs.-1 game, which greatly reduces the interaction frequency of agent systems. Thus, they can deal with large-scale mission scenarios to effectively reduce the communication resources and flight energy consumption. However, it should be pointed out that the state quantities in their numerical solution of the mean-field are all low-dimensional, and the relevant numerical solution problems can be solved using the grid-based method. In terms of solving high-dimensional MFGs, [14] perfectly avoids spatial grids of high-dimensional MFGs based on the machine learning framework, but their work is limited to the deterministic setting ( $\sigma = 0$ ). Ref. [15] is the first document to solve high-dimensional MFGs under a random setting ( $\sigma > 0$ ), which exploits the natural connection between MFGs and generative adversarial neural networks (GANs) [16]. Ref. [17]

further proposed a numerical solution for solving multi-population high-dimensional stochastic MFGs, a coupled alternating neural network (CA-Net). The numerical experiments completed the coordinated flight of large-scale multipopulation quadcopters to the destination and realize the intergroup interaction and intragroup collision avoidance. The MF method, MFG model, and its numerical solution methods gathered from these documents provide a preliminary solution idea for the flight control problem of large-scale aerial drones in 3-D scenes. However, the current situation shows that these large-scale control research schemes are limited by complex scenarios because most of the state quantities in MF terms are directly used for real-time error measurement. With the complexity of application scenarios, a large number of UAV states may change frequently, resulting in a significant decline in mission success rate.

To summarize, a large-scale agent trajectory planning method that can effectively manage both formation and obstacle avoidance in dense environments is lacking in the literature. On the one hand, obstacle-dense avoidance control for aerial swarm navigation in formation needs to break through the large-scale restrictions; on the other hand, large-scale UAV flight control methods cannot be directly applied to complex scenarios. In practice, a single agent will stay away from obstacles for safety. However, when formation imposes agents tracking targets, it may oppose obstacle avoidance, that is, sometimes formation and obstacle avoidance are contradictory; large-scale itself has the difficulty of communication and calculation, which is exacerbated by frequent formation–neighbor communication and obstacle avoidance environment interactions. How to systematically trade off the conflicting requirements of large-scale, formation, and obstacle avoidance is the key point to accomplishing large-scale noncolliding formation flights.

To bridge the gap, we propose a large-scale agent trajectory optimization method capable of navigating large-scale swarms in formations while avoiding dense obstacles. First, extending the formation method in [7], we model the formation using undirected graphs based on probability densities and define a mean-field-based differentiable metric that assesses the difference between large-scale formation shapes in 3-D workspaces. Our formation similarity metric also retains the translation, rotation, and scale invariance of the formation methods in [7], enabling quantitative evaluation of the overall performance of large-scale formation maintenance. To solve the communication and computing difficulties caused by complex scenarios of large-scale multiagent systems and formulate the trade-offs between formation and obstacle avoidance in large-scale contexts, we design a mean-field game control framework that simultaneously satisfies the dynamic constraints and minimizes energy cost, formation similarity error, and collision penalty. The latter two of them serve as mean-field terms directly related to complex environments, and the functional form we define provides greater flexibility for agent maneuvering. Finally, to verify the effectiveness and practicability of the method, we conduct extensive simulation experiments of distributed aerial large-scale UAVs.

Our main contributions are summarized as:

- A differentiable graph-theory-based mean-field term that quantifies the similarity distance between large-scale three-dimensional formations; a differentiable ellipsoid-based mean-field term that inscribes the potential energy value of dense three-dimensional obstacles.
- A general control framework for complex scenarios of large-scale multiagent systems—mean-field game control, which jointly takes the amount of communication and computation, operating energy consumption, formation similarity, and obstacle avoidance into account.
- A series of simulations with a distributed large-scale aerial swarm system validates the efficiency and robustness of our method. The comparison with baseline methods shows the advanced nature of our method.

## 2. Two Types of Mean-Field Terms

An aerial swarm formed by a large number of UAVs is considered in this paper, which is traversing an obstacle-rich area in an expected formation. Herein, two types of differentiable mean-field terms for formation similarity metric and obstacle potential energy value engraving are constructed in the following two subsections.

### 2.1. Formation Mean-Field Term

The large-scale system  $\mathcal{N}$  with  $N$  agents have a probability distribution of the spatial state  $\mathbf{x}$  varying with time  $t$ , that is, the mean field  $\rho(\mathbf{x}, t)$ , where  $\mathbf{x} = [x_1, x_2, x_3, \dots, x_n] \in \mathbb{R}^n \sim \rho$ . Usually the position vector  $\mathbf{p} = [x, y, z] = [x_1, x_2, x_3]$ . A formation of  $N$  agents is modeled by an undirected graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} := \{1, 2, \dots, N\}$  is the set of vertices, and  $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$  is the set of edges [7]. In graph  $\mathcal{G}$ , the vertex  $i$  represents the  $i^{\text{th}}$  agent with position vector  $\mathbf{p}_i = [x_i, y_i, z_i] \in \mathbb{R}^3$ . An edge  $e_{ij} \in \mathcal{E}$  that connects vertex  $i \in \mathcal{V}$  and vertex  $j \in \mathcal{V}$  means the agent  $i$  and  $j$  can measure the geometric distance between each other. In this work, each agent communicates with all other agents, thus, the formation graph  $\mathcal{G}$  is complete. Each edge of the graph  $\mathcal{G}$  is associated with a non-negative number as a weight. The weight of edge  $e_{ij}$  is given by

$$w_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|^2, \quad (i, j) \in \mathcal{E}, \quad (1)$$

where  $\|\cdot\|$  denotes the Euclidean norm. Now, the adjacency matrix  $\mathbf{A} = (A_{ij}) \in \mathbb{R}^{N \times N}$  and degree matrix  $\mathbf{D} = (D_{ij}) \in \mathbb{R}^{N \times N}$  of the formation  $\mathcal{G}$  are determined,

$$A_{ij} = w_{ij}, \quad (2)$$

$$D_{ij} = \begin{cases} \sum_j A_{ij} & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Thus, the corresponding Laplacian matrix is given by

$$\mathbf{L} = \mathbf{D} - \mathbf{A}. \quad (4)$$

With the above matrices, the symmetric normalized Laplacian matrix of the graph  $\mathcal{G}$  is defined as

$$\hat{\mathbf{L}} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}, \quad (5)$$

where  $\mathbf{I} \in \mathbb{R}^{N \times N}$  is the identity matrix.

As a graph representation matrix, Laplacian contains information about the graph structure [18]. To achieve the expected formation for the large-scale swarm, we propose a formation similarity metric as

$$\mathcal{F}^1(\rho(\mathbf{x}, t)) = \int_{\Omega} \int_{\Omega} f^1(\mathbf{x}, \hat{\mathbf{x}}) d\rho(\mathbf{x}, t) d\rho(\hat{\mathbf{x}}, t), \quad (6)$$

$$f^1(\mathbf{x}, \hat{\mathbf{x}}) = \|\hat{\mathbf{L}}(\mathbf{x}) - \hat{\mathbf{L}}_e(\hat{\mathbf{x}})\|_F^2 = \text{tr}\left\{(\hat{\mathbf{L}} - \hat{\mathbf{L}}_e)^T (\hat{\mathbf{L}} - \hat{\mathbf{L}}_e)\right\}, \quad (7)$$

where  $\Omega$  denotes state space of  $\mathbf{x}$ ,  $\text{tr}\{\cdot\}$  is the trace of a matrix,  $\hat{\mathbf{L}}$  is the symmetric normalized Laplacian of the current swarm formation, and  $\hat{\mathbf{L}}_e$  is the counterpart of the expected formation. Frobenius norm  $\|\cdot\|_F$  is used in our distance metric.

**Remark 1.** The function  $\mathcal{F}^1$  is invariant to the translation, rotation, and scale of the formation because  $\mathcal{F}^1$  is an integral of  $f^1$ .  $f^1$  is natively invariant to translation and rotation of the formation since the corresponding graph is weighted by the absolute distance between agent positions; scaling invariance is achieved by normalizing graph Laplacian with the degree matrix in (5) [7].

## 2.2. Obstacle Mean-Field Term

The large-scale system  $\mathcal{N}$  with  $N$  UAVs needs to traverse an obstacle-rich area. There are  $K$  static obstacles. The position of UAV is available. We assume that obstacles can be detected by the UAV vision sensor. In a 3-D urban environment, without loss of generality, obstacles can be considered rectangular solids  $\Omega_{obs,k} = \{(x, y, z) | |x - x_{0,k}| \leq v_{1,k}, |y - y_{0,k}| \leq v_{2,k}, |z - z_{0,k}| \leq v_{3,k}\}$ ,  $k = 1, \dots, K$  where  $(x_{0,k}, y_{0,k}, z_{0,k})$  is the center of corresponding obstacle,  $(\pm v_{1,k}, \pm v_{2,k}, \pm v_{3,k})$  are its vertices that are parallel to the  $x, y, z$ -axis, respectively. For training, we formulate obstacles as

$$\Omega_{obs,trn,k} : Q_k(x, y, z) = \frac{1}{3v_{1,k}^2}(x - x_{0,k})^2 + \frac{1}{3v_{2,k}^2}(y - y_{0,k})^2 + \frac{1}{3v_{3,k}^2}(z - z_{0,k})^2 \leq 1.1 \quad k = 1, \dots, K. \quad (8)$$

We encode ellipsoidal repulsion of obstacles for the generic UAV as

$$f^2(\mathbf{x}) = \begin{cases} \sum_{k=1}^K \gamma_{obs,k} \frac{1}{Q_k} & \text{if } \mathbf{x} \in \Omega_{obs,trn} \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

where  $\gamma_{obs,k}$ ,  $k=1, \dots, K$ , is repulsive force gain coefficient between the generic UAV and other obstacles,  $\Omega_{obs,trn} = \bigcup_{k=1}^K \Omega_{obs,trn,k}$ .

**Remark 2.** The cuboid obstacle repulsion is encoded as the unit circumscribed ellipsoid repulsion making this function differentiable [19].  $\Omega_{obs,trn}$  contains an obstacle radial bound ten percent more than in circumscribed ellipsoid of  $\Omega_{obs}$  because we found this additional training buffer alleviates collisions during validation. By training with ellipsoidal repulsion, which has gradient information within the obstacles, we incentivize the model to learn trajectories avoiding the obstacles [20].

To achieve all obstacles avoidance for the large-scale swarm, we propose an obstacle potential energy function as

$$\mathcal{F}^2(\rho(\mathbf{x}, t)) = \int_{\Omega} f^2(\mathbf{x}) \rho(\mathbf{x}, t) d\mathbf{x}. \quad (10)$$

**Remark 3.** The function  $\mathcal{F}^2$  can avoid dense obstacles for large-scale swarms, which benefits from the proper construction of 3-D obstacles on the one hand, and the organic combination of obstacle potential and mean-field method on the other hand.

## 3. A Mean-Field Game Control Framework for Complex Scenarios

In this section, we find the optimal state control for large-scale UAVs which can systematically weigh the energy consumption, formation similarity, and obstacle avoidance of each UAV when it travels complex scenarios. Specifically, we first formulate the optimal control problem for a single UAV. The interaction relationship between drones forms an N-agent noncooperative differential game. As the number of UAVs grows large, the complexity of solving the differential game in the optimal control problem can increase significantly as traditional methods need to separately deal with the increasing interactions. Therefore, we reformulate the large-scale UAVs optimal state control problem in complex scenarios as a mean-field game control problem. Subsequently, we propose a generative-adversarial-network (GAN)-based algorithm to solve the mean-field game.

### 3.1. Single-UAV Optimal Control Problem

We consider that the continuous-time state  $\mathbf{x}$  of the  $i$ th UAV has the dynamic-kinematic equation of the following form

$$d\mathbf{x}_i = h_i(\mathbf{x}_i, \mathbf{x}_i^-, \mathbf{u}_i) dt + \sigma d\omega_i(t), \quad (11)$$

where  $\mathbf{x}_i^-$  denotes the states of all other UAVs,  $\mathbf{u}_i \in \mathcal{U}_i$  is the control input (strategy). In the stochastic case, we add a noise term to the dynamics,  $\sigma$  is a volatility term—a fixed

coefficient matrix, and  $\omega_i(t)$  denotes a Wiener process (standard Brownian motion), which is identical and independent among all UAVs. The interpretation here is that we are modeling the situation when the quadcopter suffers from noisy measurements [15]. For example, under wind perturbations,  $\sigma$  can be the covariance matrix of the wind velocity [10,13].

Without loss of generality, denoting gravity as  $g$ , the acceleration of a UAV with mass  $m$  can be written as

$$\begin{cases} \ddot{x}_i = \frac{u_i}{m} (\sin(\psi_i) \sin(\varphi_i) + \cos(\psi_i) \sin(\theta_i) \cos(\varphi_i)) \\ \ddot{y}_i = \frac{u_i}{m} (-\cos(\psi_i) \sin(\varphi_i) + \sin(\psi_i) \sin(\theta_i) \cos(\varphi_i)) \\ \ddot{z}_i = \frac{u_i}{m} \cos(\theta_i) \cos(\varphi_i) - g \\ \ddot{\psi}_i = \tau_{\psi_i} \\ \ddot{\theta}_i = \tau_{\theta_i} \\ \ddot{\varphi}_i = \tau_{\varphi_i} \end{cases}, \quad (12)$$

where  $(x, y, z)$  is the spatial position of the UAV,  $(\psi, \theta, \varphi)$  is the angular orientation with corresponding torques  $\tau_\psi, \tau_\theta, \tau_\varphi$ , and  $u$  is the main thrust directed out of the bottom of the aircraft [21]. To fit a control framework, the above second-order system can be turned into a first-order system

$$\dot{\mathbf{x}}_i = h_i(\mathbf{x}_i, \mathbf{x}_i^-, \mathbf{u}_i) \implies \begin{cases} \dot{x}_i = v_{xi} \\ \dot{y}_i = v_{yi} \\ \dot{z}_i = v_{zi} \\ \dot{\psi}_i = v_{\psi_i} \\ \dot{\theta}_i = v_{\theta_i} \\ \dot{\varphi}_i = v_{\varphi_i} \\ \dot{v}_{xi} = \frac{u_i}{m} (\sin(\psi_i) \sin(\varphi_i) + \cos(\psi_i) \sin(\theta_i) \cos(\varphi_i)) \\ \dot{v}_{yi} = \frac{u_i}{m} (-\cos(\psi_i) \sin(\varphi_i) + \sin(\psi_i) \sin(\theta_i) \cos(\varphi_i)) \\ \dot{v}_{zi} = \frac{u_i}{m} \cos(\theta_i) \cos(\varphi_i) - g \\ \dot{v}_{\psi_i} = \tau_{\psi_i} \\ \dot{v}_{\theta_i} = \tau_{\theta_i} \\ \dot{v}_{\varphi_i} = \tau_{\varphi_i} \end{cases}, \quad (13)$$

where  $\mathbf{x} = [x, y, z, \psi, \theta, \varphi, v_x, v_y, v_z, v_\psi, v_\theta, v_\varphi]^T \in \mathbb{R}^{12}$  is the state with velocities  $\mathbf{v}$ , and  $\mathbf{u} = [u, \tau_\psi, \tau_\theta, \tau_\varphi]^T \in \mathbb{R}^4$  is the control.

For the  $i$ th UAV, we consider the cost function of the following form

$$\mathcal{J}_i(\mathbf{x}_i, \mathbf{x}_i^-, \mathbf{u}_i) = \int_0^T L_i(\mathbf{x}_i, \mathbf{u}_i) + \mathcal{F}_i(\mathbf{x}_i, \mathbf{x}_i^-) dt + \mathcal{G}_i(\mathbf{x}_i^T), \quad (14)$$

where  $L_i$  is a running cost incurred by the  $i$ th UAV based solely on its actions,  $\mathcal{F}_i$  is a running cost incurred by the  $i$ th UAV based on its interactions with the rest of the swarm, and  $\mathcal{G}_i$  is a terminal cost incurred by the  $i$ th UAV based on its final state.

The running cost  $L_i$  is given by

$$L_i(\mathbf{x}_i, \mathbf{u}_i) = c_1 \|\mathbf{u}_i(\mathbf{x}_i, t)\|^2, \quad (15)$$

where  $c_1$  is a constant. The running cost  $L_i$  denotes the control effort implemented by the  $i$ th UAV. The terminal cost  $\mathcal{G}_i$  is given by

$$\mathcal{G}_i(\mathbf{x}_i^T) = c_2 \|\mathbf{x}_i^T - \mathbf{x}_0\|_2, \quad (16)$$

where  $c_2$  is a constant,  $\mathbf{x}_i^T$  is the final state of the  $i$ th UAV, and  $\mathbf{x}_0$  is the target state on which we want the UAVs to reach the objective. The terminal cost  $\mathcal{G}_i$  denotes the distance between UAV  $i$ 's target state and the desired state. The interaction cost  $\mathcal{F}_i$  is given by

$$\mathcal{F}_i(\mathbf{x}_i, \mathbf{x}_i^-) = l_1 \mathcal{F}_i^1(\mathbf{x}_i, \mathbf{x}_i^-, t) + l_2 \mathcal{F}_i^2(\mathbf{x}_i, \mathbf{x}_i^-, t) + l_3 \mathcal{F}_i^3(\mathbf{x}_i, \mathbf{x}_i^-, t), \quad (17)$$

where  $l_1, l_2, l_3$  are constants,  $\mathcal{F}_i^1(\mathbf{x}_i, \mathbf{x}_i^-, t)$  is UAV  $i$ 's formation cost,

$$\mathcal{F}_i^1(\mathbf{x}_i, \mathbf{x}_i^-, t) = \frac{1}{N} \sum_{i=1}^N \frac{1}{N} \sum_{j=1}^N \left\| \hat{\mathbf{L}}(\mathbf{x}_i, \mathbf{x}_i^-, t) - \hat{\mathbf{L}}_e(\hat{\mathbf{x}}_j, \hat{\mathbf{x}}_j^-, t) \right\|_F^2 \quad (18)$$

$\mathcal{F}_i^2(\mathbf{x}_i, \mathbf{x}_i^-, t)$  is UAV  $i$ 's obstacle avoidance cost,

$$\mathcal{F}_i^2(\mathbf{x}_i, \mathbf{x}_i^-, t) = \frac{1}{N} \sum_{i=1}^N \begin{cases} \sum_{k=1}^K \gamma_{obs,k} \frac{1}{Q_k(\mathbf{x}_i, t)} & \text{if } \mathbf{x}_i \in \Omega_{obs,trn} \\ 0 & \text{otherwise,} \end{cases} \quad (19)$$

$\mathcal{F}_i^3(\mathbf{x}_i, \mathbf{x}_i^-, t)$  is UAV  $i$ 's collision avoidance cost,

$$\mathcal{F}_i^3(\mathbf{x}_i, \mathbf{x}_i^-, t) = \frac{1}{N} \sum_{j \neq i}^N \mathbb{1}_{\|E^{\frac{1}{2}}(\mathbf{x}_i(t) - \mathbf{x}_i^-(t))\| \leq e_0}, \quad (20)$$

$$\mathbb{1}_{\mathbb{A}}(\xi) = \begin{cases} 1 & \text{if } \xi \in \mathbb{A} \\ 0 & \text{otherwise,} \end{cases}$$

where  $\mathbb{1}$  means the indicator function,  $e_0$  is the safe distance between UAVs,  $E = diag(1, 1, 1/c)$  transforms Euclidean distance into ellipsoidal distance. The interaction cost  $\mathcal{F}_i$  denotes the sum of formation maintenance and trajectory collision loss about UAV  $i$ .

In summary, the optimal control problem faced by UAV  $i$  is given by

$$\begin{aligned} & \inf_{\mathbf{u}_i \in \mathcal{U}_i} \mathcal{J}_i(\mathbf{x}_i, \mathbf{x}_i^-, \mathbf{u}_i) \\ \text{s.t. } & d\mathbf{x}_i = h_i(\mathbf{x}_i, \mathbf{x}_i^-, \mathbf{u}_i) dt + \sigma d\omega_i(t). \end{aligned} \quad (21)$$

To this end, the cooperative control of a large number of UAVs in complex environments is formulated as a noncooperative differential game. Therefore, this is an  $N$ -player noncooperative game whose well-known solution is the Nash equilibrium (NE), i.e., the control decisions under which no UAV can unilaterally decrease its cost [22].

### 3.2. Mean-Field Game Control Formulation

As the UAV-swarm number  $N$  increases, the complexity of solving the differential game in (21) will increase significantly. Mean-field games (MFGs) are a class of problems that encode large populations of interacting agents into systems of coupled partial differential equations, which overcomes the difficulties of large-scale and information sharing. For the current UAV, the neighborhood interaction of formation and collision avoidance, and the environment interaction of obstacle avoidance will cause the UAV to need a great amount of communication resources and computing resources. Traditional optimization and control methods need to separately deal with the increasing interactions, which leads to the problem of dimensionality explosion. Therefore, we reformulate the large-scale UAVs optimal state control problem in complex scenarios as a mean-field game control (MFGC) problem. Under the framework of MFGC, a generic agent only react to the collective behaviors (mean-field) of all agents instead of the behavior of each agent, which greatly reduces the amount of communication and computation. Now, we can drop the index  $i$  since agents are indistinguishable in MFG. Let  $\rho(\mathbf{x}, t)$  denote the probability of state  $\mathbf{x}$  at time  $t$ ; then, the cost functional (14) is now transformed into

$$\mathcal{J}(\mathbf{x}, \rho, \mathbf{u}) = \int_0^T \left\{ \int_{\Omega} L(\mathbf{x}, \mathbf{u}) \rho(\mathbf{x}, t) d\mathbf{x} + \mathcal{F}(\mathbf{x}, \rho(\mathbf{x}, t)) \right\} dt + \mathcal{G}(\rho(\cdot, T)). \quad (22)$$

Meanwhile, the state dynamics in (11) are transformed into

$$d\mathbf{x} = h(\mathbf{x}(t), \rho(\mathbf{x}, t), \mathbf{u}(\mathbf{x}, t)) dt + \sigma d\omega(t). \quad (23)$$

According to Ito's lemma [23], (23) can be represented in terms of the mean field  $\rho(\mathbf{x}, t)$  and then will be equivalent to the Fokker–Planck equation given by

$$\partial_t \rho - \frac{\sigma^2}{2} \Delta \rho + \nabla \cdot (\rho h) = 0. \quad (24)$$

with cost functional (22) and the Fokker–Planck equation in (24), the high-dimensional MFG which describes the complex-scenario cooperative control of a large number of UAVs is summarized as

$$\begin{aligned} & \inf_{\rho, \mathbf{u}} \mathcal{J}(\mathbf{x}(t), \rho(\mathbf{x}, t), \mathbf{u}(\mathbf{x}, t)) \\ \text{s.t. } & \partial_t \rho - \frac{\sigma^2}{2} \Delta \rho + \nabla \cdot (\rho h) = 0 \\ & \rho(\mathbf{x}, 0) = \rho_0(\mathbf{x}), \end{aligned} \quad (25)$$

where  $\rho_0(\mathbf{x})$  is the initial probability distribution of the UAV swarm's states. To this end, the state control for a large number of UAVs is formulated as a high-dimensional MFG. The agents forecast a distribution of the population  $\{\rho(\cdot, t)\}_{t=0}^T$  and aim at minimizing their cost. Therefore, at a Nash equilibrium, we have that, for every  $\mathbf{x} \in \mathbb{R}^n$ ,

$$\mathcal{J}(\mathbf{x}, \rho, \hat{\mathbf{u}}) \leq \mathcal{J}(\mathbf{x}, \rho, \mathbf{u}), \quad \forall \mathbf{u} \in \mathcal{U}, \quad (26)$$

where  $\hat{\mathbf{u}}$  is the equilibrium strategy of an agent at state  $\mathbf{x}$ . Here, we assume that agents are small, and their unilateral actions do not alter the density  $\rho$  [14]. With finite UAVs, it yields an MF approximation that achieves the  $\epsilon$ -NE [10,22].

### 3.3. GAN-Based Algorithm for Complex-Scenario MFGs

In this section, we propose a generative-adversarial-network (GAN)-based approach to solving the high-dimensional MFG in (25). Inspired by Wasserstein GANs [24], APAC-Net [15,25], we use the variational primal–dual structure of MFG and phrase (25) as a convex–concave saddle-point problem. Then, we propose a GAN-based algorithm to solve the complex-scenario MFG in (25).

Now, we show the underlying primal–dual structure of the complex-scenario MFG and derive the convex–concave saddle-point problem equivalent to (25). Denote  $\phi$  as the Lagrange multiplier, we can put the differential constraint in (24) (Fokker–Planck equation) into the cost function in (22) to obtain the following extended cost function

$$\sup_{\phi} \inf_{\rho, \mathbf{u}} \int_0^T \left\{ \int_{\Omega} \rho(\mathbf{x}, t) L(\mathbf{x}, \mathbf{u}(\mathbf{x}, t)) d\mathbf{x} + \mathcal{F}(\rho(\cdot, t)) \right\} dt + \mathcal{G}(\rho(\cdot, T)) - \int_0^T \int_{\Omega} \phi(\mathbf{x}, t) \left( \partial_t \rho - \frac{\sigma^2}{2} \Delta \rho + \nabla \cdot (\rho(\mathbf{x}, t) h(\mathbf{x}, t)) \right) d\mathbf{x} dt. \quad (27)$$

Minimizing concerning  $\mathbf{u}$  to get the Hamiltonian via  $H(\mathbf{x}, p) = \inf_u \{ph + L(\mathbf{x}, \mathbf{u})\}$  and integrating by parts, we can rewrite (27) as

$$\inf_{\rho} \sup_{\phi} \int_0^T \left\{ \int_{\Omega} \left( \partial_t \phi + \frac{\sigma^2}{2} \Delta \phi + H(\mathbf{x}, \nabla \phi) \right) \rho(\mathbf{x}, t) d\mathbf{x} + \mathcal{F}(\rho(\cdot, t)) \right\} dt + \int_{\Omega} \phi(\mathbf{x}, 0) \rho_0(\mathbf{x}) d\mathbf{x} - \int_{\Omega} \phi(\mathbf{x}, T) \rho(\mathbf{x}, T) d\mathbf{x} + \mathcal{G}(\rho(\cdot, T)). \quad (28)$$

The Formular (28) is the cornerstone of our algorithm.

We solve (28) by training a GAN-like neural network. Although the idea is inspired by Wasserstein GANs [24] and APAC-Net [15,25], the loss function of our algorithm considers the formation similarity  $\mathcal{F}^1$  and dense obstacle avoidance  $\mathcal{F}^2$ , which provides greater flexibility for large-scale agents maneuvering; details are shown in Algorithm 1. We show the structure and training process of our GAN-based neural network in Figure 1.

**Algorithm 1** GAN-based algorithm for complex-scenario MFGs

**Require:**  $\sigma$  diffusion parameter,  $H$  Hamiltonian,  $g$  terminal cost,  $f$  interaction term.

**Require:** Initialize neural networks  $N_\omega$  and  $N_\theta$ , batch size  $B$ .

**Require:** Set  $\phi_\omega$  and  $G_\theta$  as in (29).

**while** not converged **do**

**train**  $\phi_\omega$ :

        Sample batch  $\{(\mathbf{z}_b, t_b)\}_{b=1}^B$  where  $\mathbf{z}_b \sim \rho_0$  and  $t_b \sim \text{Unif}(0, T)$

        Obtaining generated data  $\{(\mathbf{x}_b, t_b)\}_{b=1}^B, \mathbf{x}_b \leftarrow G_\theta(\mathbf{z}_b, t_b)$ .

        Update discriminator parameters  $\omega$  to minimize  $\ell_{\text{total}} = \ell_0 + \ell_t + \ell_{\text{HJB}}$

$\omega \leftarrow \omega - \eta_1 \nabla \ell_{\text{total}}(\omega)$

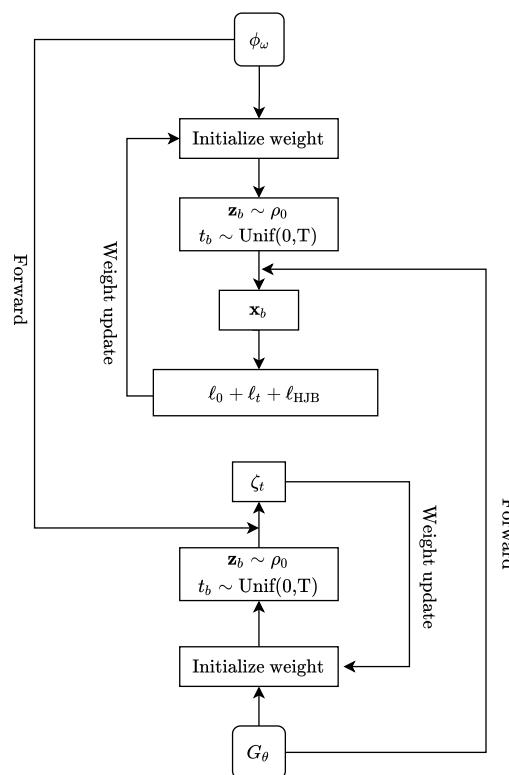
**train**  $G_\theta$ :

        Sample batch  $\{(\mathbf{z}_b, t_b)\}_{b=1}^B$  where  $\mathbf{z}_b \sim \rho_0$  and  $t_b \sim \text{Unif}(0, T)$

        Update generator parameters  $\theta$  to minimize  $\zeta_t$

$\theta \leftarrow \theta - \eta_2 \nabla \zeta_t(\theta)$

**end while**



**Figure 1.** Visualization of the structure and training process of our GAN-based neural network. Its training process is divided into two coupled alternating training parts—generator and discriminator.

First, we initialize the neural networks  $N_\omega(\mathbf{x}, t)$  and  $N_\theta(\mathbf{z}, t)$ , then set

$$\phi_\omega(\mathbf{x}, t) = (1 - t)N_\omega(\mathbf{x}, t) + tg(\mathbf{x}), G_\theta(\mathbf{z}, t) = (1 - t)\mathbf{z} + tN_\theta(\mathbf{z}, t), \quad (29)$$

where  $\mathbf{z} \sim \rho_0$  are samples drawn from the initial distribution. The formulation of  $\phi_\omega$  and  $G_\theta$  in (29) automatically satisfy the terminal and initial condition, respectively. Moreover, our algorithm encodes the underlying structure of complex-scenario MFGs via (28) and (29), which absolves the neural networks learning for the entire solution of the MFG from scratch.

Our approach for training neural networks includes parallel alternately training  $G_\theta$  (the state density distribution) and  $\phi_\omega$  (the value function). In order to obtain the equilibrium of the complex-scenario MFG, we train  $\phi_\omega$  by first sampling a batch  $\{\mathbf{z}_b\}_{b=1}^B$  from  $\rho_0$ , and  $\{t_b\}_{b=1}^B$  uniformly from  $[0, T]$ . Then, we compute the push-forward states  $\mathbf{x}_b = G_\theta(\mathbf{z}_b, t_b)$  for  $b = 1, \dots, B$ . The main loss item for training the discriminator  $\phi_\omega$  is

$$\text{loss}_\phi = \underbrace{\frac{1}{B} \sum_{b=1}^B \phi_\omega(\mathbf{x}_b, 0)}_{\ell_0} + \underbrace{\frac{1}{B} \sum_{b=1}^B \partial_t \phi_\omega(\mathbf{x}_b, t_b) + \frac{\sigma^2}{2} \Delta \phi_\omega(\mathbf{x}_b, t_b) + H(\nabla_{\mathbf{x}} \phi_\omega(\mathbf{x}_b, t_b), \mathbf{x}_b)}_{\ell_t}, \quad (30)$$

where we can optionally add a regularization term

$$\ell_{\text{HJB}} = \lambda \frac{1}{B} \sum_{b=1}^B \left\| \partial_t \phi_\omega(\mathbf{x}_b, t_b) + \frac{\sigma^2}{2} \Delta \phi_\omega(\mathbf{x}_b, t_b) + H(\nabla_{\mathbf{x}} \phi_\omega(\mathbf{x}_b, t_b), \mathbf{x}_b) + f(\mathbf{x}_b, t_b) \right\| \quad (31)$$

to penalize deviations from the HJB equations. This extra regularization term has also been found effective in, e.g., Wasserstein GANs. Finally, we backpropagate the total loss to update the weights of the discriminator  $\phi_\omega$ . To train the generator, we again sample  $\{\mathbf{z}_b\}_{b=1}^B$  and  $\{t_b\}_{b=1}^B$  as before. The loss of the generator is given by

$$\text{loss}_G = \underbrace{\frac{1}{B} \sum_{b=1}^B \partial_t \phi_\omega(G_\theta(\mathbf{z}_b, t_b), t_b) + \frac{\sigma^2}{2} \Delta \phi_\omega(G_\theta(\mathbf{z}_b, t_b), t_b) + H(\nabla_{\mathbf{x}} \phi_\omega(G_\theta(\mathbf{z}_b, t_b), t_b), \mathbf{x}_b) + f(G_\theta(\mathbf{z}_b, t_b), t_b)}_{\zeta_t}. \quad (32)$$

At last, we backpropagate this loss to update the weights of  $G_\theta$ . In conclusion, in each time slot  $t \in [0, T]$ , the neural networks will be trained. The generator  $G_\theta$  will generate the state distribution at time  $t$  and the discriminator  $\phi_\omega$  will get the result of the value function at time  $t$ .

#### 4. Simulation Results

In this section, to verify the effectiveness and robustness of our method, we conduct a series of simulation experiments of distributed aerial large-scale UAVs. To demonstrate the competitiveness of our method, we compare it against baseline methods regarding their performances at last.

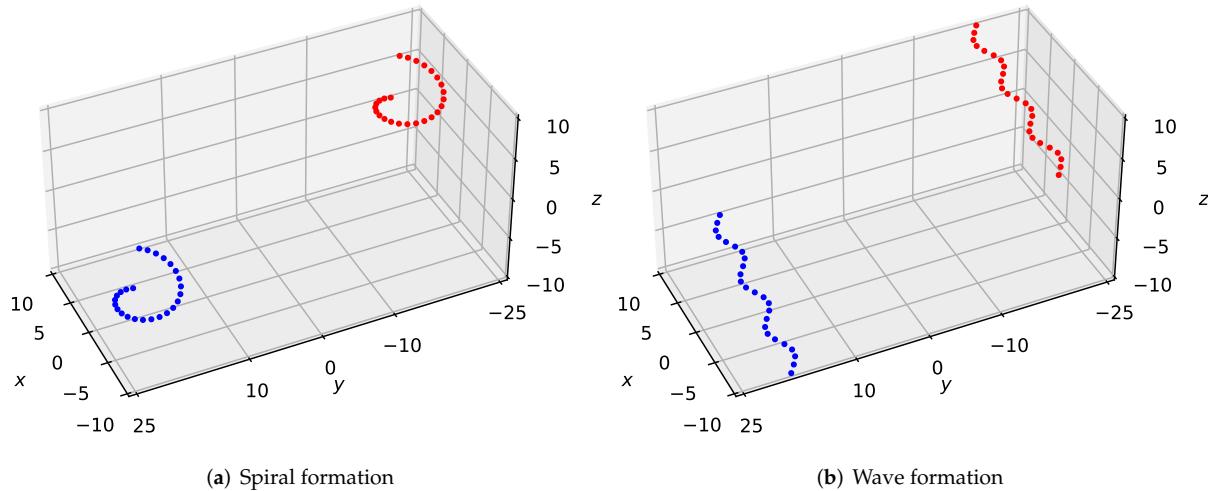
##### 4.1. Simulation Parameters

For the model hyperparameters, we set  $c_1 = 0.5$  (the weight of the control effort in (15)),  $c_2 = 5$  (the weight of the terminal cost in (16)), and  $l_1 = 5, l_2 = 10, l_3 = 5$  (the weights of the interaction cost in (17)). For the neural networks, there are three linear hidden layers with 100 hidden units per layer in both networks. Residual neural networks (ResNet) are used for both networks, with a skip connection weight of 0.5. Tanh activation function is used in  $\phi_\omega$  while ReLU activation function is used in  $G_\theta$ . We choose ADAM as the optimizer with  $\beta = (0.5, 0.9)$ ,  $\eta_1 = 4 \times 10^{-4}$  (the learning rate for  $\phi_\omega$ ),  $\eta_2 = 1 \times 10^{-4}$  (the learning rate for  $G_\theta$ ), weight decay of  $10^{-4}$  for both networks, batch size 25, and  $\lambda = 1$  (the weight of the HJB regularization term in (31)).

##### 4.2. Formation Performance of Large-Scale UAVs

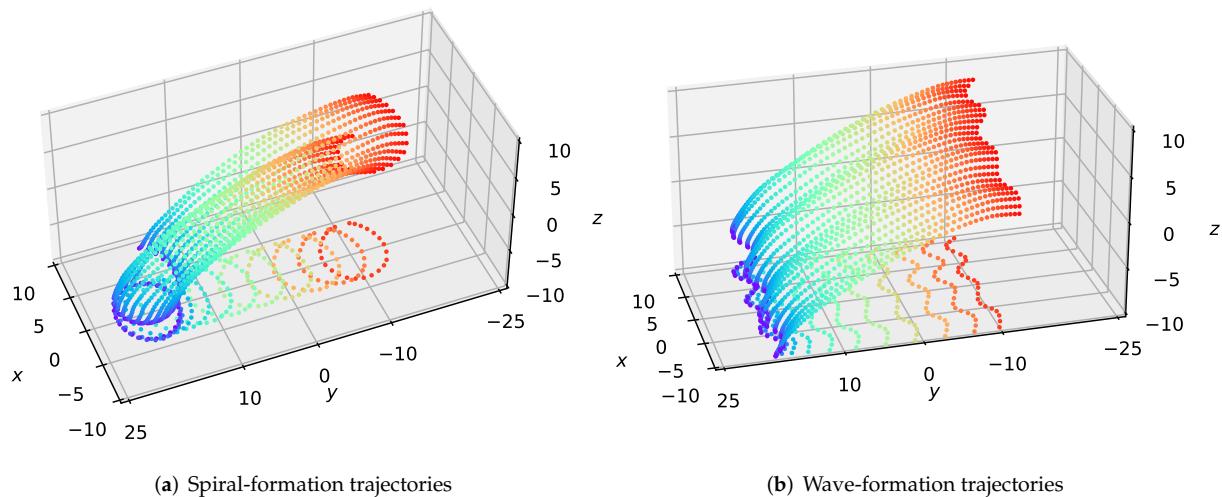
To validate the feasibility and robustness of our method for large-scale UAVs formation, we simulated twenty-five drones flying in desired formations from the bottom left side to the top right side in the 3-D scene, as shown in Figure 2. We experimented with two forms of formations—spiral formation and wave formation, which could confirm the generality of our approach. In Figure 2a, we set the initial points of the drones as  $(2\cos(\frac{2\pi i}{25}), 2\sin(\frac{2\pi i}{25}) + 18, \frac{5i}{24} - 10)$ ,  $i = 0, 1, \dots, 24$ ; the target points of the drones are  $(2\cos(\frac{2\pi i}{25}), 2\sin(\frac{2\pi i}{25}) - 18, \frac{5i}{24} + 5)$ ,  $i = 0, 1, \dots, 24$ . In Figure 2b, the initial points of the drones are  $(\frac{20i}{25} - 10, \sin(\frac{20i}{25}) - 10) +$

$18, \frac{5i}{24} - 10)), i = 0, 1, \dots, 24$ ; the target points of the drones are  $(\frac{20i}{25} - 10, \sin(\frac{20i}{25}) - 10 - 18, \frac{5i}{24} + 5), i = 0, 1, \dots, 24$ .



**Figure 2.** A large-scale desired formation consisting of twenty-five quadrotors traverses a 3-D environment from the bottom left side to the top right side.

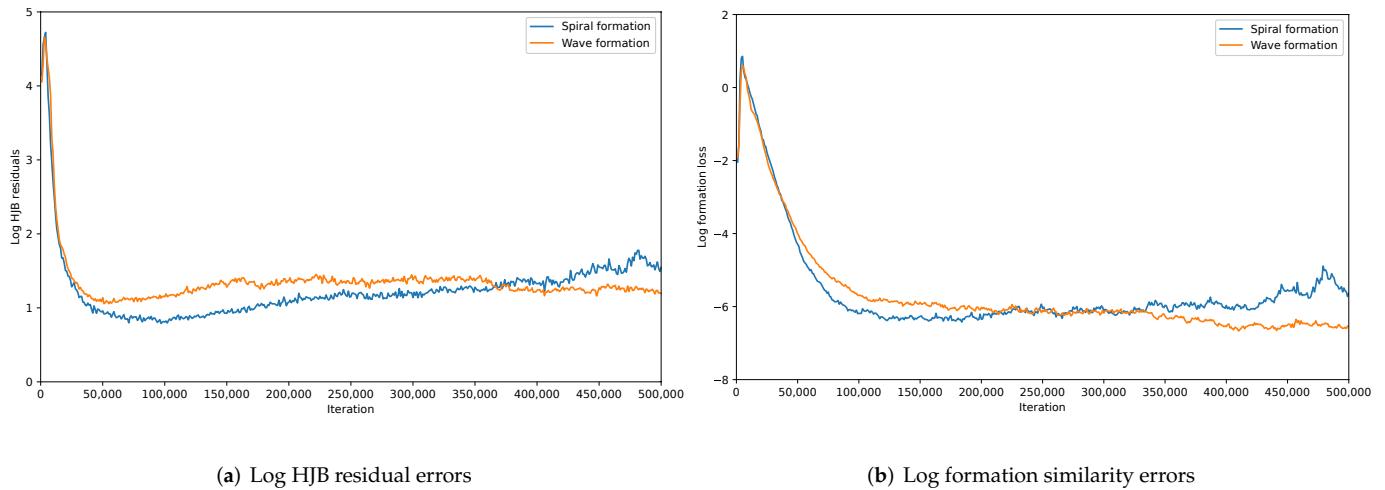
The test results are shown in Figure 3, which demonstrates that our method can maintain the formation of large-scale swarms in 3-D environments. To certify the generality of our approach in large-scale formations, we design two forms of formations consisting of twenty-five quadrotors—spiral and wave formations. As depicted in Figure 3a,b, the  $xoy$ -plane projections illustrate the shape contour of the formation maintenance, which means that the swarm maintains the desired formation well during the flight.



**Figure 3.** The visualization of the executed trajectories for the formation flight of large-scale UAVs. The  $xoy$ -plane projections represent the outline of the shape.

The convergence of our proposed MFG control algorithm and the effectiveness and stability of formation control under different forms of formations are shown in Figure 4. In Figure 4a, we plot the HJB residual errors, i.e.,  $\ell_{\text{HJB}}$  in Algorithm 1, which shows the convergence of MFGC. Without an efficient strategy control, the HJB residuals under different formations are relatively high. HJB residuals drop fast after we apply a series of controls. After around  $5 \times 10^4$  iterations, the error curves tend to be stable when we obtain

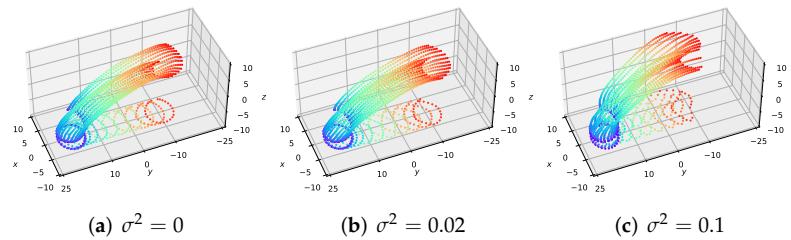
an optimal control for UAVs. In Figure 4b, we plot the formation loss. Without a proper design of the formation cost term, there is no effective formation control, and the formation loss under different formations is relatively high. Formation loss drops fast after we apply our formation cost term. After around  $1 \times 10^5$  iterations, the error curves tend to be stable when we obtain an optimal formation control for UAVs.



**Figure 4.** Illustration of MFG convergence and formation stability.

#### 4.3. Effect of Volatility Term $\sigma$ on Formation

We investigate the effect of the volatility term  $\sigma$  on the behavior of the MFGC solutions—the executed trajectories for the formation flight of large-scale UAVs. In Figure 5, we show the solutions for MFGCs using  $\sigma = 0, \sqrt{2} \times 10^{-1}$  and  $\sqrt{10} \times 10^{-1}$ . As  $\sigma$  increases, the density of UAVs widens along the paths, and the desired formation maintenance of UAVs decreases due to the added diffusion term in the MFG control equations in (25). From Table 1, it can be seen that larger volatility  $\sigma$  corresponds to greater formation cost  $\mathcal{F}^1$ , but the collision avoidance cost  $\mathcal{F}^3$  is almost invariant. These results are consistent with those in Figure 5.



**Figure 5.** Comparison of the executed trajectories for the formation flight of large-scale UAVs about volatility parameter  $\sigma = 0, \sqrt{2} \times 10^{-1}$  and  $\sqrt{10} \times 10^{-1}$ .

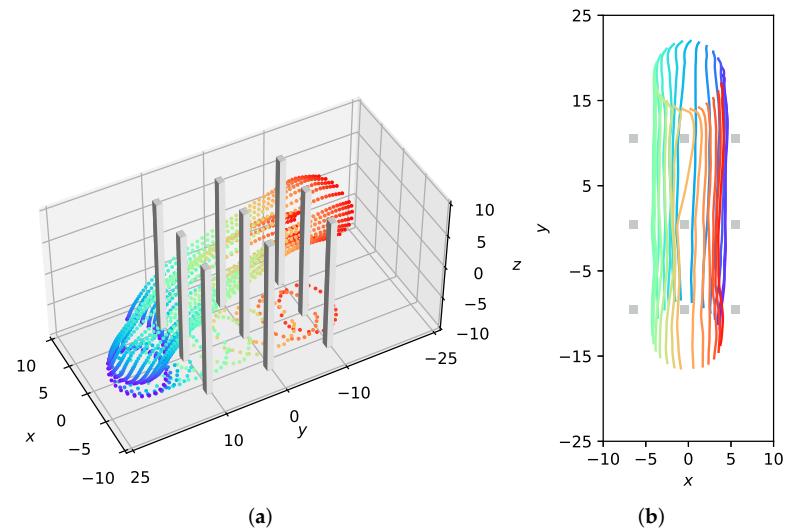
**Table 1.** Comparison with interaction costs.

Volatility Parameter $\sigma$	Formation Cost $\mathcal{F}^1$	Collision Avoidance Cost $\mathcal{F}^3$
0	$1.41 \times 10^{-3}$	$6.35 \times 10^{-2}$
$\sqrt{2} \times 10^{-1}$	$6.34 \times 10^{-2}$	$6.34 \times 10^{-2}$
$\sqrt{10} \times 10^{-1}$	$4.30 \times 10^{-1}$	$6.14 \times 10^{-2}$

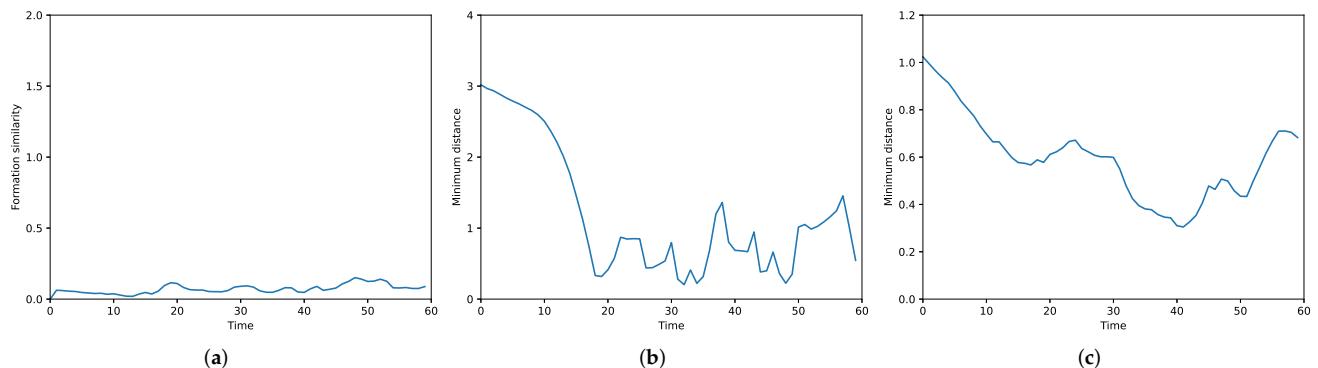
#### 4.4. Performance of Dense Obstacle Avoidance for Large-Scale UAVs Formation Flying

To verify the effectiveness of our method with large-scale UAVs formation flying through an obstacle-rich area, we design a spiral formation consisting of twenty-five quadrotors. A packed area of nine cuboid obstacles is set up in the simulation. As depicted

in Figure 6, the swarm successfully avoids the obstacles and the desired formation is well preserved during the flight. Under our MFG control, the formation similarity error is steadily close to 0 as shown in Figure 7a; the executed trajectories are zero-collision, which can be seen from Figure 7b,c.



**Figure 6.** The visualization of the executed trajectories for large-scale UAVs formation flying through an obstacle-rich area. (a) Full view of the trajectories. (b)  $xoy$ -plane projection trajectories.



**Figure 7.** Illustration of the performance of interaction terms for MFG control ( $\mathcal{F}^1, \mathcal{F}^2, \mathcal{F}^3$ ). (a) Formation similarity error. (b) Minimum distance between UAVs and obstacles. (c) Minimum distance between UAVs.

#### 4.5. Comparison with Baselines

In the last part of the simulation results, we verify the superiority of our method by comparing it to some typical baseline methods in Table 2. From Table 2, it can be seen that our approach simplifies large-scale communication and handles the most complex application scenarios. For more details of these baseline methods, please refer to the related documents [4,7,9,13], etc.

**Table 2.** Comparison with baseline methods.

Method	Scene	Scale of UAVs	Scene Complexity	Communication
[4]	Formation flight and obstacle avoidance	Small	0.67 <sup>1</sup>	$\mathcal{O}(N)$ <sup>2</sup>
[9]	Cluster flight	Large	0.5	$\mathcal{O}(1)$
[13]	Cluster flight and obstacle avoidance	Large	0.67	$\mathcal{O}(1)$
[7]	Formation flight and dense obstacle avoidance	Small	0.83	$\mathcal{O}(N)$
Ours	Formation flight and dense obstacle avoidance	Large	1	$\mathcal{O}(1)$

<sup>1</sup> The measurement method of scene complexity is as follows: here it is a scoring system, [cluster flight, formation flight] = [1', 2']; [obstacle avoidance, dense obstacle avoidance] = [1', 2']; [small, large] = [1', 2']. We accumulate the scores for each literature experiment scene according to each item and finally normalize them. <sup>2</sup>  $\mathcal{O}()$  is infinitesimal of the same order.

## 5. Conclusions

In this paper, we propose a large-scale agent trajectory optimization method capable of navigating large-scale swarms in formations while avoiding dense obstacles. First, two types of differentiable mean-field terms are developed, where the overall similarity distance between large-scale 3-D formations and the potential energy value of dense 3-D obstacles are quantified respectively. Second, we embed these two terms into a designed mean-field game control framework that simultaneously satisfies the dynamic constraints and minimizes energy cost, formation similarity error, and collision penalty. This framework solves the communication and computing difficulties caused by complex scenarios of large-scale multiagent systems and formulates the trade-offs between formation and obstacle avoidance in large-scale contexts. Finally, the solid performance of our method in simulations of the formation and dense obstacle avoidance for large-scale UAVs validates its practicality and efficiency. The comparison with baseline methods shows the advanced nature of our method. In the future, we will consider the application and adaption of this mean-field game control framework in more complex scenarios of multiagent systems, such as complex communication conditions (time-varying, partial rejection) and dense mixed obstacles (static, dynamic).

**Author Contributions:** Conceptualization, G.W. and Z.L.; methodology, G.W. and W.Y.; software, G.W.; validation, W.Y. and X.Z.; formal analysis, G.W.; investigation, Z.L.; writing—original draft preparation, G.W. and Z.L.; writing—review and editing, W.Y. and X.Z.; visualization, G.W.; supervision, W.Y. and X.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Science and Technology Innovation 2030-Key Project under Grant 2020AAA0108200.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** We thank Zhiming Zheng for his fruitful discussions, valuable opinions, and guidance.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

CPSs	Cyberphysical Systems
MF	Mean Field
MFG	Mean-Field Game
MFGC	Mean-Field Game Control
2-D	Two-Dimensional
3-D	Three-Dimensional
UAV	Unmanned Aerial Vehicle
APF	Artificial Potential Field
GANs	Generative Adversarial Neural Networks
CA-Net	Coupled Alternating Neural Network
HJB	Hamilton–Jacobi–Bellman (partial differential equation)

## References

- Lin, X.; Su, G.; Chen, B.; Wang, H.; Dai, M. Striking a Balance Between System Throughput and Energy Efficiency for UAV-IoT Systems. *IEEE Internet Things J.* **2019**, *6*, 10519–10533. [[CrossRef](#)]
- Zhang, J.; Xu, W.; Gao, H.; Pan, M.; Han, Z.; Zhang, P. Codebook-Based Beam Tracking for Conformal Array-Enabled UAV mmWave Networks. *IEEE Internet Things J.* **2021**, *8*, 244–261. [[CrossRef](#)]
- Zhang, J.; Yan, J.; Zhang, P.; Kong, X. Collision Avoidance in Fixed-Wing UAV Formation Flight Based on a Consensus Control Algorithm. *IEEE Access* **2018**, *6*, 43672–43682. [[CrossRef](#)]
- Zhang, J.; Yan, J.; Zhang, P. Fixed-Wing UAV Formation Control Design With Collision Avoidance Based on an Improved Artificial Potential Field. *IEEE Access* **2018**, *6*, 78342–78351. [[CrossRef](#)]
- Shao, S.K.; Peng, Y.; He, C.L.; Du, Y. Efficient Path Planning for UAV Formation via Comprehensively Improved Particle Swarm Optimization. *ISA Trans.* **2020**, *97*, 415–430. [[CrossRef](#)] [[PubMed](#)]
- Wu, E.; Sun, Y.; Huang, J.; Zhang, C.; Li, Z. Multi UAV Cluster Control Method Based on Virtual Core in Improved Artificial Potential Field. *IEEE Access* **2020**, *8*, 131647–131661. [[CrossRef](#)]
- Quan, L.; Yin, L.J.; Xu, C.; Gao, F. Distributed Swarm Trajectory Optimization for Formation Flight in Dense Environments. In Proceedings of the IEEE International Conference on Robotics and Automation, Philadelphia, PA, USA, 23–27 May 2022.
- Obstacle Avoidance of Resilient UAV Swarm Formation with Active Sensing System in the Dense Environment. Available online: <https://arxiv.org/abs/2202.13381> (accessed on 27 February 2022).
- Kim, H.; Park, J.; Bennis, M.; Kim, S.L. Massive UAV-to-Ground Communication and its Stable Movement Control: A Mean-Field Approach. In Proceedings of the IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications, Kalamata, Greece, 25–28 June 2018.
- Shiri, H.; Park, J.; Bennis, M. Communication-Efficient Massive UAV Online Path Control: Federated Learning Meets Mean-Field Game Theory. *IEEE Trans. Commun.* **2020**, *68*, 6840–6857. [[CrossRef](#)]
- Chen, D.; Qi, Q.; Zhuang, Z.; Wang, J.; Liao, J.; Han, Z. Mean Field Deep Reinforcement Learning for Fair and Efficient UAV Control. *IEEE Internet Things J.* **2021**, *8*, 813–828. [[CrossRef](#)]
- Xu, W.; Xiang, L.; Zhang, T.; Pan, M.; Han, Z. Cooperative Control of Physical Collision and Transmission Power for UAV Swarm: A Dual-Fields Enabled Approach. *IEEE Internet Things J.* **2022**, *9*, 2390–2403. [[CrossRef](#)]
- Gao, H.; Lee, W.J.; Kang, Y.H.; Li, W.C.; Han, Z.; Osher, S.; Poor, V. Energy-Efficient Velocity Control for Massive Numbers of UAVs: A Mean Field Game Approach. *IEEE Trans. Veh. Technol.* **2022**, *71*, 6266–6278. [[CrossRef](#)]
- Ruthotto, L.; Osher, S.J.; Li, W.; Nurbekyan, L.; Fung, S.W. A Machine Learning Framework for Solving High-Dimensional Mean Field Game and Mean Field Control Problems. *Proc. Natl. Acad. Sci. USA* **2020**, *117*, 9183–9193. [[CrossRef](#)] [[PubMed](#)]
- Lin, A.T.; Fung, S.W.; Li, W.C.; Nurbekyan, L.; Osher, S.J. Alternating the Population and Control Neural Networks to Solve High-Dimensional Stochastic Mean-Field Games. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2024713118. [[CrossRef](#)] [[PubMed](#)]
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, Quebec, Canada, 8–13 December 2014; pp. 2672–2680.
- Wang, G.F.; Yao, W.; Zhang, X.; Niu, Z.J. Coupled Alternating Neural Networks for Solving Multi-Population High-Dimensional Mean-Field Games with Stochasticity. *IEEE Trans. Cybern.* **2021**, submitted.
- Tantardini, M.; Ieva, F.; Tajoli, L.; Piccardi, C. Comparing Methods for Comparing Networks. *Sci. Rep.* **2019**, *9*, 17557. [[CrossRef](#)] [[PubMed](#)]
- Chang, K.; Xia, Y.; Huang, K. UAV Formation Control Design with Obstacle Avoidance in Dynamic Three-dimensional Environment. *SpringerPlus* **2016**, *5*, 1124. [[CrossRef](#)] [[PubMed](#)]
- Onken, D.; Nurbekyan, L.; Li, X.J.; Fung, S.W.; Osher, S.; Ruthotto, L. A Neural Network Approach for High-Dimensional Optimal Control. *arXiv* **2022**, arXiv:2104.03270v1.
- García Carrillo, L.R.; Dzul López, A.E.; Lozano, R.; Pégard, C. *Quad Rotorcraft Control*; Springer: London, UK, 2013; pp. 23–34.

22. Lasry, J.M.; Lions, P.L. Mean field games. *Jpn. J. Math.* **2007**, *2*, 229–260. [[CrossRef](#)]
23. Schulte, J.M. Adjoint Methods for Hamilton-Jacobi-Bellman Equations. Ph.D. Thesis, University of Munster, Munster, Germany, November 2010.
24. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the Machine Learning Research, International Convention Centre, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 214–223.
25. Gao, H.; Lin, A.; Banez, R.A.; Li, W.C.; Han, Z.; Osher, S.; Poor, H.V. Belief and Opinion Evolution in Social Networks: A High-Dimensional Mean Field Game Approach, In Proceedings of IEEE International Conference on Communications, Montreal, QC, Canada, 14–23 June 2021.